Retail Pricing Strategy

Campaign Effectiveness and Predictive Modeling for Competitor Behavior

PREPARED FOR:

RETAILZ

JOSÉ MIGUEL MENDES
06/2025

01. OBJECTIVES

02. PROJECT STAGES

03. DATASET DESCRIPTION

04. BUSINESS QUESTIONS

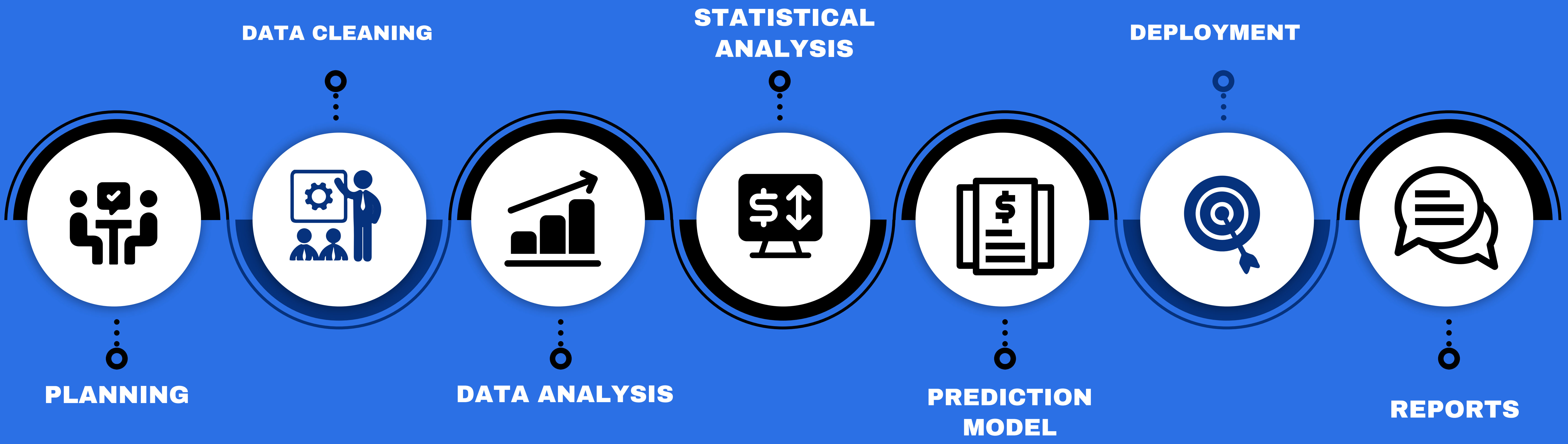
05. MODELING

06. DEPLOYMENT

OBJECTIVES

- Analyze historical competitor pricing behavior to identify discount patterns, seasonality, and competitiveness variations across categories.
- Detect reactive pricing dynamics among competitors.
- Develop a predictive pricing model, accessible via an API.

PROJECT STAGES



DATASET DESCRIPTION

- Period Covered:
  - January 3rd, 2023 → October 28th, 2024
- Granularity: Daily (one row per SKU per day)

- Total SKUs: 3,605

- Total Days: 664

- Zero sales only on holidays:
  - December 25, 2023
  - January 1, 2024

DATASET CLEANING

- Negative quantity values identified as input errors and removed from the dataset.
    - Invalid entries removed (quantity < 0): 292 rows

- Days with no sales treated as normal operational gaps, not missing data.

