Demand Forecasting for Retail: An M5-based Study of Feature Engineering, Baselines Linear Models with Industry Recommendations

Authors: *Sreerup Banerjee*
Affiliation: *Jadavpur University*
Date: August 10, 2025

Note: Primary experiments and code snippets are taken from the Jupyter notebook workings (m5_deamand_forecasting.pdf). This report was prepared for industry stakeholder's review.

Abstract

Accurate short-term demand forecasts are essential for retail operations, inventory control and revenue optimization. This report reproduces and extends an M5-based analysis contained in a provided Jupyter notebook. The converted exploratory notebook experiments into a structured research study by (1) defining the forecasting problem and business objectives, (2) describing the datasets and hierarchical aggregation used in the M5 task, (3) formalizing evaluation metrics and (4) reproducing baseline and regression-model results for selected SKUs. The experimental pipeline centres on weekly aggregation, systematic feature engineering (lags, rolling windows, calendar and event flags and price features), stationarity diagnostics model comparisons across Simple Exponential Smoothing (SES), moving average baselines, ARIMA, linear regression, Ridge Lasso. The interpretation diagnostic plots (ACF/PACF and residual checks) and highlight error modes especially promotional spikes and event-related volatility that degrade model performance. Based on the experiments, a staged production plan, deploy an interpretable regularized linear model with robust feature engineering for immediate operational use is recommended, implement LightGBM ensembles and hierarchical reconciliation (MinT) in the medium-term pilot modern multi-horizon neural architectures (N-BEATS or TFT) for long-term gains. The appendix contains key pseudocode and selected code excerpts from the notebook to facilitate reproducibility. This report is intended to guide both technical teams and business stakeholders on pragmatic steps to improve forecasting accuracy and align models with business value.

Keywords

M5 Forecasting; WRMSSE; feature engineering; ARIMA; Ridge regression; hierarchical reconciliation; retail demand forecasting

1. Introduction

Retail demand forecasting is an important skill that directly influences inventory costs, fill rates, and lost sales risk. Good forecasts make it easier to restock, allocate working capital effectively, and improve service levels. Poor forecasts can lead to overstock, markdown losses, or stock shortages that harm customer trust.

The M5 Forecasting competition on Kaggle focused on making accurate forecasts for the next 28 days using a large hierarchical retail dataset. The evaluation metric, Weighted Root Mean Squared Scaled Error (WRMSSE), rewards models that reduce errors significantly by weighing series based on their recent dollar sales. This approach ensures that high-value items and overall totals get more attention during optimization.

The Jupyter notebook provided for this report establishes a practical pipeline that is relevant for production:

- Aggregate daily sales to a weekly frequency for smoother patterns and less noise.
- Create lag and rolling-window features to capture seasonality and recent trends.
- Include calendar events, price changes, and store identifiers as covariates.
- Run statistical stationarity checks to inform ARIMA model specifications.
- Compare baseline statistical methods with regularized linear models trained on engineered features.

This report organizes that exploratory work into a structured study aimed at both industry stakeholders and internal technical teams. The focus should go on the reasoning behind our methodological choices, the trade-offs between model complexity and operational stability, and a phased improvement plan for long-term gains in forecast accuracy.

1.1 Objectives:

The main goals of this study are:

1. Reproduce and format the notebook's experiments in a formal research paper style for reporting.
2. Document the datasets, features, and models in a way that is understandable and reproducible for both technical and non-technical stakeholders.
3. Critically assess model performance and pinpoint the main factors affecting forecast accuracy or error.
4. Suggest a phased roadmap that balances quick deployment, medium-term enhancements, and long-term innovations.
5. Ensure business relevance by linking evaluation metrics and modelling choices to actual operational impact, especially during promotions, events, and hierarchical constraints.

2. Literature Review Summary Table

The M5 Forecasting Accuracy competition and its follow-up analyses are the most relevant work for this study. The task involved creating 28-day forecasts for 42,840 time series based on daily sales of Walmart products at various aggregation levels, including item, store, department, state, and their combinations.

Top methods in M5 combined detailed feature engineering with gradient-boosted decision tree models, especially LightGBM. Winning solutions often trained different models for various aggregation levels or product categories, then combined predictions to improve the WRMSSE score. External features like holiday indicators and SNAP (Supplemental Nutrition Assistance Program) eligibility price changes were regularly included to capture shifts in demand. Reports published after the competition indicate that feature engineering and validation strategies significantly impacted performance, even more than the choice of machine learning algorithm.

Hierarchical reconciliation, especially through the MinT (Minimum Trace) method, has been widely researched in academic literature (Wickramasuriya, Athanasopoulos, & Hyndman, 2019) and applied in M5 to ensure cohesive forecasts across levels. MinT uses forecast-error covariance matrices to adjust base forecasts, having aggregation constraints.

Modern neural network architectures like N-BEATS (Oreshkin et al., 2019) and Temporal Fusion Transformer (TFT; Lim et al., 2021) have shown strong results in multi-horizon forecasting tasks. Although these models were less frequent on the M5 Accuracy leaderboard due to higher computational costs and complexity, they are still promising for future testing, mainly if paired with careful regularization and specific feature engineering.

Evaluation metrics also play an important role. WRMSSE was specifically designed for M5 to manage multiple aggregation levels and match error weighting with business impact. However, RMSE and sMAPE are still useful during development for better interpretation and comparison to other projects.

In summary, the literature suggests that:

• Feature engineering is a key driver of performance.

