Operand: Pricing and Promotion White Paper

January 2025

Re-envisioning Price and Promotional Optimization for E-Commerce

Operand is building the autonomous data team for consumer brands. The first core capability they've rolled out in January 2025 is a new approach to pricing and promotional optimization.

1 Introduction

In today's high-stakes retail environment, many so-called "dynamic" pricing and promotion solutions fail to deliver tangible, bottom-line results. The main issue is that pricing decisions do not exist in a vacuum: they depend on every facet of a business, from advertising spend and inventory constraints to competitor dynamics and macroeconomic trends. Too often, legacy tools ignore this web of interdependencies or rely on generic competitor data. They may claim to be "automated," yet they overlook real-time changes in substitute products and fail to connect the dots between internal store data and rapidly evolving market conditions.

Operand addresses this problem by deploying highly intelligent agentic systems that ingest and interpret all available signals—whether it's ad platform metrics, SKU-level stock forecasts, or dynamically scraped competitor prices for semantically similar products. Backed by an ex-McKinsey, human-in-the-loop layer for critical oversight, these agents ensure that no relevant data is overlooked. The result is a comprehensive, month-by-month pricing and promotion strategy that learns continually from real-world conditions and aligns with both short-term profit goals and long-term brand objectives. This white paper presents how Operand's framework unifies robust demand modeling, multi-constraint optimization, and agentic data processing to drive sustained revenue and margin growth in an ever-changing retail landscape.

2 Data Preparation & Infrastructure

A robust data pipeline ensures the continuity and reliability of your pricing strategy. Brands generally collect data from diverse internal and external systems, which must be cleansed and merged to form a single source of truth for our agents to understand.

2.1 Data Ingestion

It is best practice to create an ETL (Extract, Transform, Load) pipeline that unifies data from CRMs, ERPs, marketing platforms, spreadsheets, and competitor-scraping APIs. Table 1 summarizes common data sources, their usage, and examples of associated fields.

Table 1: Key Data Sources

Examples	**Usage**	**Description**
Internal Sales Data	Determines historical demand patterns and pricing history	Transaction logs, unit sales, returns, timestamps
Costs & Margins	Informs profit calculations (variable vs. fixed costs)	COGS, shipping costs, overhead expenses
Customer Data	Enables segmentation and targeted promotions	Loyalty tiers, demographic info, online behavior
Inventory	Ensures feasibility of recommended price/promo combos	Stock levels, reorder points, lead times
Competitor Data	Benchmarks relative position and reveals market shifts	Rival pricing, promotions, # of sellers
Macroeconomic Indicators	Gauges broader economic trends impacting disposable income	Inflation rates, consumer sentiment, GDP growth
Market Trends	Accounts for external factors influencing demand spikes or dips	Seasonal events, category-level forecasts

2.2 Agentic Pre-Processing & Intelligent Cleanup

Despite systematic data ingestion, raw data often contains SKU mismatches, naming inconsistencies, or unaligned product identifiers across sales channels. Operand's agentic pre-processing solution, powered by a large language model (LLM), automatically resolves these discrepancies. This engine can identify that "BlueWidget-XL" and "BWXL" refer to the same product, unify them under a single code, and flag only the most ambiguous records for human review. By correcting store-level labeling quirks, it allows subsequent stages—such as feature engineering and modeling—to start with unified data.

2.3 Data Validation & Versioning

Once cleaned, each monthly dataset undergoes further validation to detect anomalies (e.g., suspiciously high sales spikes without a recorded promotion). The pipeline applies additional rules, such as ensuring no null values in critical fields (price, units sold, etc.). It then versions the final dataset under a unique timestamp, which facilitates historical back-testing. This step ensures the integrity of the analytics process, allowing future analysts to revisit the exact dataset that informed a specific price decision.

3 Feature Engineering

Feature engineering translates raw data into meaningful signals that a machine learning or econometric model can ingest.

3.1 Time-Series & Seasonal Features

Time-series features capture the cyclical nature of retail demand. Brands often mark each row with month, quarter, or year, and use cyclical encodings (e.g., sine and cosine transformations) to capture seasonal fluctuations. Custom flags can highlight holidays (e.g., Black Friday, Easter) or category-specific peak seasons (e.g., Back to School).

3.2 Demand-Related Features

Price and promotion history, competitor pricing, macroeconomic signals, and consumer behaviors form the core of demand-related features. Such features clarify how each SKU's sales patterns vary under discount events, competitor undercuts, or seasonal booms.

3.3 Lagged & Rolling Features

Since consumer demand can reflect inertia or seasonal carry-over, lagged features capture past behavior. Rolling averages or rolling standard deviations help smooth out short-term noise, revealing underlying trends. Table 2 offers typical feature transformations and their potential value.