BUSINESS QUESTION ANALYSIS

AWKWARD
PROBLEM SOLUTIONS™

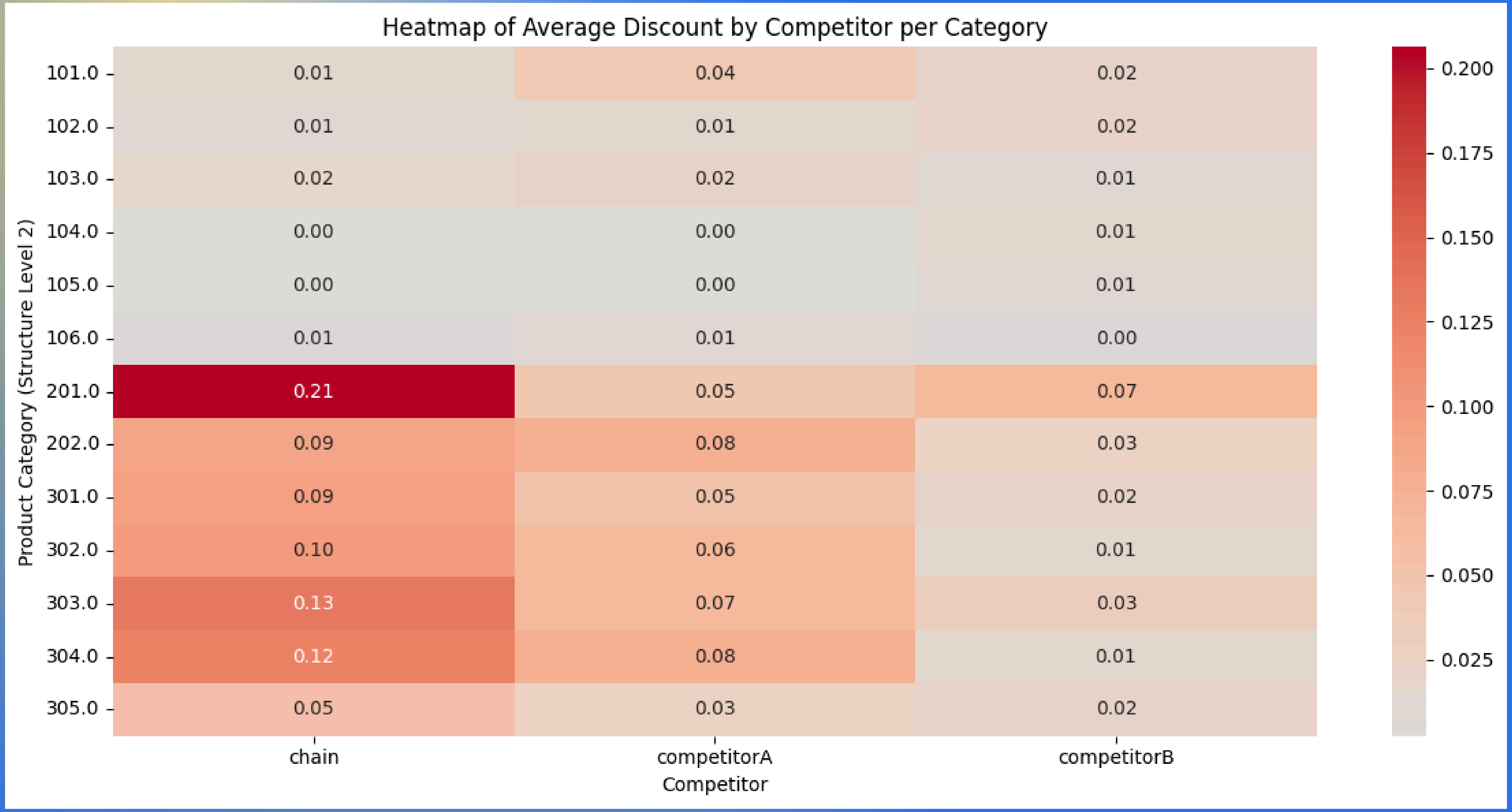
Which competitors tend to be more expensive/cheaper?

structure_level_2	cheapest_competitor	most_expensive_competitor
101.0	competitorB	chain
102.0	competitorB	competitorA
103.0	competitorB	competitorA
104.0	competitorB	competitorA
105.0	competitorB	chain
106.0	competitorB	chain
201.0	competitorB	competitorA
202.0	competitorB	chain
301.0	competitorB	competitorA
302.0	competitorB	chain
303.0	competitorB	competitorA
304.0	competitorB	competitorA
305.0	competitorB	chain