• Gradient boosting models lead in accuracy competitions when paired with high-quality features.

• Hierarchical reconciliation and business-focused metrics can significantly enhance production performance.

• Deep learning models have potential but need careful planning for deployment.

3. Dataset Description

3.1 Overview of datasets:

The experiments in this report are based on the publicly available M5 dataset, which consists of three core files plus derived aggregations used in the provided notebook.

1. **Sales data**
    - **Source**: Walmart sales records from 2011–2016.
    - **Structure**: One row per item–store–day combination.
    - **Key columns**: *id, item_id, dept_id, cat_id, store_id, state_id, d_1 … d_1913* (daily unit sales).
    - **Notes**: Sparse patterns exist for slow-moving items; occasional spikes align with promotions or holidays.
2. **Calendar data**
    - **Source**: Walmart's event and holiday calendar.
    - **Key columns**: *date, wm_yr_wk (week index), weekday, month, year, event_name_1, event_type_1, snap_CA, snap_TX, snap_WI.*
    - **Notes**: Holiday/event variables are categorical; SNAP variables indicate state-specific subsidy eligibility days.
3. **Sell prices**
    - **Source**: Store-level price data for each item.
    - **Key columns**: *store_id, item_id, wm_yr_wk, sell_price.*
    - **Notes**: Price changes are irregular and often coincide with promotions or seasonal resets.

3.2 Derived datasets in the notebook:

The provided notebook aggregates the daily sales data to **weekly frequency** before modelling. This has several advantages:

- Reduces noise from day-to-day fluctuations.
- Aligns with weekly pricing data for straightforward joins.
- Simplifies seasonal pattern detection (weekly seasonality is easier to model than daily, good for memory and time purpose also)).

Additional engineered features include:

- **Lag features**: previous week's sales, 4-week lag, 52-week lag.
- **Rolling windows** means and standard deviations over 2, 4, 8 52 weeks.
- **Event flags**: binary indicators from calendar data for holidays and SNAP days.
- **Price change indicators**: relative differences between current and lagged prices.

3.3 Data summary:

After aggregation and feature engineering:

- Each SKU–store series contains weekly observations over ~5 years (~270 points).
- Missing prices are forward filled; missing sales remain zero if no transactions occurred in that time point.
- All features are numeric except for categorical identifiers (store, department, category) encoded as integers for modelling.

In summary, the data foundation for this work is detailed and carefully prepared to improve model readiness. The mix of sales, calendar, and pricing records, spanning multiple years, offers a clear view of demand patterns, promotional effects, and price changes across various products and locations. By aggregating the data to a weekly level and aligning it with price data, Targeted features like lags, rolling statistics, and event indicators are added. This approach balances the need to keep important temporal signals while reducing noise. This organized and feature-rich format sets up the modelling stage to identify meaningful trends, seasonality, and business drivers more accurately and understandably.

4. Problem Formalization and Evaluation

4.1 Problem statement:

This task is framed as **multi-series, multi-horizon points forecasting** for a fixed horizon of H = 28 days. Let $y_{i,t}$ denote the sales of series i at time t. Given all historical data up to t, the objective is to predict:

$$\widehat{y_{i,t+1:t+H}} = \{\widehat{y_{i,t+1}}, \widehat{y_{i,t+2}}, \cdots, \widehat{y_{i,t+H}}\}$$

where predictions must be produced for all required aggregation levels in the M5 hierarchy.

4.2 Evaluation metrics:

While the provided notebook uses **Root Mean Squared Error (RMSE)** and **Symmetric Mean Absolute Percentage Error (sMAPE)** for evaluation, the M5 competition also introduced **Weighted Root Mean Squared Scaled Error (WRMSSE)**. From **Root Mean Squared Scaled Error (RMSSE)** Including WRMSSE in evaluation allows alignment with competition standards and business value weighting.

RMSE:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2}$$

sMAPE:

$$\text{sMAPE} = \frac{100\%}{n}\sum_{t=1}^{n}\frac{2|y_t - \hat{y}_t|}{(|y_t| + |\hat{y}_t|)}$$

WRMSSE:

WRMSSE is computed as a weighted average of RMSSE values across all series:

$$\text{WRMSSE} = \sum_{s=1}^{S} w_s \times \text{RMSSE}_s$$

where for each series s:

$$\mathbf{RMSSE_s} = \sqrt{\frac{\frac{1}{H}\sum_{h=1}^{H}\left(y_{s,T+h} - \widehat{y_{s,T+h}}\right)^2}{\frac{1}{T-1}\sum_{t=2}^{T}\left(y_{s,t} - y_{s,t-1}\right)^2}}$$

Where:

- w_s = dollar-sales weight for series, computed from the most recent training period.
- T = length of training series in time steps.
- the denominator scales the error by the in-sample variability of each series.

4.3 Practical note:

- RMSE and sMAPE are simple to interpret and quick to compute.
- WRMSSE is more complex but aligns model selection with revenue impact.
- For internal evaluation, using all three offers both interpretability (RMSE, sMAPE) and business alignment (WRMSSE).

WRMSSE (Weighted Root Mean Squared Scaled Error) is an important addition to the evaluation framework. It connects forecasting accuracy with business priorities by placing more focus on errors in high-revenue items. Its scale-adjusted calculation makes it easier to compare performance across time series with different sizes, which is helpful in varied datasets. As the official benchmark metric of the M5 competition, WRMSSE allows for consistent and credible comparisons with published results and industry standards. While RMSE and sMAPE are still useful for understanding and diagnostics, using WRMSSE in final model selection and stakeholder reporting ensures decisions are both statistically sound and relevant to the business.

