Evaluating Time-Series and Machine Learning Models for Sales Forecasting

Alexander Coles, Tanmayee Kolli,
Simran Mallik, Chris Park,
Vaishnavi Vuyyuru

December 2nd, 2025

Group 14

TABLE OF CONTENTS

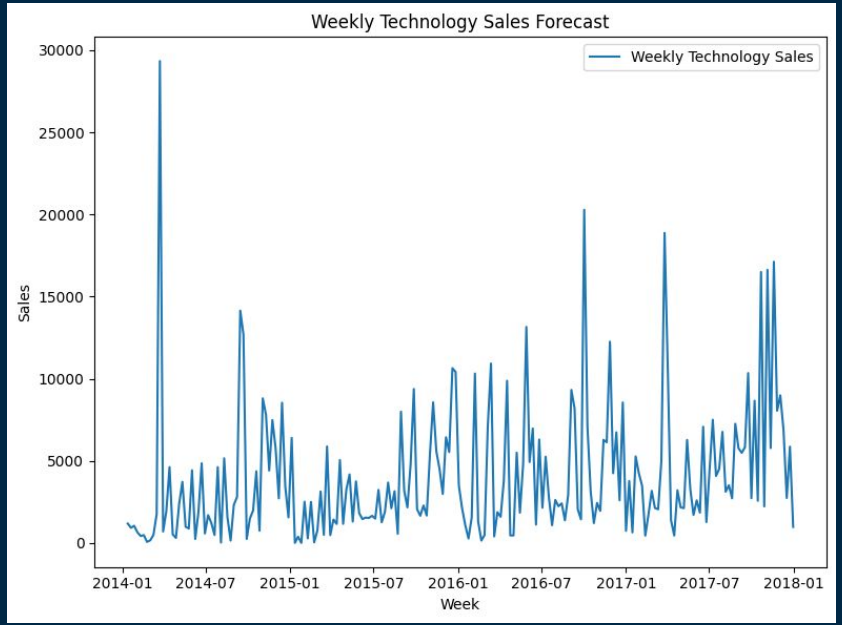
MOTIVATION & PROBLEM STATEMENT

01

REAL-WORLD MOTIVATION

Why sales forecasting matters in retail:

- Retailers rely on accurate weekly revenue forecasts for inventory, pricing, promotions, operational strategy, resource allocation, budgeting
- Unpredictable sales peaks cause forecast failures
- Poor forecasts → stockouts, markdown loss, over-ordering, budgeting errors, etc.



INDUSTRY CONNECTION

How we chose to frame our thoughts:

- We were hired as an analytics consulting team for a large, unspecified superstore
- They want us to assess if machine learning or deep learning methods can outperform classical forecasting models
    - Overall, they want us to recommend the **best forecasting model** for their business planning

PROBLEM STATEMENT & OBJECTIVE

Research Question:

How do time-series models (ARIMA, SARIMA, Prophet) compare to machine and deep learning models (Random Forest, XGBoost, LSTM) in forecasting weekly sales revenue?

Objectives:

- Compare classical vs ML/DL model types
- Evaluate feature strategies (statistical feature engineering, lags-only)
- Identify best forecasting model based on chosen metrics and interpretability
- Understand effective techniques and trade-offs

Hypothesis:

- We hypothesize that XGBoost will outperform statistical models because it can incorporate promotions, discounts, and seasonal behavior to better predict real retail demand fluctuations.

02

DATA: SOURCE, CHALLENGE, & PREPARATION

DATASET OVERVIEW

- **Source**: Kaggle
- **Format**: CSV
- **Shape**: (9994, 21)
- **Features**: ["Row ID", "Order ID", "Order Date", "Ship Date", "Ship Mode", "Customer ID", "Customer Name", "Segment", "Country", "City", "State", "Postal Code", "Region", "Product ID", "Category", "Sub-Category", "Product Name", "Sales", "Quantity", "Discount", "Profit"]
- **Example Row:**

Row ID	Order ID	Order Date	Ship Date	Ship Mode	Customer ID	Customer Name	Segment	Country	City	State	Postal Code	Region	Product ID	Category	Sub-Category
1	CA-2016-152156	2016-11-08	11/11/2016	Second Class	CG-12520	Jane Doe	Consumer	United States	Henderson	Kentucky	42420	South	FUR-BO-10001798	Furniture	Bookcases

Product Name	Sales	Quantity	Discount	Profit
Bush Somerset Collection Bookcase	261.96	2	0	41.9136

KEY FEATURES

Sales & Profit:
Tracks revenue and profitability across dimensions.

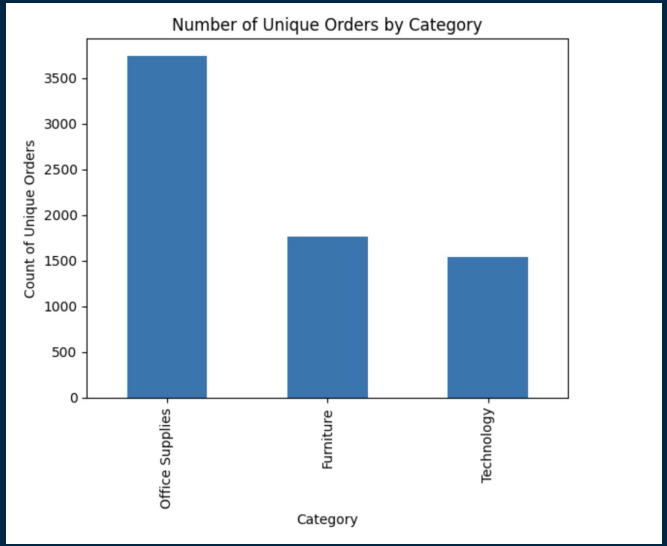
Product & Category:
Analyzes performance by product & category levels.

Geographic Insights:
Enables analysis by Country, Region, State, City.

Customer & Segment:
Provides insights into customer behavior and segments.

DATA CHALLENGES / PREPARATION

- Data contains orders strictly in the US, spanning 48 states across the North, West, South, and East Regions
- Data ranges from 01/12/2014 - 12/31/2017.
- No missing values or observed outliers
- For all models, we aggregated data by week to establish a consistent baseline.
- The modeling was intentionally subsetted due to the data imbalance observed across different product categories.