BUSINESS QUESTION ANALYSIS

AWKWARD
PROBLEM SOLUTIONS™

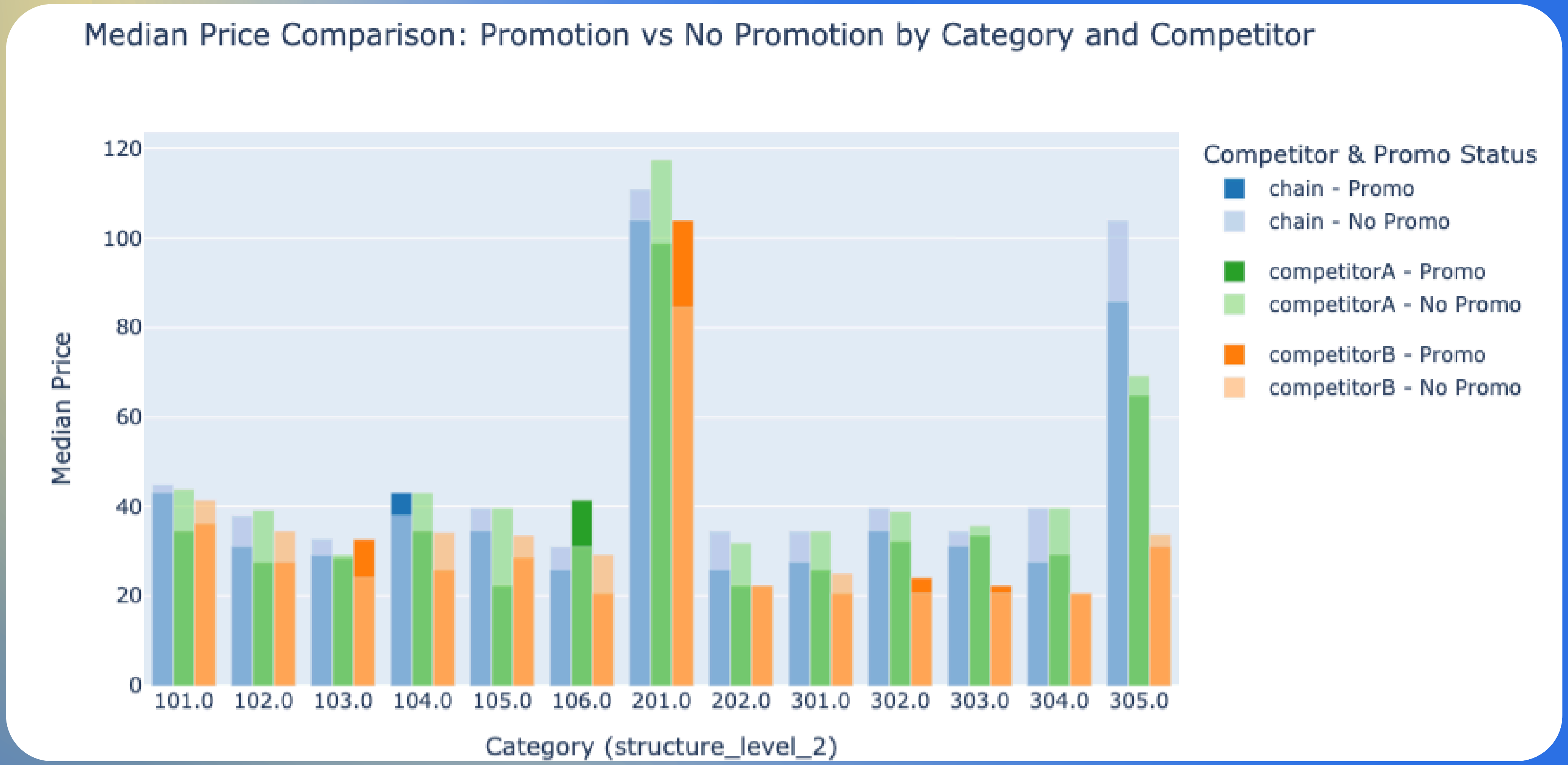
What is the average discount per competitor across product categories?