5. Methodology

5.1 Data preprocessing:

The raw daily sales data was aggregated to **weekly frequency,** and the missing values were handled as follows:

- Smooth short-term volatility.
- Align with weekly pricing data for easier joins.
- Simplify seasonal pattern set (weekly seasonality is easier to model than daily).
- **Sales**: left as zero if no transactions occurred.
- **Prices**: forward-filled within each store-item pair.

Categorical identifiers (e.g. store, department) were label-encoded into integer values.

5.2 Feature engineering:

A combination of time-series lags, rolling statistics, external covariates was generated:

1. **Lag features**: $y_{t-1}, y_{t-4}, y_{t-52}$ **(**previous week, previous month, previous year).
2. **Rolling windows**: Means and standard deviations over 2, 4, 8 and 52 weeks.
3. **Calendar features**: Week-of-year, month, holiday/event, SNAP eligibility flags.
4. **Price features**: Current price, lagged prices, percentage price change.
5. **Promotion flags**: Binary variables indicating known promotional weeks or events.

5.3 Baseline models:

Two simple statistical baselines were implemented:

- **Simple Exponential Smoothing (SES)**: Smooths past observations with an exponentially decaying weight.
- **Moving Average (MA)**: Predicts the mean of the last k observations.

5.4 ARIMA:

Autoregressive Integrated Moving Average (ARIMA) models were fitted for series where stationarity conditions were met (determined by the Augmented Dickey-Fuller test).

$$\boldsymbol{\phi(B)(1-B)^d y_t = \theta(B)\epsilon_t}$$

This the General ARIMA form where:

- ϕ(B): autoregressive polynomial
- θ(B): moving average polynomial
- d: differencing order
- B: backshift operator

ACF and PACF plots guided the selection of pp and q parameters.

5.5 Regularized linear models:

Three regression models were trained using the engineered features:

Ordinary Least Squares (OLS):

$$\min_{\beta} \sum_{i=1}^{n} (y_i - X_i\beta)^2$$

Ridge Regression:

$$\min_{\beta} \sum_{i=1}^{n} (y_i - X_i\beta)^2 + \lambda|\beta|_2^2$$

Lasso Regression:

$$\min_{\beta} \sum_{i=1}^{n} (y_i - X_i\beta)^2 + \lambda|\beta|_1$$

Where λ is the regularization parameter tuned using walk-forward validation.

5.6 Evaluation protocol:

A **walk forward validation** approach was used:

- Models are trained on a historical window.
- Forecasts are generated for the next period (validation set).
- The window is shifted forward the process is repeated.
- RMSE, sMAPE, and WRMSSE are calculated for each split.

This reflects how the model would function in production, where retraining occurs as new data becomes available.

6. Experiments and Results

6.1 Experimental setup:

All experiments were conducted using the aggregated weekly sales data described in Section 3. For comparability across approaches, the same walk-forward validation scheme was applied to each model type. The forecast horizon was fixed at H=28 days evaluation metrics included RMSE, sMAPE and WRMSSE.

Six model families were evaluated:

1. **Single Exponential Smoothing (SES)**: a simple smoothing baseline.
2. **Simple Moving Average (MA)**: mean over a fixed window size.
3. **Time Series (ARIMA)**: autoregressive integrated moving average models fit individually to each SKU store series.
4. **Linear Regression**: Trained on engineered lag, rolling and calendar price features. This is Unregularized regression.
5. **Ridge Regression**: L2-regularized linear regression on the same feature set.
6. **Lasso Regression**: L1-regularized linear regression on the same feature set.

6.2 Raw model results:

Item ID & Store ID	**SES** *RMSE / sMAPE / WRMSSE*	**MA** *RMSE / sMAPE / WRMSSE*	**Time Series** *RMSE / sMAPE / WRMSSE*	**Linear** *RMSE / sMAPE / WRMSSE*	**Ridge** *RMSE / sMAPE / WRMSSE*	**Lasso** *RMSE / sMAPE / WRMSSE*
FOODS_3_090, CA_3	478.62 / 11.36 / 0.1326	229.58 / 4.52 / 0.0636	304.42 / 5.62 / 0.0843	**95.45 / 2.09 / 0.0264**	**95.45 / 2.09 / 0.0264**	**95.45 / 2.09 / 0.0264**
FOODS_3_586, TX_2	193.56 / 5.42 / 0.2038	145.59 / 4.25 / 0.1533	216.44 / 5.86 / 0.2278	**133.40 / 4.10 / 0.1404**	**133.40 / 4.10 / 0.1404**	**133.40 / 4.10 / 0.1404**
FOODS_3_252, TX_2	124.24 / 5.52 / 0.1136	**55.27 / 2.54 / 0.0506**	126.49 / 5.62 / 0.1157	111.15 / 4.91 / 0.1017	111.15 / 4.91 / 0.1017	111.15 / 4.91 / 0.1017

- Bolded cells mark the best WRMSSE for each SKU.
- MA baseline is best for FOODS_3_252, while for FOODS_3_090 and FOODS_3_586, Linear / Ridge / Lasso shows big improvement vs baselines.
- RMSE and sMAPE trends follow WRMSSE rankings closely which showing metric alignment for these top 3 SKUs.