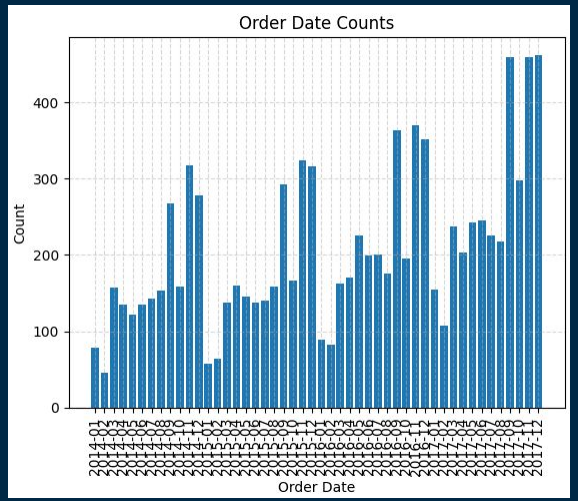


EXPLORATION, MODELING, & METHODOLOGY

03

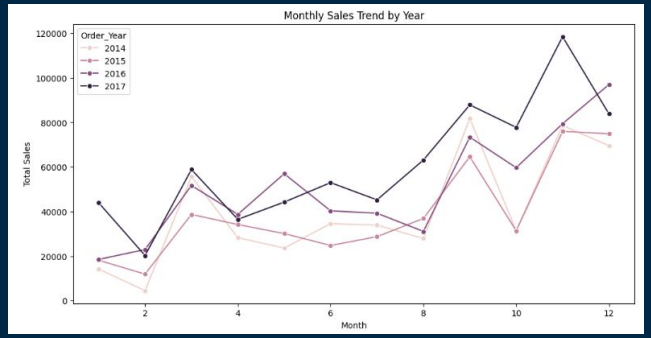
EXPLORATORY DATA ANALYSIS

Orders vs Date



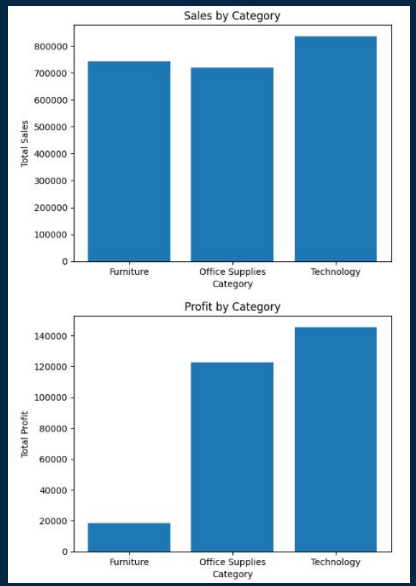
Clear cyclical pattern

Sales Trend for Each Year



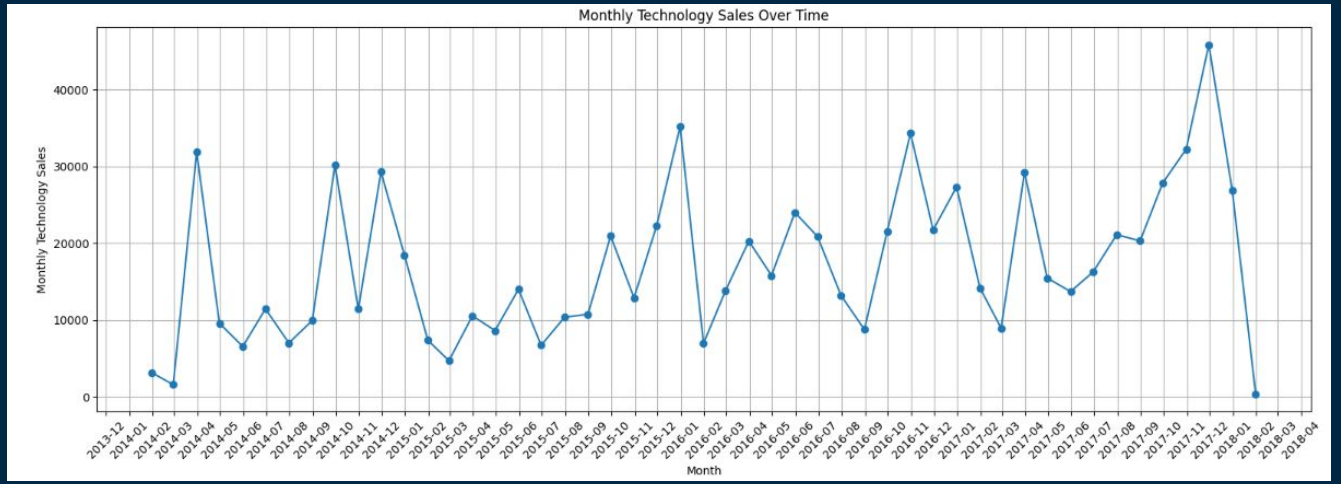
Similar upward sales trends for years 2014 to 2017

Sales and Profit



Technology showcased highest profit

EXPLORATORY DATA ANALYSIS



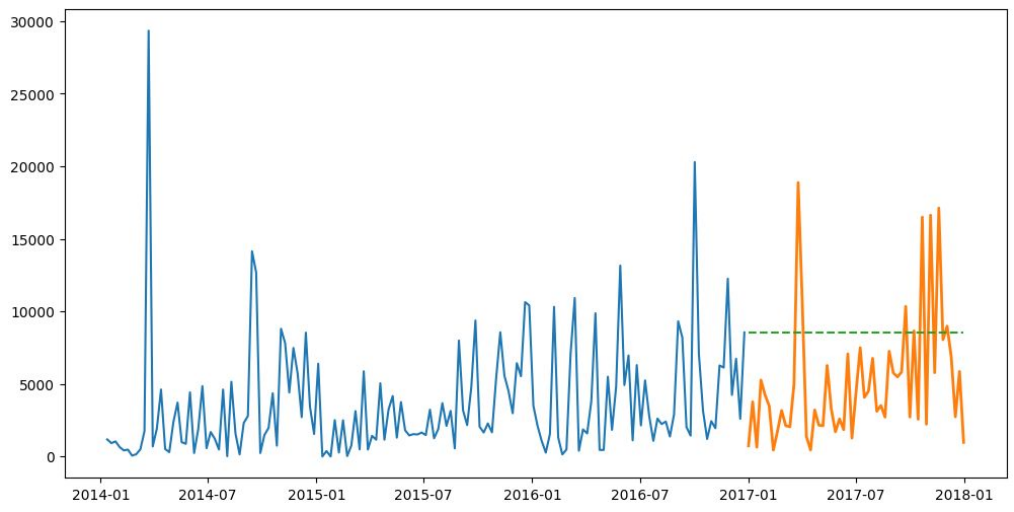
Monthly sales trend for only Tech, which is not as clear as total sales trend. This motivated us to explore whether we can accurately model it

MODELING ROADMAP

BASELINE

BASELINE MODEL

While later models had the results of previous model to compare to, we needed at least some baseline to compare



Metric	Value
MAE	4,891.68
RMSE	5,476.59
MAPE	260.66%
MdAPE	126.68%
Coverage	5.7%

ARIMA/SARIMA

ARIMA

Why ARIMA?

- Good first benchmark before adding in seasonality
- Traditional statistical time series model for forecasting based on past values and errors
- Captures trend (AR), lagged errors (MA), and stationarity adjustments (I)
- Want to understand the effect of trend on weekly tech sales by itself

Our Process

- Used our aggregated weekly revenue (208 weeks)
- Ensured that the time series is stationary & consistent
  - Ran ADF test to understand if we needed to process the data more
    - Had a low p-value (p = 0.0000)
- Performed a grid search to find best p,d,q combination
  - p (0,1,2), d (0,1), and q (0,1,2)

BEST ARIMA MODEL

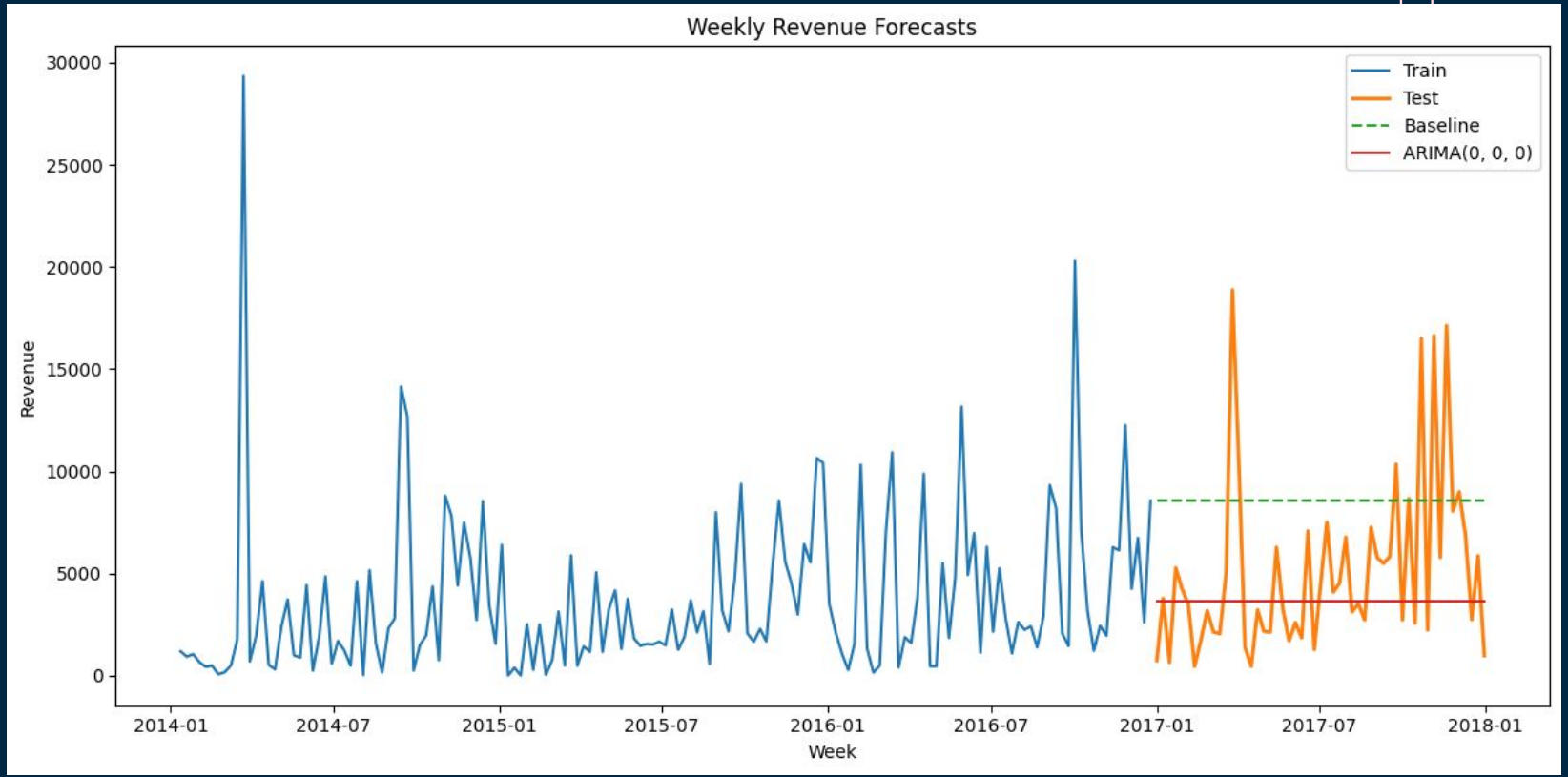
ARIMA (0,0,0)