BUSINESS QUESTION ANALYSIS

AWKWARD
PROBLEM SOLUTIONS™

How does price vary depending on promotional strategies?

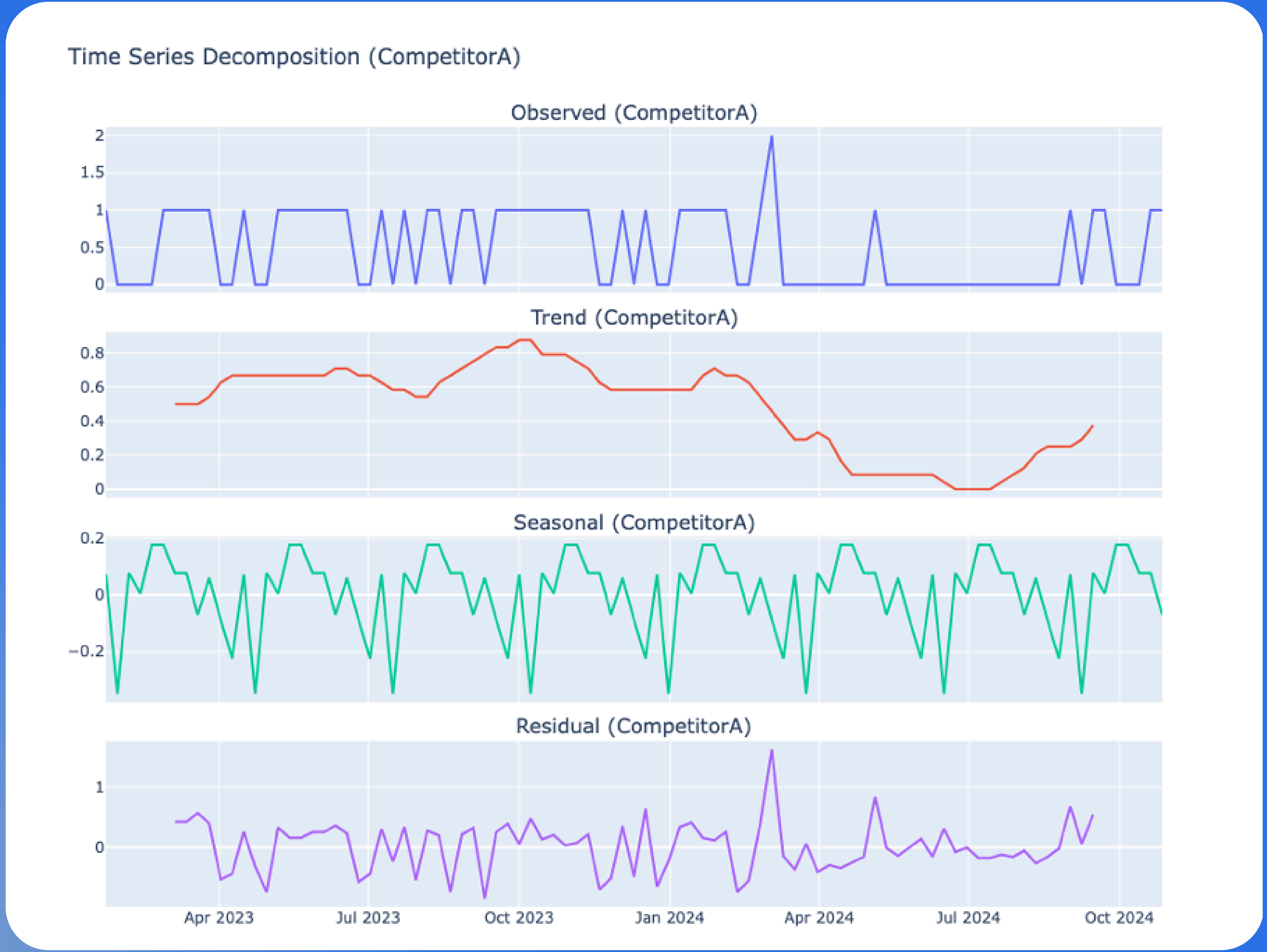


BUSINESS QUESTION ANALYSIS

AWKWARD
PROBLEM SOLUTIONS™

Is there seasonality in promotions?

