ARIMA vs XGBoost — Who Wins the Forecasting Battle?

A Real-World Comparison Using Pharma Weekly Sales Data (2019–2021)

Forecasting is the lifeblood of business analytics — from pharma sales and supply chain demand to finance and retail.
But in an age where machine learning dominates, a big question remains:

> **Does XGBoost truly outperform ARIMA, or does the classic statistical model still rule stable time series forecasting?**

In this project, we put both to the test using **weekly national pharma sales data**.

Dataset Overview

We analyzed 3 years of weekly sales data (2019–2021) containing:

- `sales_units` → total national sales
- `calls_made` → number of marketing calls made in that week
- `is_holiday_week` → binary indicator for reduced activity

Objective:
Predict next 10 weeks of sales and compare model performance across four setups.

The Four Models Compared

#	Model Type	Inputs	Core Strength
ARIMA (No Exog)	Statistical	Only past sales	Stable & interpretable

ARIMA (With Exog)	Statistical + Regression	Sales + Calls + Holiday	Explains cause-effect
XGBoost (No Exog)	Machine Learning	Lag features only	Learns nonlinear dependencies
XGBoost (With Exog)	Machine Learning + Regression	Lag + Calls + Holiday	Adapts to business events

---

1 ARIMA (No Exogenous Variables)

Parameters:

```
order = (2,0,2)   # p=2, d=0, q=2
trend = 'ct'      # constant + linear trend
```

Concept:
ARIMA predicts future values purely based on past sales.
It's great when the data is **smooth, stable, and trend-driven**.

Performance (No Exog):

Metric	ARIMA	XGBoost
RMSE	**339.49**	782.36
MAE	**234.44**	633.72

Winner: ARIMA
Despite XGBoost's sophistication, ARIMA's statistical simplicity fit the series better.

---

2 ARIMA (With Exogenous Variables)

Parameters:

```
order = (2,0,2)
trend = 'ct'
exog = ['calls_made','is_holiday_week']
```

Concept:
We added **external business drivers**:

- `calls_made` → field activity
- `is_holiday_week` → expected downtime

This turns ARIMA into a hybrid regression + time series model that models *both* trend and cause-effect.

Performance (With Exog):

Metric	ARIMA	XGBoost
RMSE	**195.05**	563.43
MAE	**181.84**	498.16

Winner: ARIMA (With Exog)
Adding exogenous variables improved both models — but ARIMA improved *much more*, achieving the lowest overall errors.

---

3 XGBoost (No Exogenous Variables)

Parameters:

```
n_estimators=400,
learning_rate=0.05,
max_depth=5,
subsample=0.8,
colsample_bytree=0.8
```

Concept:
We built lag-based features:

- `lag_1`, `lag_2`, `rolling_mean_3`

Then trained XGBoost to learn nonlinear lag patterns.
It's flexible — but without exogenous inputs, it only "sees" short-term memory.

Result:
Couldn't beat ARIMA on this smooth, stable data.
Overfitted slight noise and missed the consistent trend.

---

4 XGBoost (With Exogenous Variables)

Features:
`lag_1`, `lag_2`, `rolling_mean_3`, `calls_made`, `is_holiday_week`

Concept:
Combines temporal memory (lags) with real-world context (calls, holidays).
This gives it a more complete understanding of business activity.

Feature Importance (XGBoost):

1. `lag_1` – Strongest influence
2. `lag_2`, `rolling_mean_3` – Trend smoothers
3. `calls_made` – Positive marketing effect
4. `is_holiday_week` – Predicts downturns

Result:
Performs competitively but still not as efficient as ARIMA for this dataset size and structure.

Performance Summary

Model	RMSE	MAE	Strength
ARIMA (No Exog)	339.49	234.44	Stable trend model
ARIMA (With Exog)	**195.05**	**181.84**	Best performer overall
XGBoost (No Exog)	782.36	633.72	Overfits, lacks context
XGBoost (With Exog)	563.43	498.16	Improves with features, still higher error

---

Interpretation

1. **ARIMA outperforms XGBoost** on this dataset because the time series is smooth, stationary, and strongly autocorrelated.
2. XGBoost underperforms when given too few samples or lag features.
3. Adding exogenous variables (`calls_made`, `is_holiday_week`) enhances both models — but ARIMA benefits more since these features directly align with its linear framework.
4. In real-world, volatile data (more features, multiple regions, promotions), XGBoost would eventually overtake ARIMA.

Final Model Decision

Scenario	Winning Model	Reason
No external drivers	**ARIMA (No Exog)**	Simpler, stable fit
With business drivers	**ARIMA (With Exog)**	Best overall accuracy & explainability
Highly nonlinear or dynamic data	**XGBoost (With Exog)**	Adapts to complex relationships

"Even though XGBoost is a powerful machine learning model, ARIMA remains unbeatable when the data behaves like a true time series — smooth, seasonal, and predictable."

Full Project Resources

GitHub Notebook: [your-github-link-here]
Dataset: Weekly pharma sales (2019–2021)

Key Takeaway

> “ARIMA interprets the past — XGBoost learns the future.”

Both models have their place — it’s not about which algorithm is “better,” but about **which one fits the story your data is trying to tell.**