Insufficient for retail forecasting

- Compared to Baseline: ARIMA underperformed significantly
- Failed to capture holiday spikes and repeated patterns across years

Metric	Value
AIC	3012.72
MAE	2,998.11
RMSE	4,541.87
MAPE	95.19%
MdAPE	48.61%
Coverage	5.7%

BEST ARIMA MODEL



SARIMA

Why SARIMA?

- Extends ARIMA by adding seasonal components to capture time-based patterns
- Captures repeating yearly patterns
- More ideal for weekly forecasting with repetitive patterns

Our Process

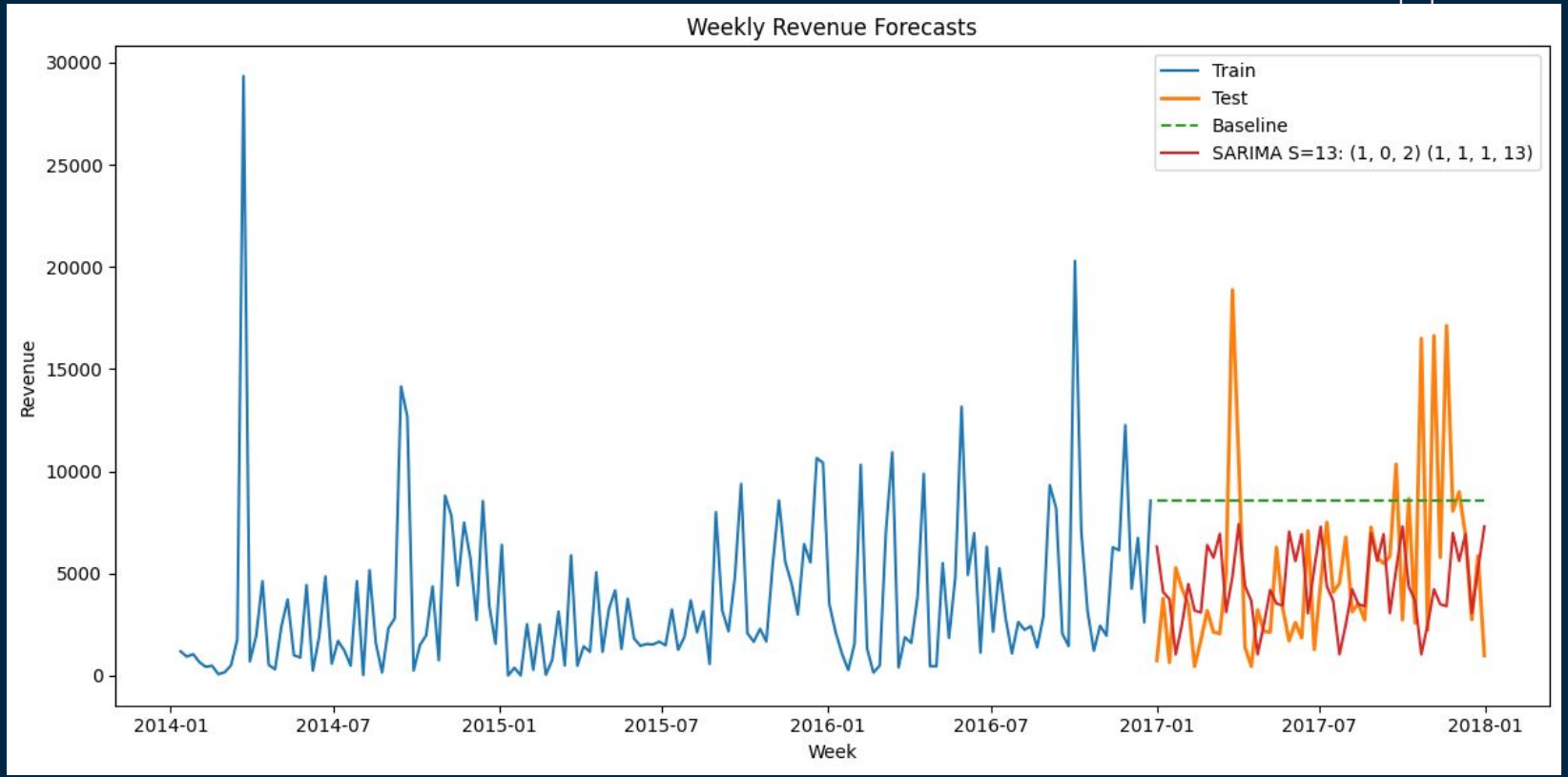
- Detected potential seasonal periods using Autocorrelation Function
  - Scanned for repeating peaks in past sales
  - Strong signal at lag-52 weeks indicating yearly pattern
- Also tested other cycles to compare
  - 52 (annual), 13 (quarterly), 4 (monthly), 2 (bi-weekly)

SARIMA MODELS

Best Model: **SARIMA (1,0,2)(1,1,1,13)**

Seasonal Period	SARIMA	AIC	RMSE	MdAPE	Coverage
52 (yearly - assumed)	(1,0,2)(1,1,1,52)	967.06	7,540.25	89.50%	3.8%
13 (quarterly)	(1,0,2)(1,1,1,13)	2440.98	4,814.00	49.76%	11.3%
4 (monthly)	(1,0,2)(1,1,1,4)	2787.51	4,908.93	55.72%	9.4%
2 (bi-weekly)	(1,0,1)(1,1,1,2)	2904.15	4,628.88	56.32%	15.1%

BEST SARIMA MODEL



PROPHET

PROPHET

Why Prophet?