- Competitor A shows a ~77-day promotional cycle, especially in 2023.
- Fewer campaigns in 2024, suggesting a strategic or budgetary shift.
- March 2024: Campaign A2 ran the entire month — an unusual extended duration.

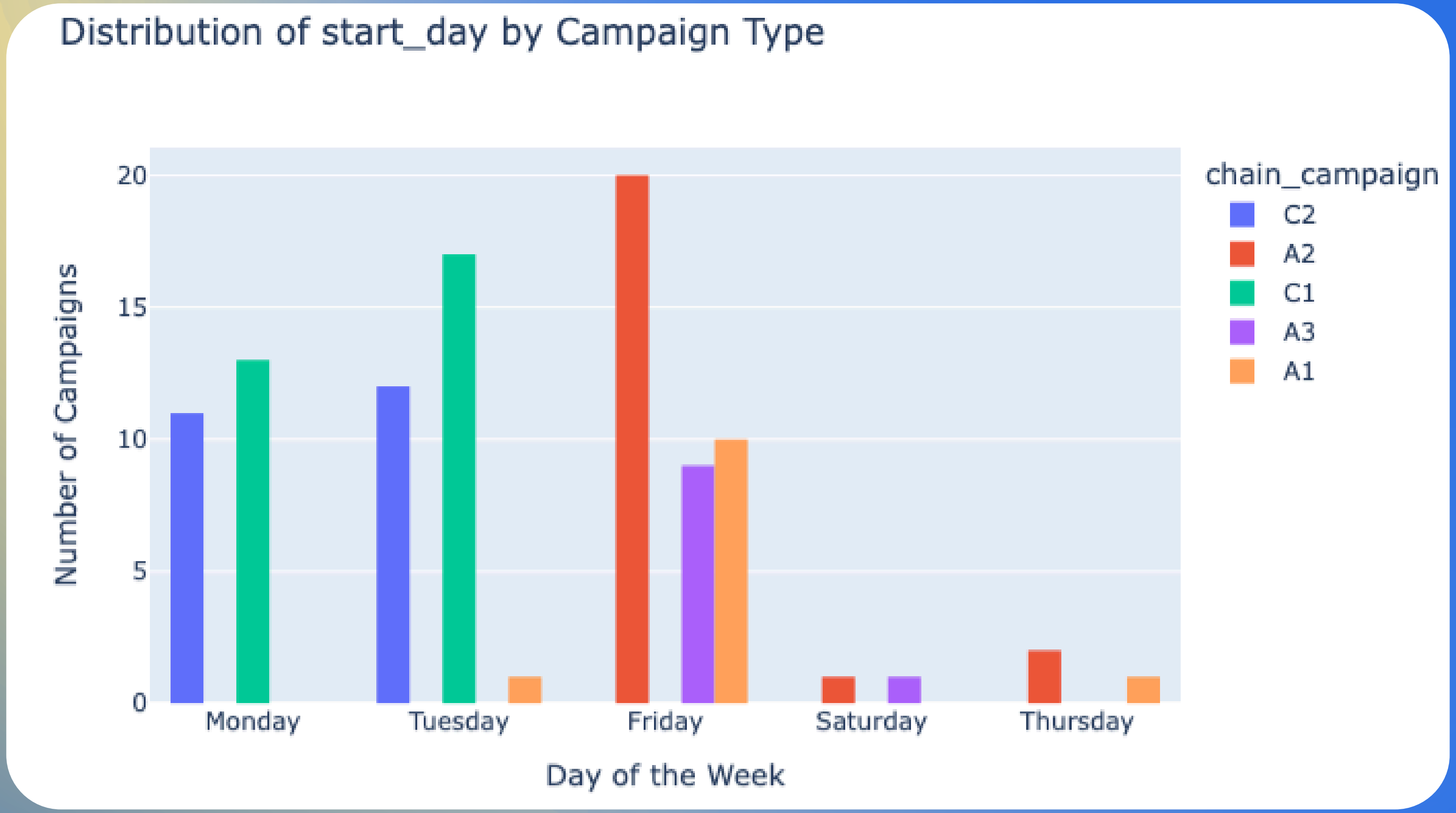


BUSINESS QUESTION ANALYSIS

AWKWARD
PROBLEM SOLUTIONS™

Is there seasonality in promotions?