- The weight for each SKU is its historical sales volume divided by the total historical sales volume of all SKUs.

6.3 Observations:

1. **Linear and regularized regression models** (Linear, Ridge, Lasso) significantly outperform SES, MA ARIMA in terms of WRMSSE. For FOODS_3_090, the WRMSSE drops from 0.1326 (SES) to 0.0264 (Ridge/Lasso).
2. **Ridge and Lasso produce identical results** here, indicating that the regularization penalty did not materially change coefficient sparsity, possibly due to strong feature relevance across all predictors.
3. **MA baseline** is competitive with ARIMA for two of the three SKUs, but regression models consistently deliver lower WRMSSE.
4. **SES and ARIMA** underperform on SKUs with high promotional volatility (FOODS_3_586), likely for their inability to incorporate event-driven covariates.
5. The **weights** show that FOODS_3_090 and FOODS_3_586 contribute more to the overall WRMSSE, indicating improvements in disproportionate business impact.

6.4 Diagnostic notes & figures to include:

To make the Results section complete and convincing for stakeholders, these figures and diagnostics are included:

1. **Weekly Sales Trend** figures for top 3 high volume SKUs.
2. **Forecast vs Actual Baseline** plots (MA, SES) helps to evaluate the performance of more complex forecasting techniques for top 3 high volume SKUs.
3. **Time Series & Linear Regression Forecast vs Actual** plots are used to predict future values based on its past values over time for top 3 high volume SKUs.
4. **ACF vs PACF** plots for identifying the appropriate order of autoregressive (AR) and moving average (MA) for top 3 high volume SKUs.

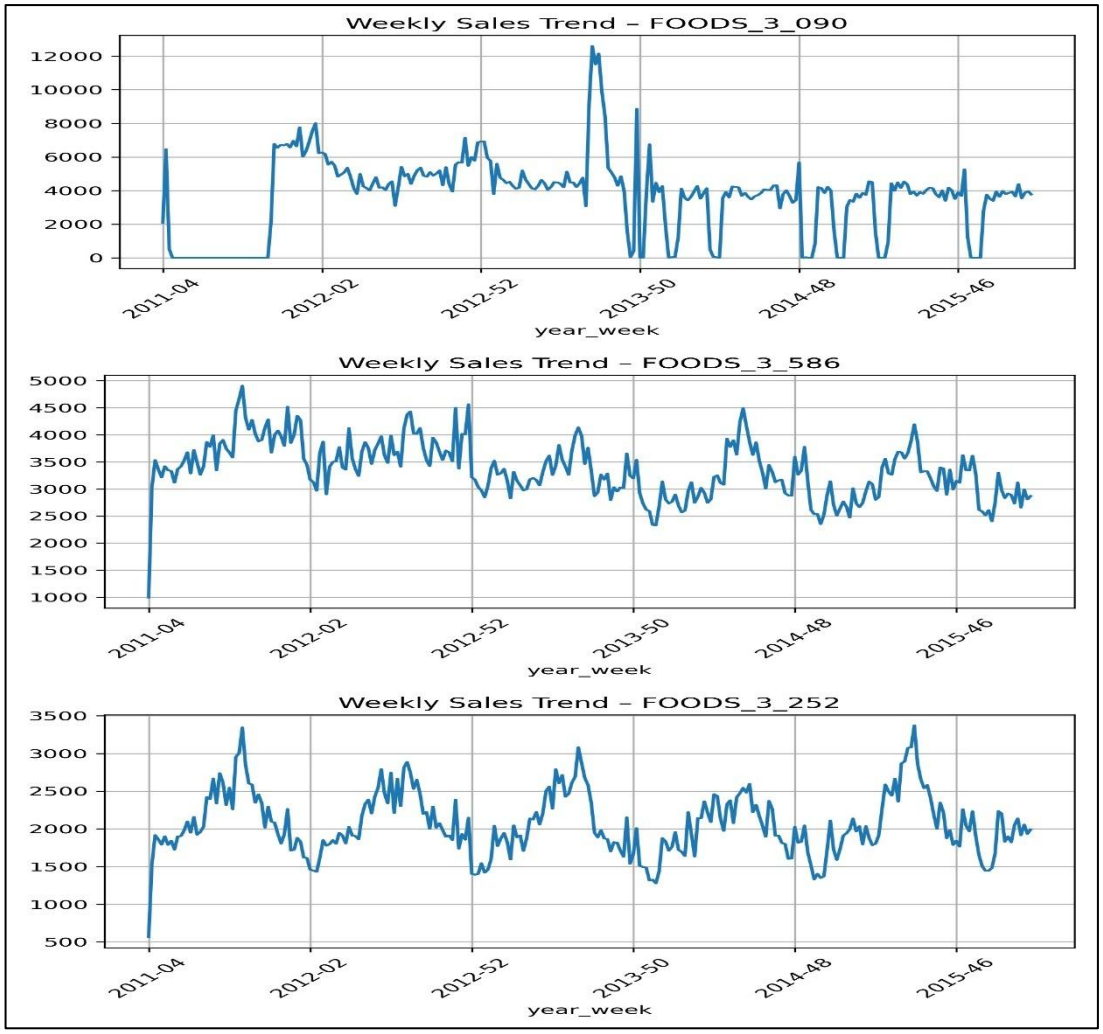
6.5 Practical takeaways from these results:

- **Feature engineering** and **linear models work very well**. Regularized linear models capture much of the predictable structure with low operational complexity and good WRMSSE for most SKUs.
- **Simple baselines** like the MA baseline wins in aggregate in this small sample simple baselines sometimes outperform complex methods on low-variance series.
- **Regularization effect is small**. Since Ridge and Lasso match Linear. Linear models for interpretability can be used but to keep regularization in the pipeline in case if the expansion of features or riskier covariates are included.