- Forecasting model that captures trends and seasonality using flexible, additive time-series framework
  - Sales(t) = Trend(t) + Seasonality (t) + Holiday effects + error
- Used as a starting model
  - Fast to implement
  - Used for forecasting problems in industry

Our Process

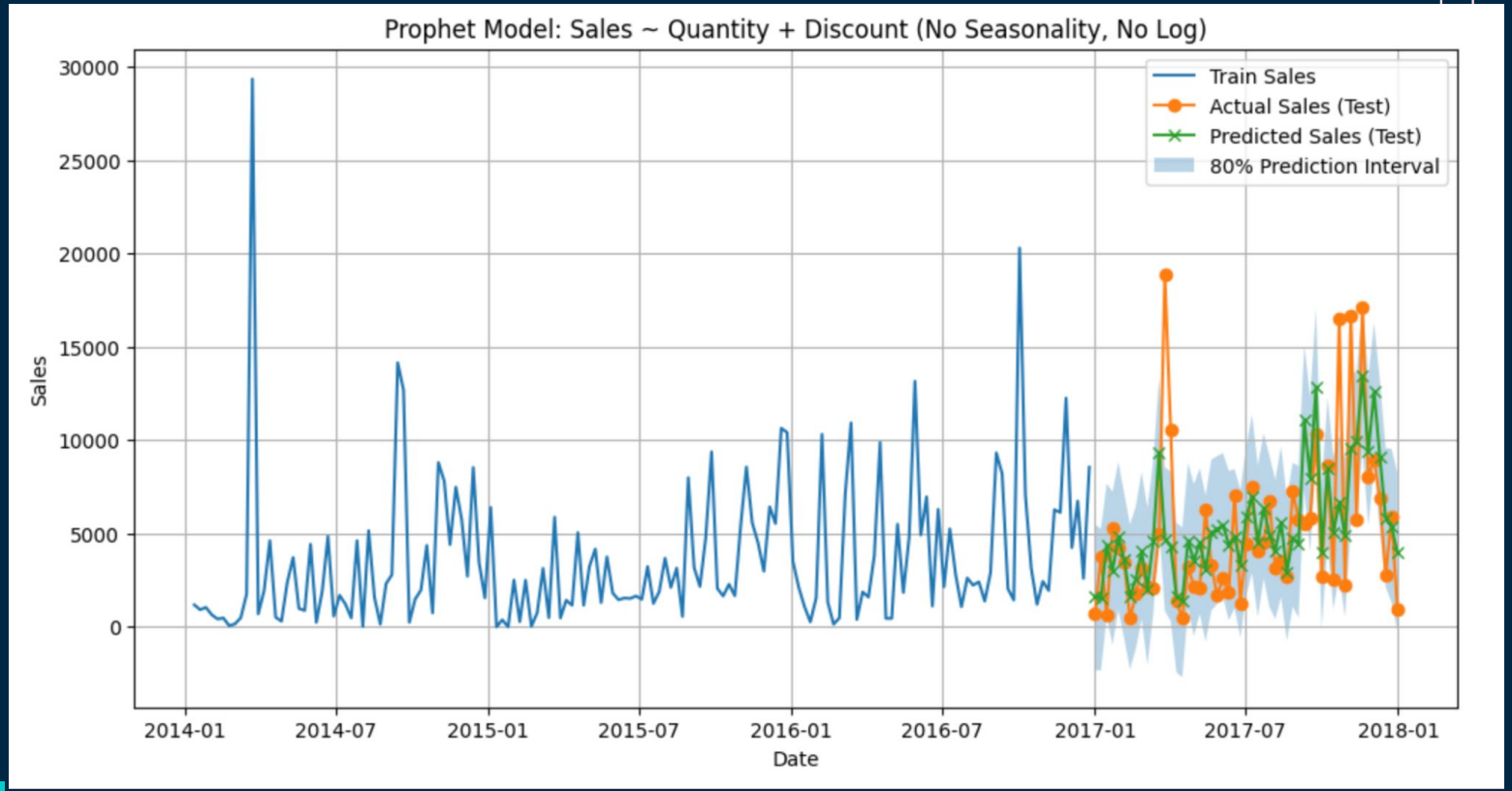
- Created initial model
- Tested different models with different type and number of features, feature transformations, seasonality components

PROPHET MODELS

Best Model: V1 - Simple Model with Sales, Quantity, Discount

Model Version	Model Description	RMSE	MdAPE	Coverage
V1	Simple model with Sales, Quantity, Discount	3495	47.41%	88.68%
V2	Model with Sales, Quantity, Discount, Quantity * Discount	3555.87	51.63%	84.91%
V3	Model with Sales, Quantity, Discount, and Ship Mode and Region One-Hot Encoded Variables	3482.76	47.75%	83.02%
V8	Model with Sales, Quantity, Discount, weekly and yearly seasonality	3688.33	49.33%	77.36%

BEST PROPHET MODEL



RANDOM FOREST

RANDOM FOREST

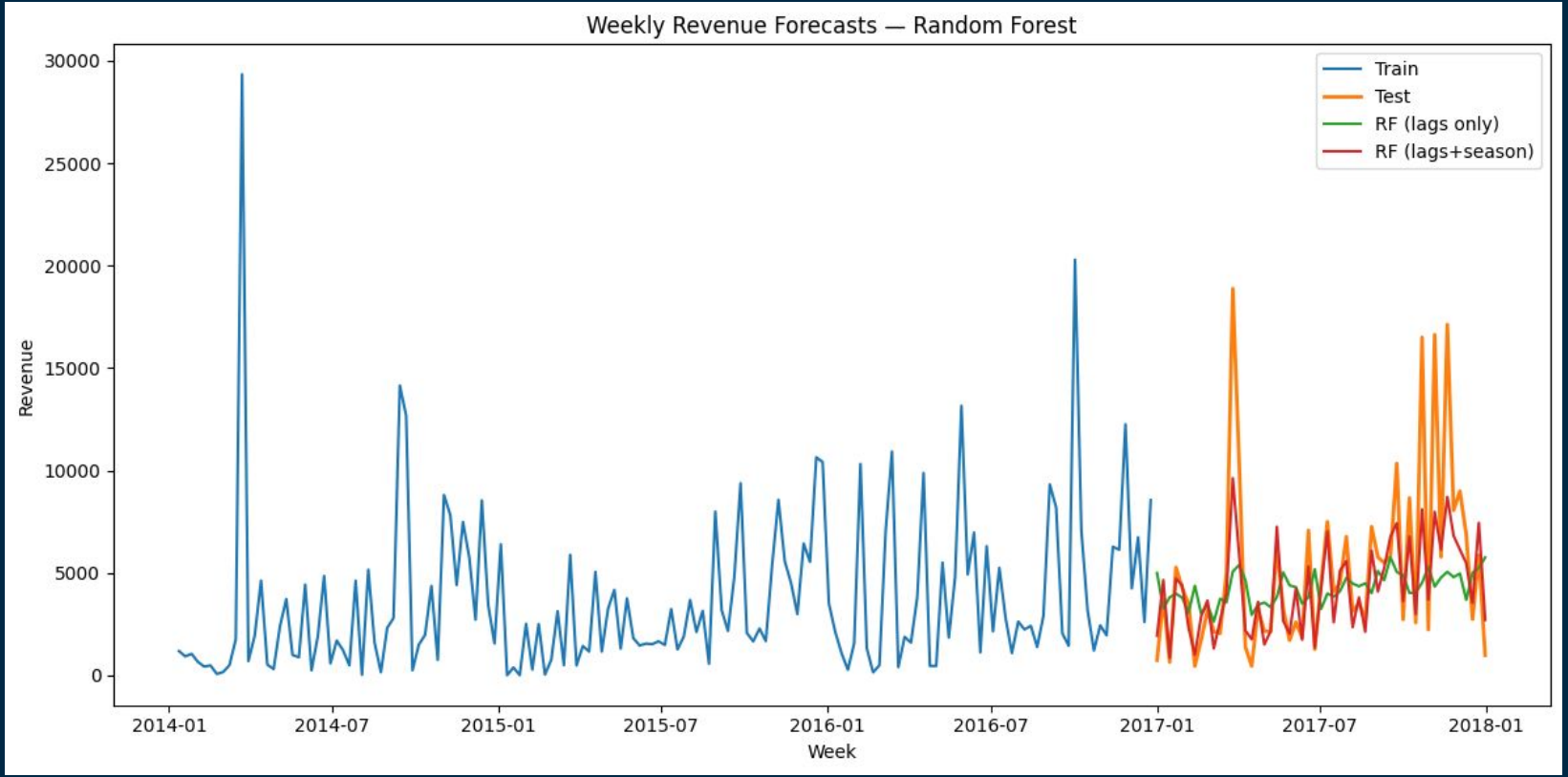
Why Random Forest?