- Chain maintains a steady promotion rhythm, often starting campaigns on weekdays.

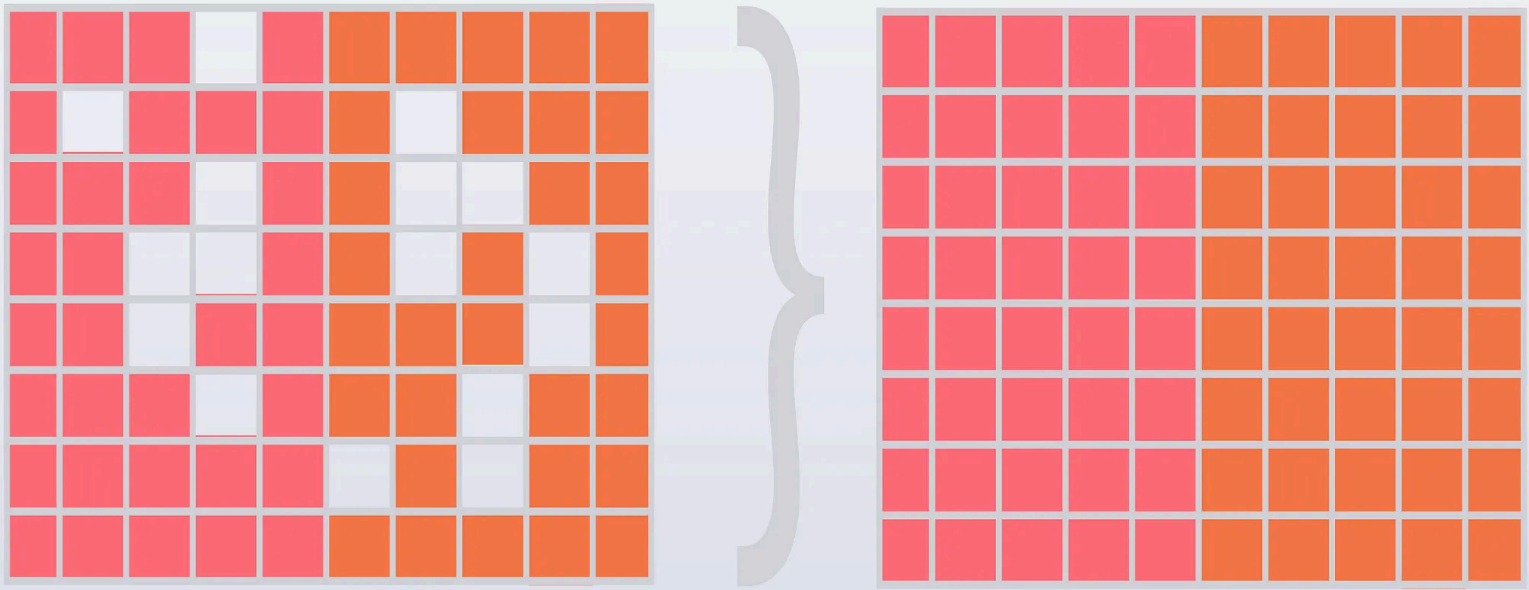


AWKWARD
PROBLEM SOLUTIONS™

MODELING

DATA PREPARATION

- Missing competitor prices (CompetitorA & B) across SKU-date pairs
- Supervised ML Imputation
  - Trained on records with available competitor prices
  - Key features:
    - Chain price (same SKU/date)
    - Temporal data (weekday, month, holidays)
    - Campaign indicators (promo flags, campaign type)
    - Seasonality flags