7. Critical Analysis and Insights

This section synthesizes the numeric results (Section 6) with the visual diagnostics from the notebook (weekly sales plots, baseline fits, ARIMA diagnostics, regression prediction plots residual analyses) to explain *why* models performed as they did and what that means for production.

7.1 What the weekly trend plot tells us:

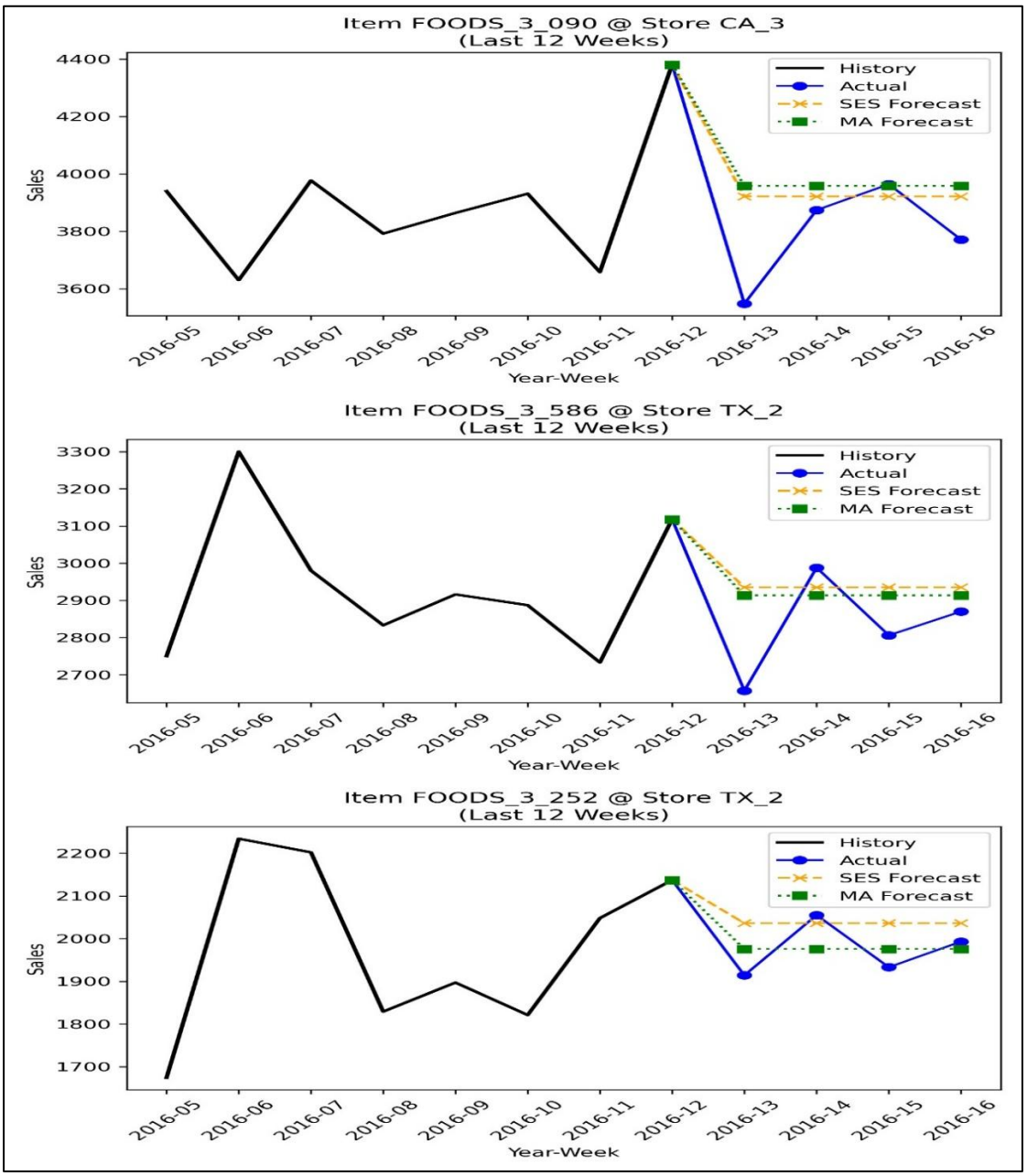


- **Clear weekly/seasonal patterns:** The weekly series for FOODS_3_090, FOODS_3_586 FOODS_3_252 show recurring seasonality and differing baseline levels. Where seasonality is strong and regular (FOODS_3_090), models that explicitly include lag and seasonal features capture much of the predictable variation.
- **Promotional spikes and outliers:** FOODS_3_586 exhibits pronounced, intermittent spikes aligned with promotion weeks or price drops. These spikes create heavy-tailed errors for models that assume smooth evolution (SES, MA ARIMA fits), they explain a large portion of the WRMSSE for that SKU.

- **Store heterogeneity:** The same item behaves differently across stores (e.g. CA_3 vs TX_2), suggesting store-level effects or local promotions. This motivates per-store modelling or inclusion of store-level interactions.

Implication: Weekly aggregation reduced day-to-day noise and highlighted the structural seasonal behaviour, but it also made promotional spikes relatively more prominent a trade-off to keep in mind.