- Traditional time-series models (like ARIMA) can struggle with overlapping seasonal patterns (e.g., weekly vs yearly)
  - Must explicitly specify seasonal period(s)
- Random Forest can better capture patterns automatically through feature engineering
- Ensemble method that reduces risk of overfitting

Our Process

- Created lag features, i.e., past values of Sales in weekly increments
  - Compared to baseline
- Included cyclical element (sine/cosine) to model seasonal cycles. Used rolling averages. Added holidays
  - Compared to previous iteration and baseline

RANDOM FOREST RESULTS



RANDOM FOREST RESULTS

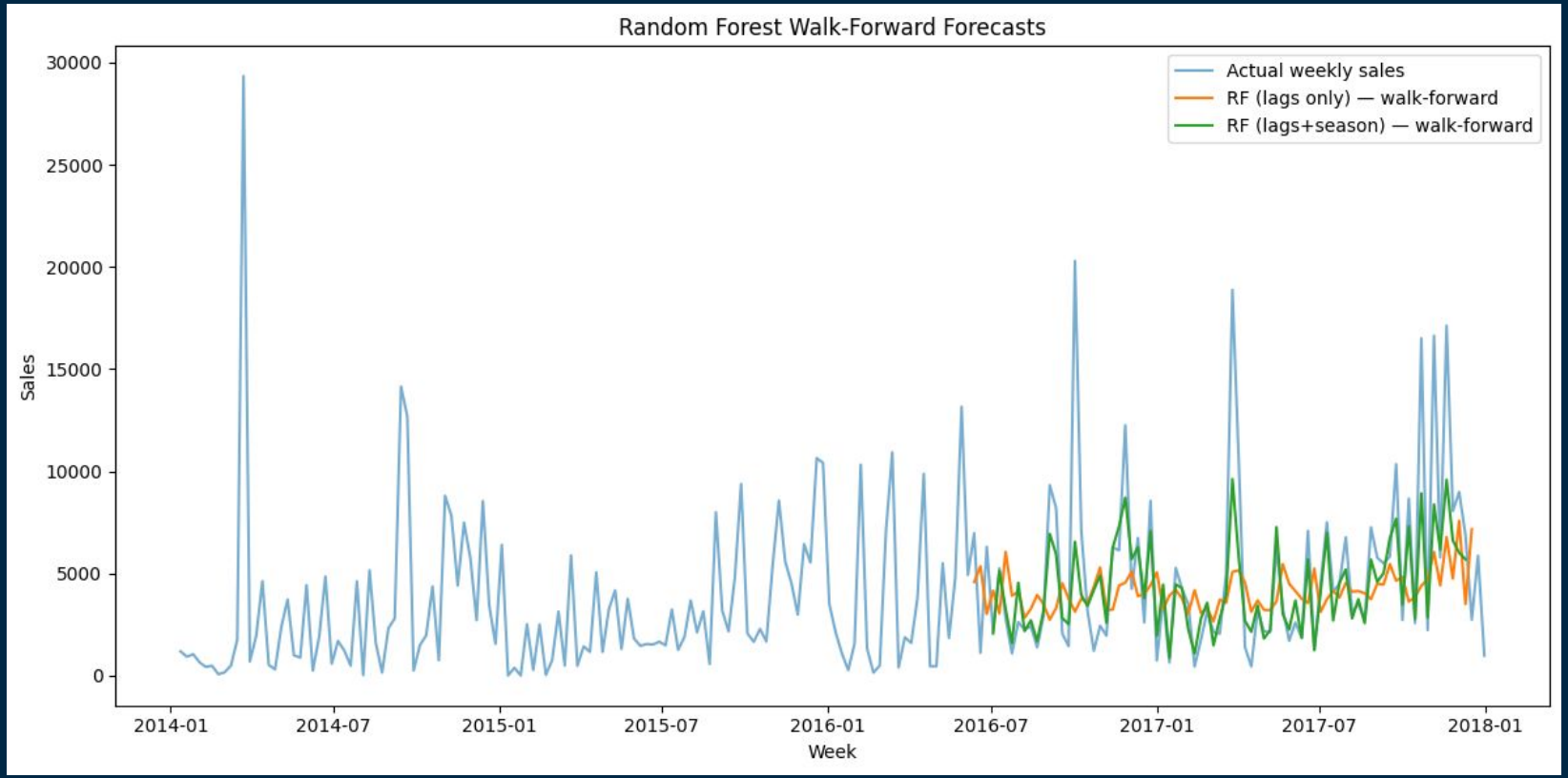
Model Version	Model Description	RMSE	MdAPE
Baseline	Naive	5476.59	126.68%
Lags	Trained RF model using only past sales (Lags)	4226.89	49.06%
Lags + Seasonality	Added calendar and rolling features.	2693.33	24.69%
Lags (Walk-forward)	Validation of previous models using rolling measurement	4246.66	50.38%
L + S (Walk-forward)		2789.72	21.19%

VALIDATION OF RESULTS

Walk-forward

- Initial models used traditional holdout method to evaluate
  - Training: 2014-2016
  - Test: 2017
- Rather than evaluate single time (i.e., 2017), we can use a rolling evaluation of 4 week periods to continuously retrain and re-evaluate model performance
- More accurately simulates usage in real-time
- Captures more robust and reliable estimate of model's ability

WALK-FORWARD VALIDATION



XGBOOST

XGBOOST

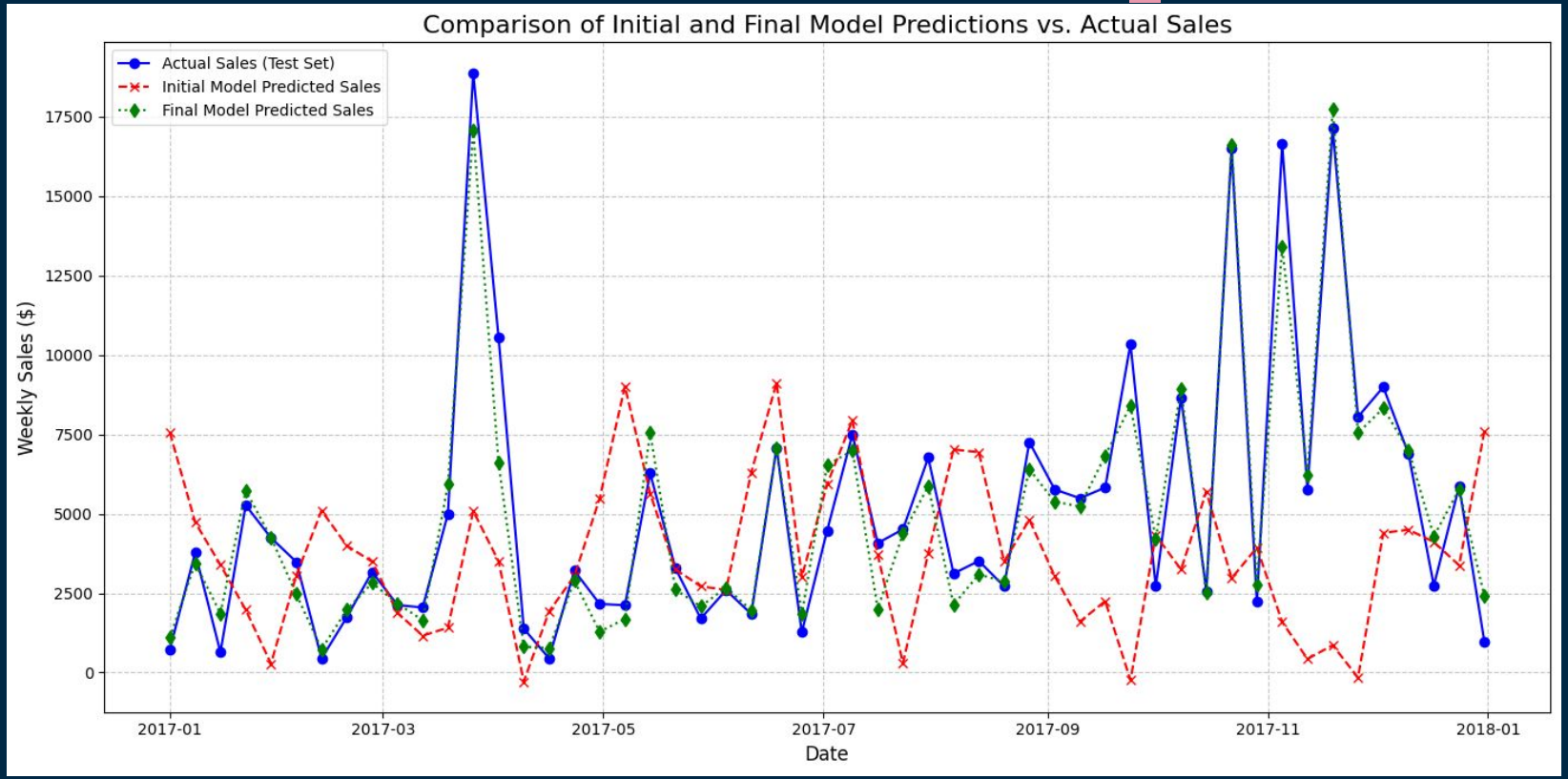
Why XGBoost?

