Discount Optimization and Explainable Demand Forecasting using SHAP

This methodology and Python code used for discount simulation, demand forecasting using SARIMAX models, and explainability using SHAP. The project aims to help businesses simulate the effect of discounts on demand and optimize pricing strategies per product and region.

# 1. Forecasting with SARIMAX

SARIMAX models were built to forecast product-level demand using historical sales and exogenous variables like pricing and competitor pricing.

- Preprocessing and transforming prices to log scale (e.g., Log\_MRP, Log\_Competitor\_Price).

- Building SARIMAX model per (product, region) combination.

- Forecasting demand for 12 months into the future.

Output: metrics\_df (performance), detailed\_results (forecasts and elasticity).

# 2. Saving Results

A utility function `save\_results\_to\_csv()` is used to save:

- Forecasts  
- Elasticity coefficients  
- Actual vs. predicted comparisons  
- Model performance metrics

These are saved as CSV files with a customizable base filename.

# 3. Discount Simulation Engine

The `create\_price\_discount\_simulation()` function estimates how demand changes across discount levels:

- Split historical prices into 4 buckets

- For each bucket, calculate average demand

- Simulate predicted demand under 0-50% discounts using elasticity

Elasticity Formula:

new\_demand = demand \* np.exp(elasticity \* np.log(1 - discount/100))

# 4. Optimal Discount Identification

Function: calculate\_optimal\_discounts()

- Goal: Find discount that maximizes revenue per price bucket

- Output: Optimal discount with expected demand and revenue

Saved using: save\_optimal\_discounts()

# 5. Explainability with SHAP

Function: generate\_shap\_explanation()

Purpose: Quantify impact of Log\_MRP and Log\_Competitor\_Price on SARIMAX model predictions using SHAP.

- Trained SARIMAX model using Log\_MRP and Log\_Competitor\_Price

- Used shap.KernelExplainer with background sample from exog\_train

- Visualized mean SHAP values via bar plot

## Business Insights

- Log\_MRP and Log\_Competitor\_Price were key demand drivers.

- High SHAP values for Log\_Competitor\_Price indicate strong competitor influence.

- Localized SHAP patterns help customize pricing by geography.

## Assumptions and Limitations

- SARIMAX is assumed well-calibrated.  
- Exogenous data is clean and relevant.  
- KernelExplainer is computationally expensive.  
- SHAP may be unstable for low sample size (<30 rows).

# 6. Sample SHAP Invocation

shap\_df = generate\_shap\_explanation(  
 exog\_train=exog\_train,  
 exog\_test=exog\_test,  
 model\_results=model\_results,  
 exog\_vars=['Log\_MRP', 'Log\_Competitor\_Price'],  
 product='RMU(Ring Main Unit)',  
 geography='West'  
)

# Conclusion

The workflow enables data-driven pricing strategies by:  
- Forecasting demand  
- Simulating elasticity effects  
- Optimizing discounts  
- Explaining model behavior

# References

- SHAP documentation: https://shap.readthedocs.io/en/latest/  
- SARIMAX (Statsmodels): https://www.statsmodels.org/stable/generated/statsmodels.tsa.statespace.sarimax.SARIMAX.html  
- KernelExplainer: https://shap.readthedocs.io/en/latest/generated/shap.KernelExplainer.html