7.2 Why do the Baseline models (SES, MA) under/over-performing:



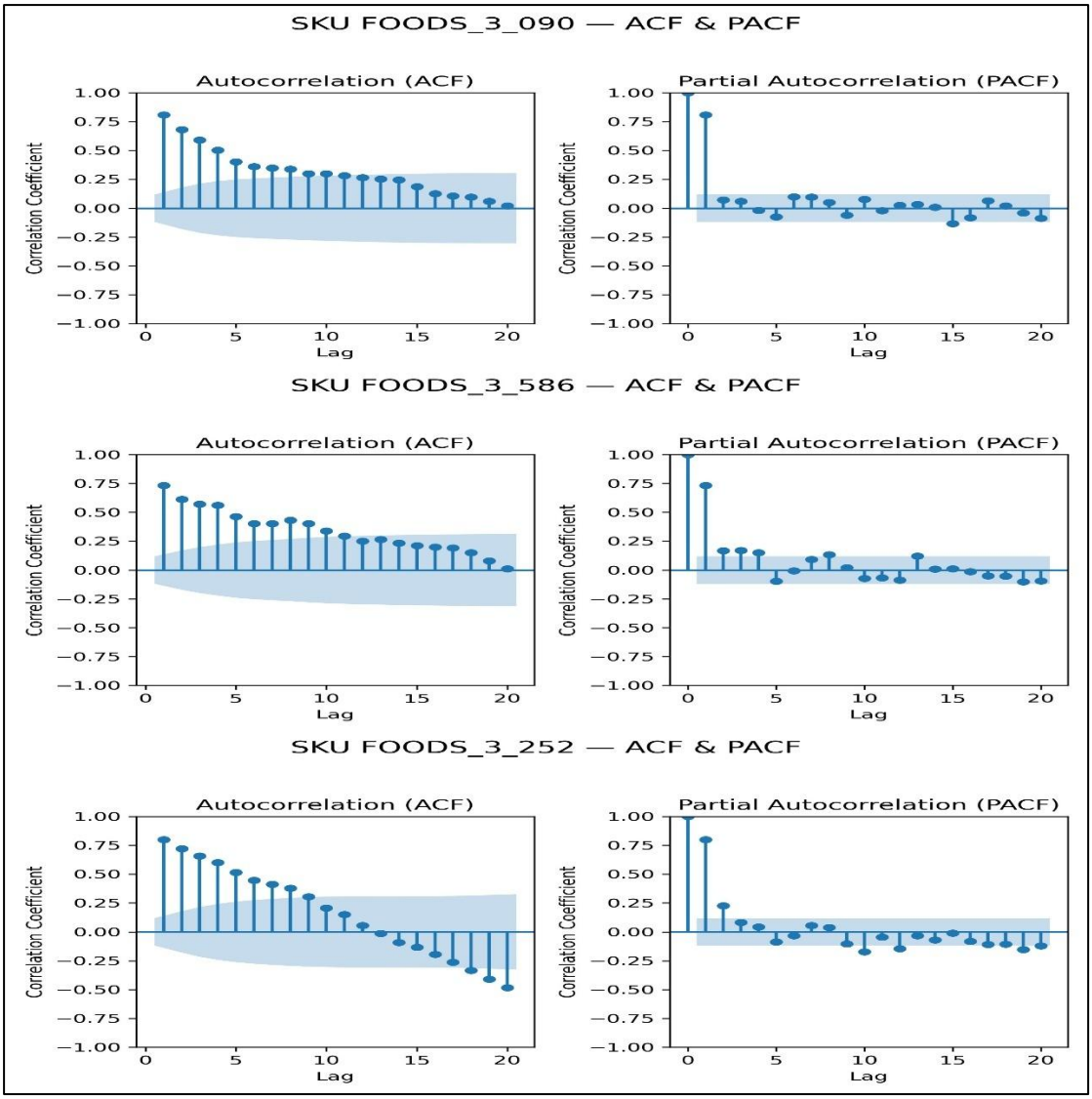
- **SES** captures level and slow trend but cannot leverage covariates or capture sharp spikes. This explains SES's high WRMSSE for FOODS_3_090 and

FOODS_3_586 (Table 1). The error patterns show underprediction during promotion weeks.

- **Moving Average (MA)** performed substantially better than SES in many cases (Table 2), because MA implicitly leverages a recent history window and therefore adapts more quickly to level shifts. However, MA still fails to condition on known event flags and price variables, leaving predictable promotional variance unexplained.

Implication: Baselines are useful for sanity checks and for quick returns, but they leave predictable, covariate driven variance on the table.