- Extreme Gradient Boosting model that uses an **ensemble learning algorithm**
- Sequential tree boosting where each new tree is fit to the gradient of the loss function (errors), correcting the mistakes of previous trees
- Handles noisy data and non-linearity, and able to handle trends via lags
- Used by other companies (Amazon, Uber, Walmart) to forecast demand

Our Process

- Modeled and predicted weekly sales using:
  - Feature Selection
  - Seasonality
  - Cross validation
  - Randomly sampled hyperparameter tuning

XGBOOST MODEL PROGRESSION



XGBOOST RESULTS

Model Version	Model Description	Test RMSE	Test MdAPE
1	Non seasonal baseline model using only lags	5419.67	66.79%
2	Seasonal Feature Rich Version with Lags and rolling statistics model	1713.10	22.21%
3	Seasonal one-hot encoded model	2537.48	26.89%
4	Randomly sampled randomized hyperparameter tuning (model depth and learning rate) with one-hot encoding	2179.641	27.44%
5	Randomized hyperparameter tuning on all hyperparameters with one-hot encoding	2284.77	20.55%
6	Lag- and rolling-based XGBoost with MI-only feature selection and CV	1,134	16.20%
7	Seasonal, holiday-aware XGBoost with multiplicative interactions, lags, MI-selected features, and nested CV	1,079.58	12.71%

FINAL XGBOOST MODEL

Seasonal and holiday-aware XGBoost with lagged, MI-selected features and nested CV

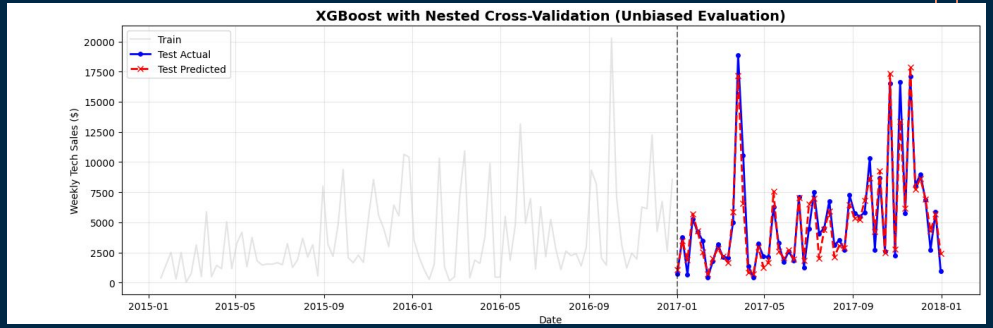
Goal: Unbiased hyperparameter tuning and evaluation with rich seasonal drivers

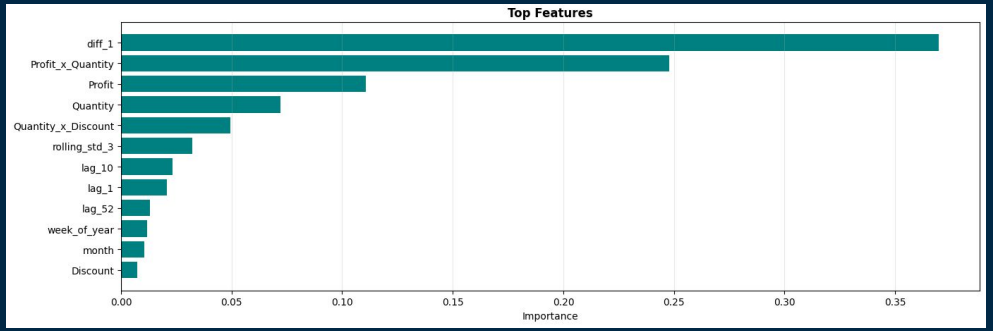
- Test RMSE: 1,079.58
- Test MdAPE: 12.71%

Mutual information used to select ~20 highest-signal features

Strength: statistically rigorous, leakage-free evaluation with strong percentage accuracy on test set

Weakness: outer CV error is pessimistic (small data, volatile early folds), creating a gap vs. final test RMSE





PROCESS & ATTEMPTS (Including Failures)

What we tried

- Cross validation for XGBoost Models

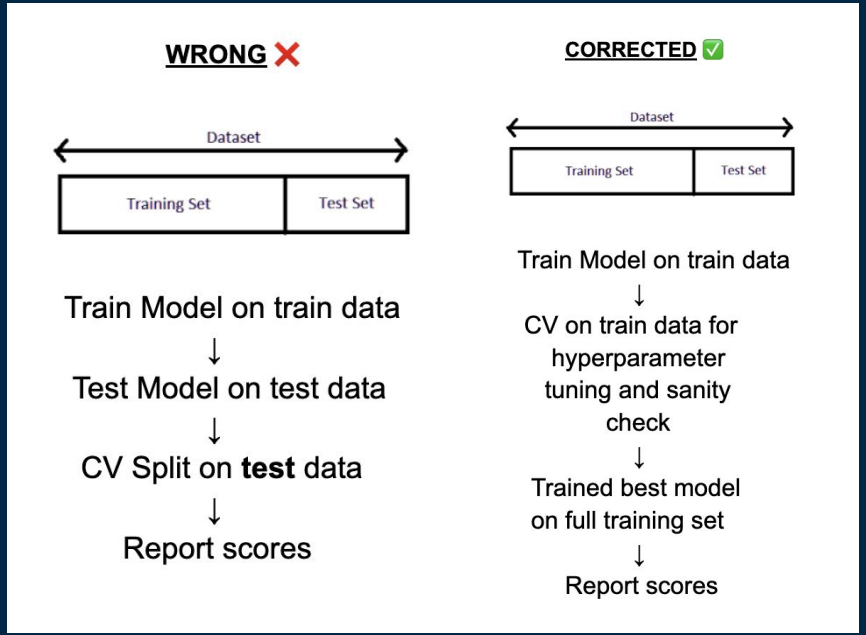
Mistake

- Performed Cross validation on testing data → optimistic model results.

What we learned

- Data leakage can easily occur from not building, training, and testing model in correct order.

Workflow diagram:



PROCESS & ATTEMPTS (Including Failures)

What we tried

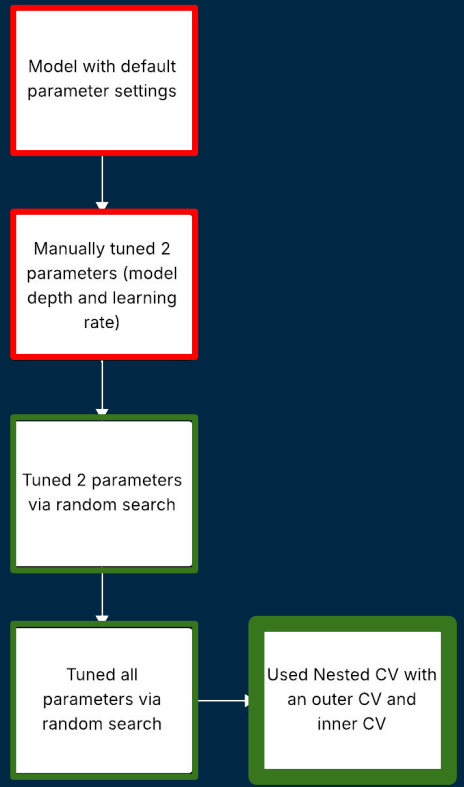
- Manual hyperparameter tuning for model depth (md) and learning rate (lr)

Why we thought it might work

- Other parameters depend on md and lr, and tuning these allows for fewer possible combinations of values

What we learned

- Manual hyperparameter tuning is error-prone and introduces bias.
- Tuning loop to randomly sample combinations avoids data contamination
- Using a nested CV gives unbiased performance scores
  - Outer CV:Estimates true performance on unseen data
  - Inner CV: Tunes hyperparameters without contaminating performance estimate



LSTM

LSTM

Why LSTM?