AWKWARD
PROBLEM SOLUTIONS™

MODEL SPECIFICATION

- Lag-based features:
  - Reflect historical price trends at SKU & category levels
  - Built with strictly causal windows (no future leakage)
- Chain price: Strong baseline predictor
- Temporal signals:
  - Includes seasonal flags: Christmas, Back-to-school, Summer, Black Friday

PIPELINE

- Pipeline steps:
  - Custom TimeFeaturesExtractor
  - SimpleImputer (mean strategy)
  - RandomForestRegressor for non-linear robustness
- Tuning via RandomizedSearchCV (3-fold):
  - n_estimators: 50–200
  - max_depth: 5–30
  - min_samples_split, min_samples_leaf: 2–10

Handling Lag Features in Real-Time Prediction

Lag-based features rely on historical data, but are not directly available in real-time API requests.

- To maintain model accuracy during inference, we implemented a method using pre-aggregated historical data:
  - For a given SKU and target date, the system searches historical data for the closest previous date with matching day and month.
  - If no exact match is found, it falls back to the most recent prior date.

- This approach fills lag-based features.

MODEL PERFORMANCE OVERVIEW

- Evaluation Metrics:
  - MAE = Average error in price units
  - sMAPE = Relative % error between predicted and actual prices

- Feature Engineering for Retraining:
  - 498 new records added to improve learning in weak segments
  - Lag-based features imputed using monthly SKU-level historical averages

Competitor	Model Version	MAE	sMAPE
Competitor A	Original	4.35	9.70%
Competitor A	Retrained	3.9	10.67%
Competitor B	Original	4.67	10.53%
Competitor B	Retrained	4.47	10.75%

PERFORMANCE INSIGHTS

- Segment-Level Improvements (structure_level_2):
  - Major improvements after retraining:
    - Competitor A – Structure 104: MAE ↓ from 4.79 → 2.15 | sMAPE ↓ 10.85% → 5.88%
    - Competitor B – Structure 104: MAE ↓ from 11.83 → 1.97 | sMAPE ↓ 24.83% → 5.25%

- Minor degradations:
  - Competitor B – Structure 101: MAE ↑ from 4.17 → 6.74 | sMAPE ↑ 8.20% → 13.49%

AWKWARD
PROBLEM SOLUTIONS™

DEPLOYMENT

DEPLOYMENT

AWKWARD
PROBLEM SOLUTIONS™

- Solution Architecture:
  - REST API built with Python (Flask).
  - SQLite + Peewee ORM for storing forecasts and actual prices.
  - Models are dynamically loaded on startup:
    - Model files hosted on Hugging Face Hub (GitHub size limitations).
- Containerization (Docker):
  - Reproducible and isolated environment for reliable deployment.

API FUNCIONALITY

- Available Endpoints:
  - POST /forecast_prices/
    - Input: sku (string), time_key (YYYYMMDD)
    - Validates format and avoids duplicates
    - Generates forecasts → stores in DB → returns response
  - POST /actual_prices/
    - Input: sku, time_key, pvp_is_competitorA_actual, pvp_is_competitorB_actual
    - Validates types and date format
    - Updates actual prices in the database

KNOWN ISSUES & RISKS

AWKWARD
PROBLEM SOLUTIONS™

- Input Validation Risks
  - Errors triggered by invalid sku or incorrectly formatted time_key
  - Defensive validation helps, but edge cases may persist
  -
- Model Drift
  - Models trained on static historical data
  - Market changes may reduce accuracy over time → requires retraining pipeline
- External Dependency
  - Dependency on Hugging Face for model files
  - Startup fails if model cannot be downloaded (e.g., network issues)

FUTURE IMPROVEMENTS

AWKWARD
PROBLEM SOLUTIONS™

- Automation & Accuracy
  - Implement automated retraining pipelines.
  - Add data quality alerts and stronger input validation.
- Infrastructure & Scalability
  - Move toward Docker-based orchestration for fault tolerance and performance under load.
- Smarter Forecasting
  - Integrate new features: seasonality, market events.
  - Explore time series for higher accuracy.

BORCEELLE

Thank You