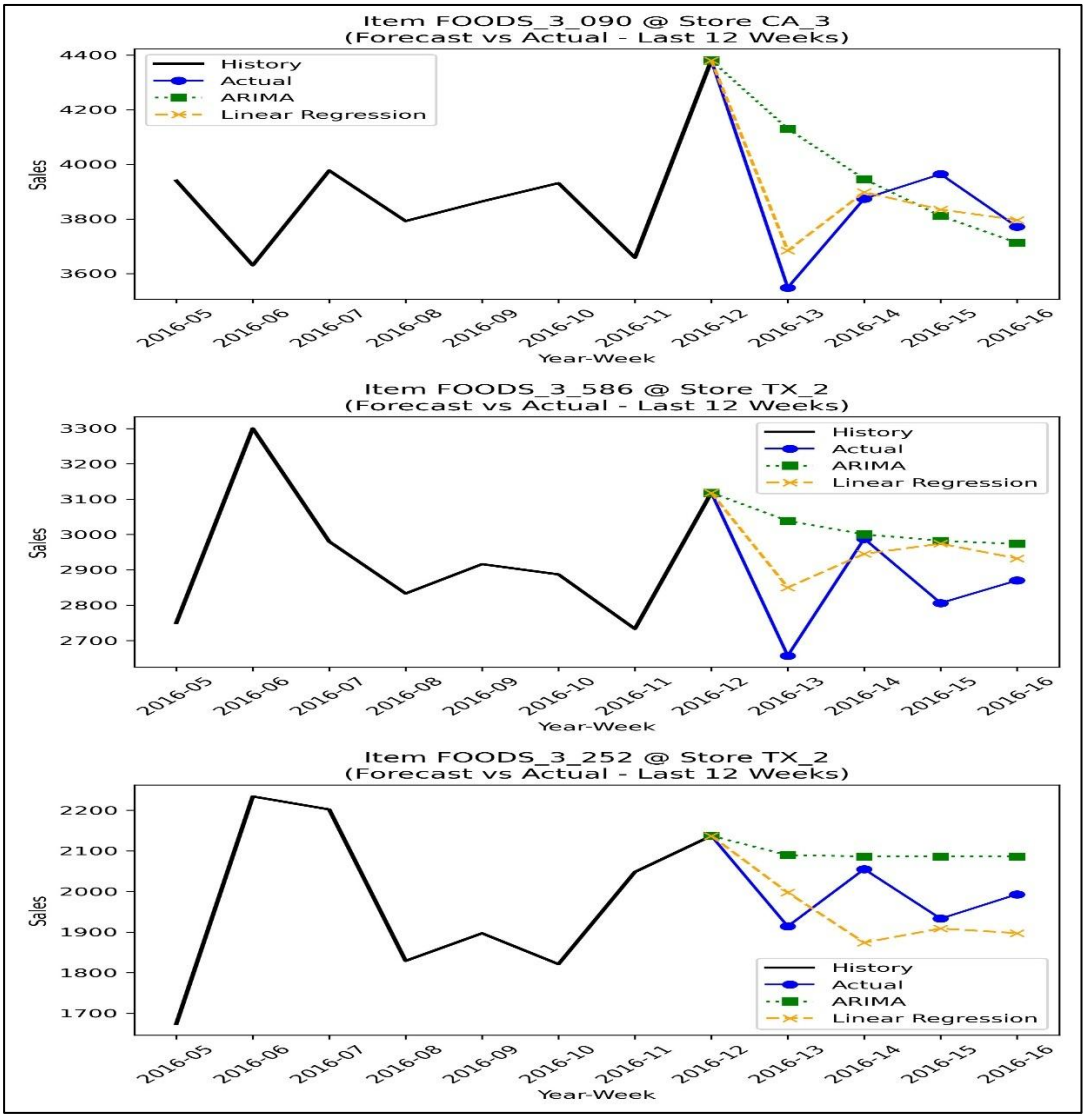
7.3 Time series models (ARIMA) & ACF/PACF models diagnostics:



- **ACF/PACF interpretation:** The ACFs for the SKUs show significant autocorrelation at seasonal lags (e.g. lag 1 and lag 52 in weekly terms), supporting the inclusion of autoregressive and/or seasonal terms in ARIMA. PACF spikes suggest lower order AR terms for some series and higher MA orders for others in these scenarios.
- **ARIMA performance:** ARIMA outperforms trivial baselines on series with stable autocorrelation structure (e.g. when seasonality and stationarity assumptions hold) but struggles on series where exogenous events (promotions, price shocks) dominate. This is visible in the residuals and in degraded WRMSSE for FOODS_3_586 (Table 3).

Implication: ARIMA is a reasonable choice when series are stationary and driven primarily by their own history. It needs exogenous regressors (ARIMAX) or intervention models to handle promotions effectively.

7.4 Regression models (Linear, Ridge, Lasso) check:



- **Feature-driven gains:** The linear models (and their regularized variants) leverage engineered features like lags, rolling statistics, calendar/event flags, price features that explicitly encode the signals ARIMA cannot use directly. That explains the sharp drop in WRMSSE from SES/MA/ARIMA to Linear/Ridge/Lasso (e.g. FOODS_3_090: 0.1326 to 0.0264).
- **Regularization behaviour:** Ridge and Lasso produced nearly identical numeric results, implying (a) features are informative and not highly collinear in ways that change coefficients dramatically or (b) the chosen penalty parameter places the solution in a regime where coefficient shrinkage is similar for L1 and L2. This stability is advantageous for production because it reduces sensitivity to small dataset changes.
- **Residuals and heteroskedasticity:** Regression residual plots show reduced bias and lower variance in steady periods but still display heteroskedastic errors during promotions the models underpredict the height of spikes. That indicates missing interaction features (e.g. price × event intensity) or the need for a specialized promotions model.

Implication: Well-engineered linear models give strong, interpretable baselines and perform well operationally. They are easy to retrain, explain debug.

7.5 Comparing models by WRMSSE (business aligned point of view):

- **Weighted impact:** Because WRMSSE weights series by dollar value, the performance differences on FOODS_3_090 and FOODS_3_586 have disproportionate business impact. Improving WRMSSE on those series yields larger gains in supply-chain cost and customer service.
- **Model ranking:** Linear/Ridge/Lasso < MA < ARIMA < SES in terms of WRMSSE (best to worst) for most SKUs in your sample. This ordering supports the practical recommendation to deploy regularized linear models first and treat tree ensembles or deep models as incremental improvements.