- Neural network approach that learns patterns directly from raw data
- Processes sequences through memory cells that retain long-term information
- Want to compare deep learning approach against traditional statistical methods

Our Process

- Used our aggregated weekly revenue (208 weeks)
- Ensured proper data methodology to avoid leakage & contamination
  - Train/Validation/Test split (no iterating on test results)
  - Scalers fit only on training data
- Applied techniques to handle small dataset size
  - Data augmentation (4x training samples via noise/scaling jitter)
  - Heavy regularization (L2, Dropout, Gaussian Noise)
- Model receives only raw sales sequences + time encoding
  - No hand-crafted features (rolling mean, momentum, etc.)

BEST LSTM MODEL

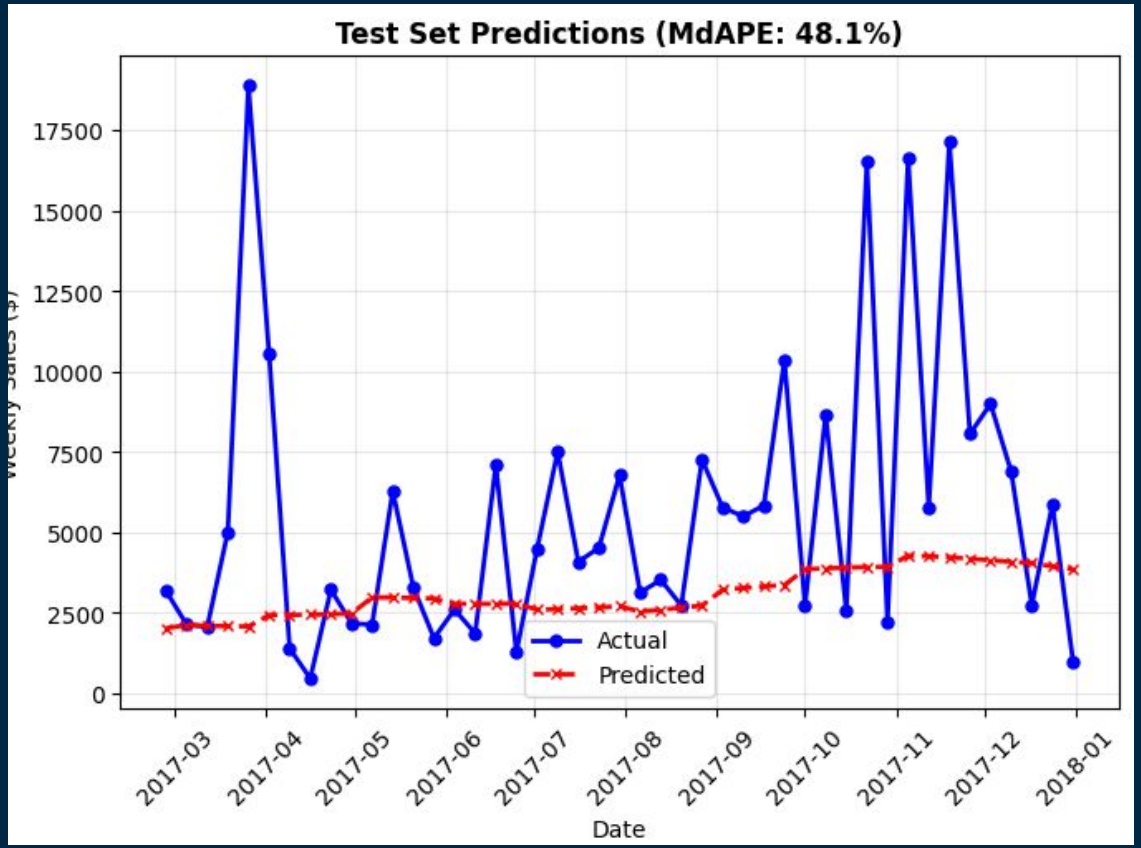
Regularized LSTM (Lookback=8, Units=24)

Metrics comparable to ARIMA/SARIMA; underperformed against Tree models

- **Marginal Improvement:** Comparable to the ARIMA (48.1% vs 48.6%) despite significantly higher complexity.
- **Data Scarcity:** The deep learning architecture struggled to generalize on the small dataset (208 samples) compared to Random Forest and XGBoost.
- **Volatility:** Failed to capture extreme holiday spikes as effectively as the feature-engineered tree models.

Metric	Value
MAE	3,252
RMSE	4,960
MAPE	61.7%
MdAPE	48.1%
Val Los	0.53

BEST LSTM MODEL



04

EXPERIMENTS & EVALUATION METRICS

EXPERIMENTAL SETUP (GENERALIZED)

- Subsetted to technology category
- Initial Train / Test Split for Models
  - Train data: data < 01/01/2017 (~70% of all data)
  - Test data: data >= 01/01/2017 (~30% of all data)

- Baseline models
  - SARIMA model using lag 52 (SARIMA Models)
  - Simple XGBoost model with only lags (XGBoost Models)
  - Simple Prophet model with only Sales, Quantity, Discount (Prophet Models)