Table 2: Sample Feature Engineering

Transformation	**Purpose**
Lagged Demand	Captures momentum from prior months
Rolling Mean of Price over 3-6 months	Smooths out short-term price fluctuations
Promotion Depth Indicator	Highlights promotional intensity (0–100% discount)
Competitor Price Gap	Shows relative positioning in the market
Macro Sentiment	Explains demand variation due to external factors
Seasonal Flag	1 if Month in {Nov, Dec}, else 0; identifies known holiday or event-driven spikes

4 Demand Modeling

Demand modeling predicts how many units a SKU can sell in a future period given a proposed price and promotional plan. Approaches range from simple log-log regressions to advanced machine learning.

$$\text{Log-Log Demand Model (Econometric)} \tag{1}$$

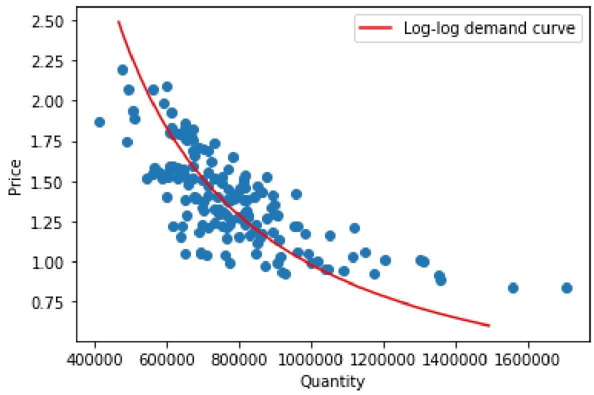

Figure 1: Sample Log-Log Demand Model (Econometric)

This specification reveals elasticity: a 1% change in price results in a β change in demand, all else equal. For richer patterns, Gradient Boosting or Random Forest algorithms are sometimes used to capture non-linearities and interactions, such as how discount efficacy might differ in holiday months vs. off-season.

4.1 Model Training & Diagnostics

Typically, companies split historical data chronologically (e.g., first 24 months for training, next 6 for validation, final 6 for testing) to mirror real-world forecasting. Cross-validation with a rolling window helps measure robustness over time. Residual plots and methods like SHAP values clarify whether the model systematically misses demand in certain scenarios (e.g., during large competitor promotions).

5 Price & Promotion Optimization Framework

After forecasting demand under different scenarios, the system shifts focus to profit optimization or an alternative goal (like revenue, market share, or brand-building constraints).

5.1 Objective Function

Brands commonly optimize for profit or margin. The profit equation typically factors in variable costs and direct promotional costs:

$$\text{Profit Maximization} = \sum(\text{Price} \times \text{Units Sold}) - \sum(\text{Variable Costs} + \text{Promotional Costs}) \tag{2}$$

5.2 Constraints

In a real-world scenario, brands cannot exceed inventory, violate contractual constraints (like MAP), or dip below brand-specified margin floors. Table 3 highlights common constraints.

Table 3: Common Constraints

Example	Description
Inventory	Ensure that total units sold do not exceed available stock
Margin Floor	Maintain minimum required profit margins for each SKU
Minimum Advertised Price (MAP)	Adhere to contractual pricing agreements with suppliers
Promotion Frequency	Limit discounts to a specified number of times per quarter
Brand Image	Restrict certain products from deep discounting to preserve brand perception

5.3 Optimization Methods

- **Analytical**: If the demand function is a simple parabola or log-linear, one may solve analytically for the derivative = 0.
- **Grid Search**: Enumerate feasible price points and promotion levels, then choose the highest-profit scenario.
- **Mixed-Integer or Linear Programming**: Useful when scaling to multiple SKUs with shared resource constraints (e.g., manufacturing capacity).
- **Reinforcement Learning**: An adaptive approach for real-time pricing, particularly in dynamic online marketplaces.

5.4 Process Verification

While we're working with OpenAI's unreleased models to push the barrier of intelligence with our agents, we want to ensure these results are accurate, which is why we manually check every single result. Whether it's our ex-McKinsey consultants or ex-AI researchers, we know how important this is and understand the necessity of verification.

6 Monthly Update & Deployment Process

A key advantage of this system is that it repeats every month (or even weekly/daily), adapting to new insights and recalibrating prices or promos.

6.1 Forecast Next Month's Demand

At month's end, the pipeline pulls in fresh data on sales, competitor moves, and macro trends. The LLM-based pre-processor resolves any newly introduced SKUs or categories, while standard validation rules handle anomalies. The demand model then forecasts the upcoming period's demand under different possible price and promotion scenarios.

6.2 Optimization & Recommendation

The profit-optimization engine uses these forecasts to evaluate each scenario:

1. **Calculate Profit**: For every feasible Price-Promo pair, compute based on Equation 2.
2. **Check Constraints**: Filter out or penalize scenarios violating margin floors, inventory caps, or MAP policies.
3. **Rank & Recommend**: Present the top options (or a recommended single option) to decision-makers, usually with a confidence interval around predicted demand.

6.3 Approval & Execution

While the system can autonomously propose changes, critical decisions—like large price cuts for premium items—may require senior management approval. Once approved, updates flow automatically to e-commerce platforms (e.g., Shopify, Amazon Seller Central) or physical store pricing systems. Marketing teams can coordinate promotional ads on Google or social media to reflect the new discount structure and brand messaging.

7 Conclusion

By methodically ingesting, cleaning, and synthesizing data each month—and then leveraging state-of-the-art demand models to optimize profit or strategic KPIs—brands can maintain agile, high-return pricing and promotion decisions. Operand's agentic approach to data processing and end-to-end recommendation ensures a unified, continuously improving framework. Over time, repeated monthly cycles accumulate knowledge from real-world feedback loops, producing ever more accurate forecasts and maximizing revenues, margins, and brand equity in an evolving marketplace.