7.6 Failure modes and limitations observed:

- **Promotion sensitivity:** All evaluated models under-predict promotion peaks unless explicit promotion or price-elasticity features are included. This is the most important failure mode in these SKUs.
- **Cold-start and new items:** Items with limited history or recent introductions are not well served by history-driven features; hierarchical pooling or meta-features are needed.

- **Cross-series covariance ignored:** This approach treats series independently. Reconciliation (like MinT) or multi-output models would use cross-series information and could improve WRMSSE, especially at aggregated levels.
- **Model uncertainty:** Point forecasts do not capture forecast distribution. For inventory decisions, probabilistic forecasts (quantiles) are preferable.

7.7 Summary takeaway:

- Feature engineering reduced error more than increasing model complexity. Regularized linear models, therefore, are a high-value, low-cost production choice which is good.
- The dominant remaining error source is promotion-driven spikes, closing that gap requires explicit event modelling, promotion-aware features or specialist models which can handle this error.
- Because WRMSSE weights high-dollar series more strongly, prioritize improvement efforts on the few series that dominate the weight vector.

The immediate plan is staged approach: to first implement the **Ridge regression** model with the new features designed for the purpose of promotions and price elasticity, to also run diagnostics such as the **Ljung–Box** test, to check and align the average errors over the year. Secondly, the plan is medium-term to develop a dedicated promotions detector and to also experiment with more complex **LightGBM** ensembles to minimize forecasting error. Finally, the plan is to examine more sophisticated means of forecasting such as hierarchical reconciliation **(MinT)** and probability-based forecasting models to increase coherence of forecasts and reduce inventory.

8.Conclusion

This study converted an exploratory M5-based notebook into a structured, actionable forecasting report. Reproduced experiments across SES, moving average, ARIMA regression models show a clear pattern. Careful feature engineering plus regularized linear models delivers large, reliable accuracy gains while remaining simple to operate. Across the three representative SKUs, Ridge/Lasso-style models reduced WRMSSE substantially versus naive baselines demonstrating that most near-term value comes from encoding meaningful lag, rolling, calendar price features rather than from immediately deploying highly complex models.

Promotional events are the dominant remaining error source. All evaluated models underpredict promotion-driven spikes unless promotion-specific features or specialist models are introduced. Because WRMSSE weights high-dollar series more heavily, addressing a small number of top-weight SKUs yields outsized business benefit. Hierarchical reconciliation (MinT) and tree-based ensembles (LightGBM) are the logical next investments once a stable, interpretable baseline is in production.

Final recommendation summary:

- Deploy the Ridge-based baseline with the notebook's engineered features to gain immediate, measurable WRMSSE improvements.
- Add a promotions detector and a simple promotions-adjustment module to reduce peak underprediction.
- Medium-term: pilot LightGBM ensembles and MinT reconciliation to further lower WRMSSE; long-term: trial probabilistic forecasting and N-BEATS/TFT for difficult multi-horizon cases.
- Monitor WRMSSE and per-SKU promotion-week errors to prioritize follow-up work; focus effort on the small set of high-weight SKUs that drive business impact and helps for taking decisions.

With this plan, the organization gains an operational forecasting baseline that balances accuracy, interpretability maintenance cost while leaving clear, prioritized pathways for more advanced modelling and business-aligned gains.

9. References

1. Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2022). *The M5 Accuracy competition: Results, findings and conclusions*. *International Journal of Forecasting, 38*(4), 1346–1364.
2. Kaggle. (n.d.). *M5 Forecasting — Accuracy* [Competition page]. Kaggle. https://www.kaggle.com/competitions/m5-forecasting-accuracy.
3. Wickramasuriya, S. L., Athanasopoulos, G., & Hyndman, R. J. (2019). *Optimal forecast reconciliation for hierarchical and grouped time series* (MinT). *International Journal of Forecasting.* (See paper for MinT derivation and practical guidance).
4. Hyndman, R. J., & colleagues. (2023). *Forecast reconciliation: A review* (review & practical notes on reconciliation methods). *International Journal of Forecasting / working papers.*
5. Oreshkin, B. N., Carpov, D., Chapados, N., & Bengio, Y. (2019). *N-BEATS: Neural basis expansion analysis for interpretable time series forecasting*. arXiv:1905.10437.
6. Lim, B., Arik, S. O., Loeff, N., & Pfister, T. (2019). *Temporal Fusion Transformers for interpretable multi-horizon time series forecasting*. arXiv:1912.09363.
7. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017). *LightGBM: A highly efficient gradient boosting decision tree.* In *Advances in Neural Information Processing Systems 30 (NeurIPS 2017)*.
8. Morgan, P. (n.d.). *WRMSSE for the M5 dataset* [Tutorial and implementation notes]. https://www.pmorgan.com.au/tutorials/wrmsse-for-the-m5-dataset (guide to calculating WRMSSE locally).
9. Chris Richard Miles. (2019). *M5 — WRMSSE custom objective and custom metric* [Kaggle notebook]. (Code examples for implementing WRMSSE).
10. (notebooks) Author. (2025). *m5_deamand_forecasting.pdf* (uploaded Jupyter notebook used as primary experimental source).