- Cross validation
  - XGBoost
  - Random Forest

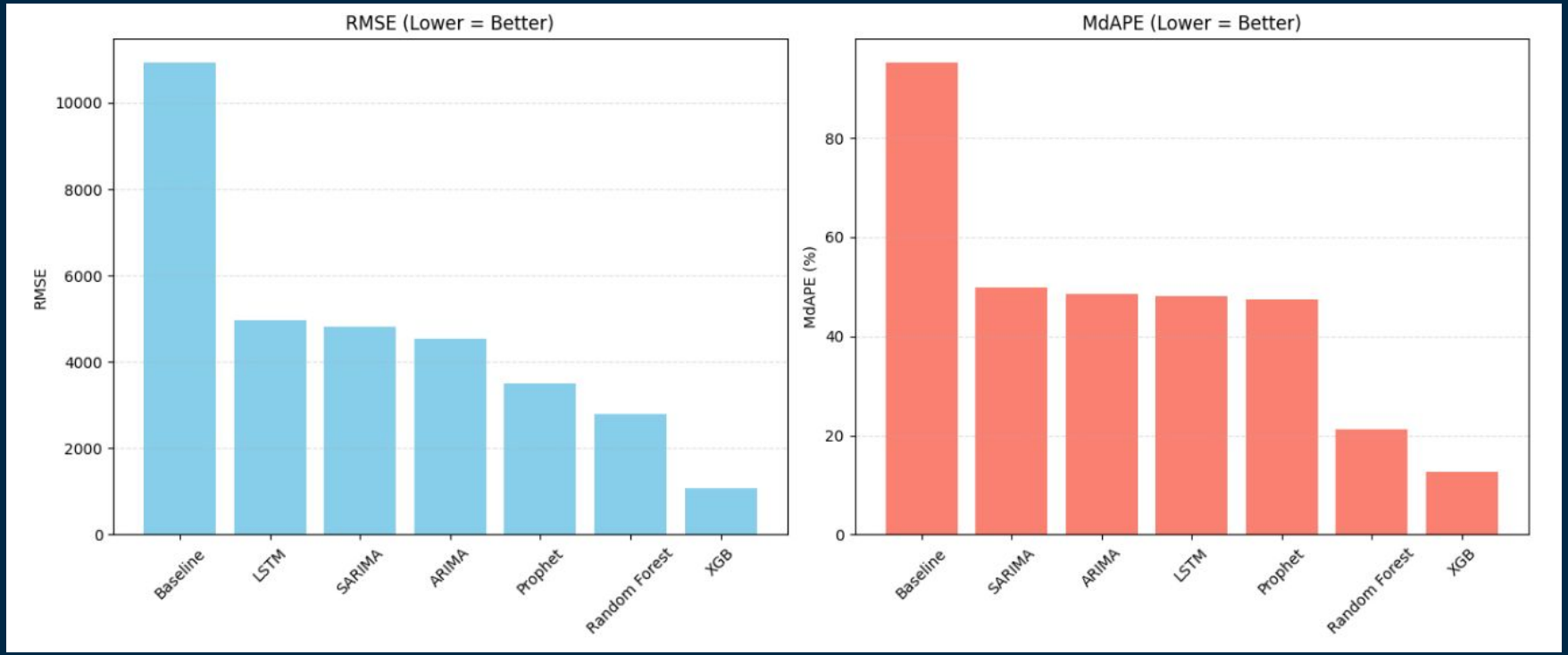
METRICS CONSIDERED

Metrics Considered	Why is it Appropriate?	What Does it Mean?
RMSE	Flags models with large deviations from forecast	Low value: Fewer large spikes mispredicted (EOY spikes)
MAE	Measures prediction error in $, interpretable for business impact	Low value: Predicted value closer to real weekly sales
MAPE	Allows for fair comparison across weeks with high vs low sales, regardless of fluctuations	Low value: Predicted value close to true sales regardless of scale
MdAPE	Gives robust central tendency, and is less sensitive to holiday outliers	Low value: typical weekly forecast error is low
COVERAGE	Indicates range of possible weekly sales outcomes and how confident we can be that the prediction is accurate	High value: strong reliability and good uncertainty estimation under fixed interval width

KEY METRICS: RMSE & MdAPE

Metric	Focus	Justification
RMSE	Magnitude of error (dollar amount)	**Penalizes large errors heavily**, ensuring the model accurately predicts crucial, high-revenue events like holiday spikes (EOY).
MdAPE	Typical error (percentage)	**Ignores the impact of extremities**, reflecting the typical or median forecast accuracy in a way that is robust against extreme, unpredictable spikes or troughs.

EVALUATION METRICS MODEL COMPARISON



INTERPRETATION & ERROR ANALYSIS

05

INSIGHTS

- Ensemble methods produced the best results with XGBoost performing the best overall
  - XGBoost can capture seasonal and other complex patterns that traditional time-series models are unable to capture
- XGBoost is a perfectly valid method for forecasting
- This matches our hypothesis. We thought this would be the case because XGBoost is a very popular model due to its performance on structured data
- Seasonality exists in our data, which makes sense for retail data

ERROR ANALYSIS

Model	Key Limitation	RMSE	MdAPE	Coverage
ARIMA	Captured trend only (no spikes)	4,541.87	48.61%	5.7%
SARIMA	Detected seasonal timing but not magnitude	4,814.00	49.76%	11.3%
Prophet	Handled seasonality but cannot handle multiplicative effects	3,495	47.41%	88.68%
Random Forest	Learned short-term patterns, struggled with long-term	2,789.72	21.19%	-
XGBoost	Combined seasonality, lags, promotions well but risk of overfitting	1,079.58	12.71%	-
LSTM	Captured sequential trends but failed to predict peak magnitudes	4,960	48.10%	-

06

SUMMARY & TAKEAWAYS

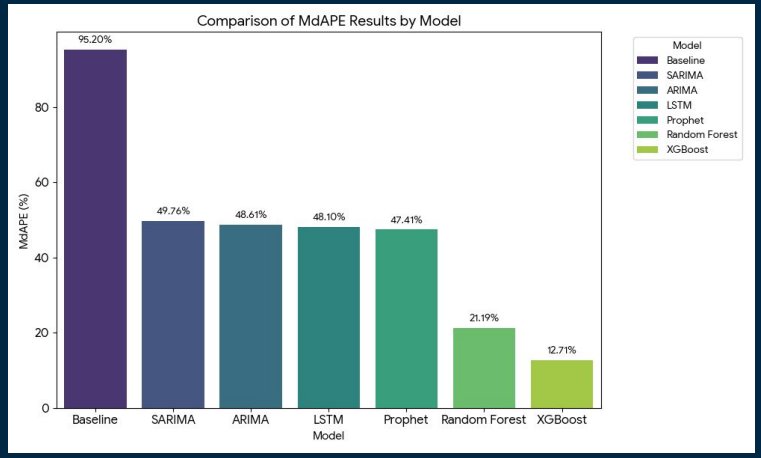
Summary

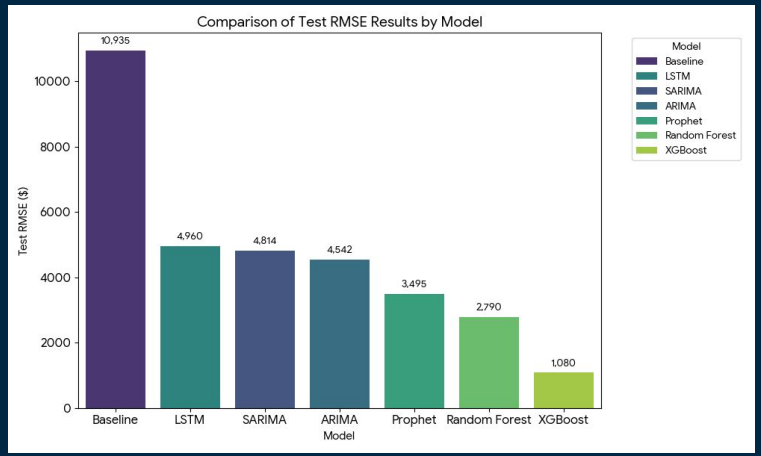
The Experiment:

- **Data Preparation:** Data cleaning, Exploratory Data Analysis (EDA), and advanced feature engineering.
- **Modeling:** Benchmarked 6 distinct forecasting architectures (ARIMA, SARIMA, Prophet, Random Forest, XGBoost, LSTM) on weekly technology sales.
- **Evaluation:** Ranked model performance using RMSE and MdAPE.

Final Recommendation:

- Deploy the Seasonal & Holiday-Aware XGBoost model.
- Most consistent accuracy across the entire year, minimizing error margins for standard operations while successfully capturing critical demand surges.





KEY TAKEAWAYS

Lessons Learned in Feature Engineering & Evaluation

XGBoost & Seasonality: The Best Formula

- **Best Performance:** XGBoost optimized with seasonal features yielded the highest accuracy.
- **ML + Seasonality:** Combining ML with seasonal signals beats classical methods.

Seasonality Drives Model Decisions

- **Cycles Matter:** Strong annual patterns ruled out generic baselines.
- **Model Selection:** Success required seasonality-aware models (SARIMA, Seasonal XGBoost).

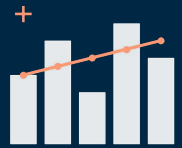
Interactions Unlock Hidden Value

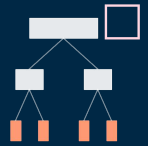
- **Beyond Raw Data:** Single features tell only half the story.
- **Complex Drivers:** Interaction terms (e.g., Quantity × Discount) captured critical relationships between business drivers.

Rich Features > Simple Lags

- **Lags Insufficient:** Relying solely on past sales history led to poor forecasts.
- **Feature Engineering:** Accuracy depended on rich features like rolling statistics and holiday flags.

IDEAS FOR THE FUTURE





1. Uncertainty Quantification

Implement conformal prediction to provide risk-adjusted confidence ranges for inventory planning.

2. Integrate External Regressors

Add macroeconomic indicators (CPI) & competitor pricing to capture broader trends.

3. Increase Granularity

Move from Category-level forecasting to Sub-Category or SKU-level for precise inventory.

THANK YOU!

ANY QUESTIONS?