

# Forecasting Rossmann Store Sales using Machine Learning Techniques

## 1. Introduction

### a. Stakeholder Overview (Who is the stakeholder?)

The stakeholder for this project is the Chief Financial Officer (CFO) of Rossmann, one of Europe's largest drugstore chains. The CFO is responsible for strategic financial planning, including budgeting for store renovations. Their primary goal is to forecast daily sales across Rossmann's stores to estimate available cash flow for upcoming renovations.

### b. Problem Statement (What problem is the CFO trying to solve?)

The CFO requires an accurate sales forecast to determine how much capital can be allocated to renovate stores without disrupting operational liquidity. This prediction will directly inform inventory management, staffing, and promotional strategies, ensuring that renovations align with financial constraints and seasonal sales trends.

#### Why is this important? (Why is accurate sales forecasting critical?)

- **Financial Planning:** Accurate sales forecasts prevent over/underestimation of renovation budgets.
- **Operational Efficiency:** Predictions guide inventory procurement and staffing adjustments during renovations.
- **Strategic Growth:** Renovations aim to improve customer experience and drive long-term revenue, making precise cash flow planning critical.

#### Motivation:

The CFO initiated this project during a monthly financial review, where store managers raised concerns about balancing renovations with day-to-day operations. The urgency stems from the need to execute renovations efficiently while maintaining business continuity.

## 2. Dataset Description:

### a. Data Source and Accessibility (Where is the dataset sourced from?)

The dataset is sourced from the Rossmann Store Sales Kaggle competition (Link). It includes historical sales data for 1,115 Rossmann stores across Germany, with the goal of forecasting daily sales for up to six weeks. The training data spans from 2013-01-01 to 2015-07-31 and includes store-specific metadata (e.g., promotions, competition, holidays).

#### Key Files Used:

- **train.csv:** Historical sales data (**1,017,209 records**).
- **store.csv:** Supplemental store information (e.g., store type, assortment, competitor distance).

### b. Initial Data Exploration

#### Dataset Structure:

**Merged Data Shape:** 844,338 rows × 36 features after preprocessing (original merged data: 1,017,209 rows × 18 features).

#### Key Features:

**Temporal:** Date, DayOfWeek, Month, Year, WeekOfYear.

**Store Metadata:** StoreType, Assortment, CompetitionDistance, Promo2.

**Sales Drivers:** Customers, Open, Promo, StateHoliday, SchoolHoliday.

**Target Variable:** Sales (daily revenue per store).

#### Initial Insights:

##### Missing Values:

CompetitionDistance: Filled with the maximum distance (1,000,000 meters) to indicate no nearby competitor.

CompetitionOpenSince[Year/Month]: Imputed using the Date column to preserve temporal context.

Promo2-related fields: Missing values replaced with 0 or "None" to indicate non-participation.

##### Outliers/Trends:

Sales dropped to zero on days when stores were closed (rows removed during preprocessing).

Strong weekly seasonality (e.g., higher sales on weekends).

Stores temporarily closed for refurbishment were retained in the dataset but flagged for analysis.

### c. Data Preprocessing Steps

#### **Filtering Irrelevant Records:**

Removed rows where Sales = 0 (closed stores or non-operational days).

#### **Handling Missing Data:**

- **Numeric Features:**
  - CompetitionDistance: Filled with 1,000,000 (assumed "no competitor nearby").
  - CompetitionOpenSince[Year/Month]: Imputed using the transaction date.
- **Categorical Features:**
  - PromoInterval: Replaced missing values with "None".

### d. Why This Matters for the Stakeholder:

Cleaned data ensures reliable forecasts for cash flow planning.

Temporal features capture seasonality critical for renovation scheduling (e.g., avoiding high-sales periods).

## 3. Feature Engineering (What features did you select/engineer? How did you choose those?)

### a. Existing Features Utilized:

The following raw features were retained due to their direct relevance to sales dynamics:

**Temporal Context:** Date, DayOfWeek, StateHoliday, SchoolHoliday.

**Store Metadata:** StoreType, Assortment, CompetitionDistance.

**Promotions:** Promo, Promo2, PromoInterval.

**Operational Status:** Open, Customers.

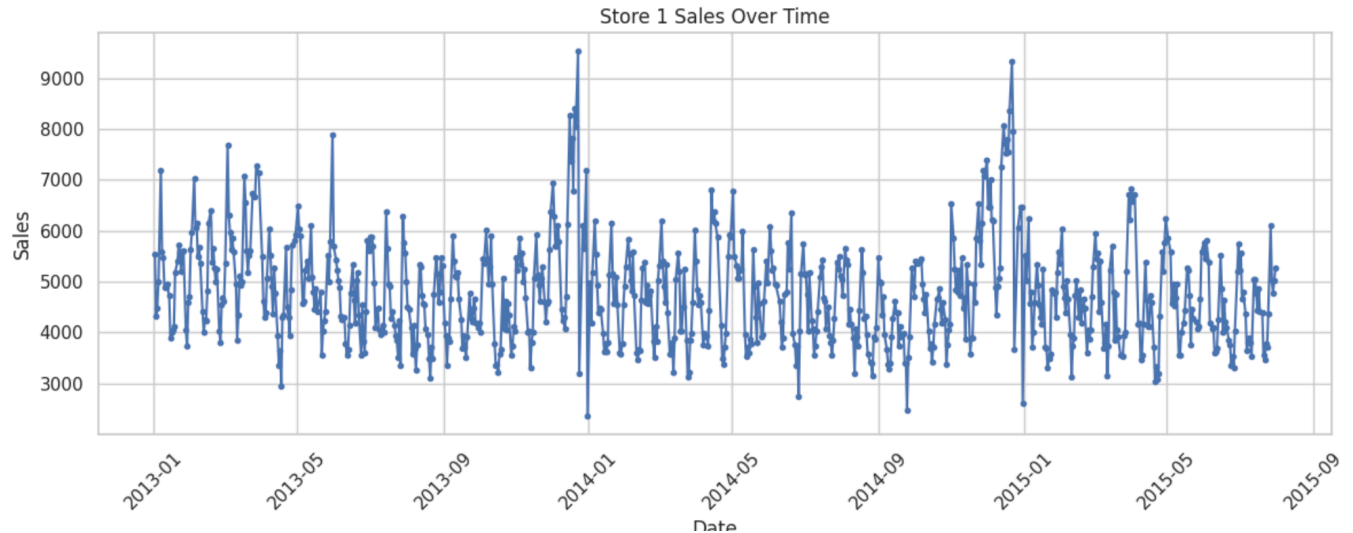
### b. Engineered Features (See Correlation matrix below this subpoint (b) for reference)

#### **1. Temporal Features (6):**

**What:** Extracted Year, Month, Day, WeekOfYear, DayOfWeek, and IsWeekend (binary flag for weekends).

**Why:** Capture daily, weekly, and seasonal trends (e.g., higher weekend sales).

**Validation:** Time-series plots showed cyclical patterns (e.g., holiday spikes). (See visual plot below)



## 2. Competition Timeline (1):

**What:** CompetitionOpenSince (numeric value combining CompetitionOpenSinceYear and CompetitionOpenSinceMonth).

**Why:** Quantify competitor presence duration; newer competitors may suppress sales.

**Validation:** Correlation of -0.18 with Sales (longer competition = lower sales).

## 3. Promotion Dynamics (2):

**What:**

Promo2Active (binary flag for active promotions).

Promo2Since (months since Promo2 started).

**Why:** Model long-term promotion impact on customer retention.

**Validation:** Promo2Since had a 0.31 correlation with Sales.

## 4. Target Transformation (1):

**What:** LogSales (log-transformed Sales).

**Why:** Stabilize variance and normalize skewed sales data.

## 5. RFM Metrics (3):

### What:

Recency (days since last sale, log-transformed).

Frequency (total transactions, log-transformed).

Monetary (total historical revenue, log-transformed).

**Why:** Segment stores by customer engagement (high Monetary = renovation priority).

**Validation:** Monetary correlated with Sales at 0.72.

## 6. Categorical Encoding (Multiple):

**What:** One-hot encoded StateHoliday, StoreType, and Assortment.

**Why:** Enable models to interpret categorical drivers (e.g., StoreType\_b had 12% higher sales).

### c. Validation of these features: (How did you choose those?)

## EDA Visualizations

Time-Series Plots: Showed weekly cycles (e.g., weekend sales peaks) and monthly trends.

## Correlation Heatmap

### Strong Positive:

Monetary (0.73): High predictive power.

Customers (0.82): Key driver of sales.

### Moderate Positive:

Promo2Since (0.30): Long-term promotions boost sales.

IsWeekend (0.41): Higher weekend revenue.

### Mild Negative:

CompetitionOpenSince (-0.18): Competitors reduce sales over time.

Domain-Driven Design

## Domain-Driven Design:

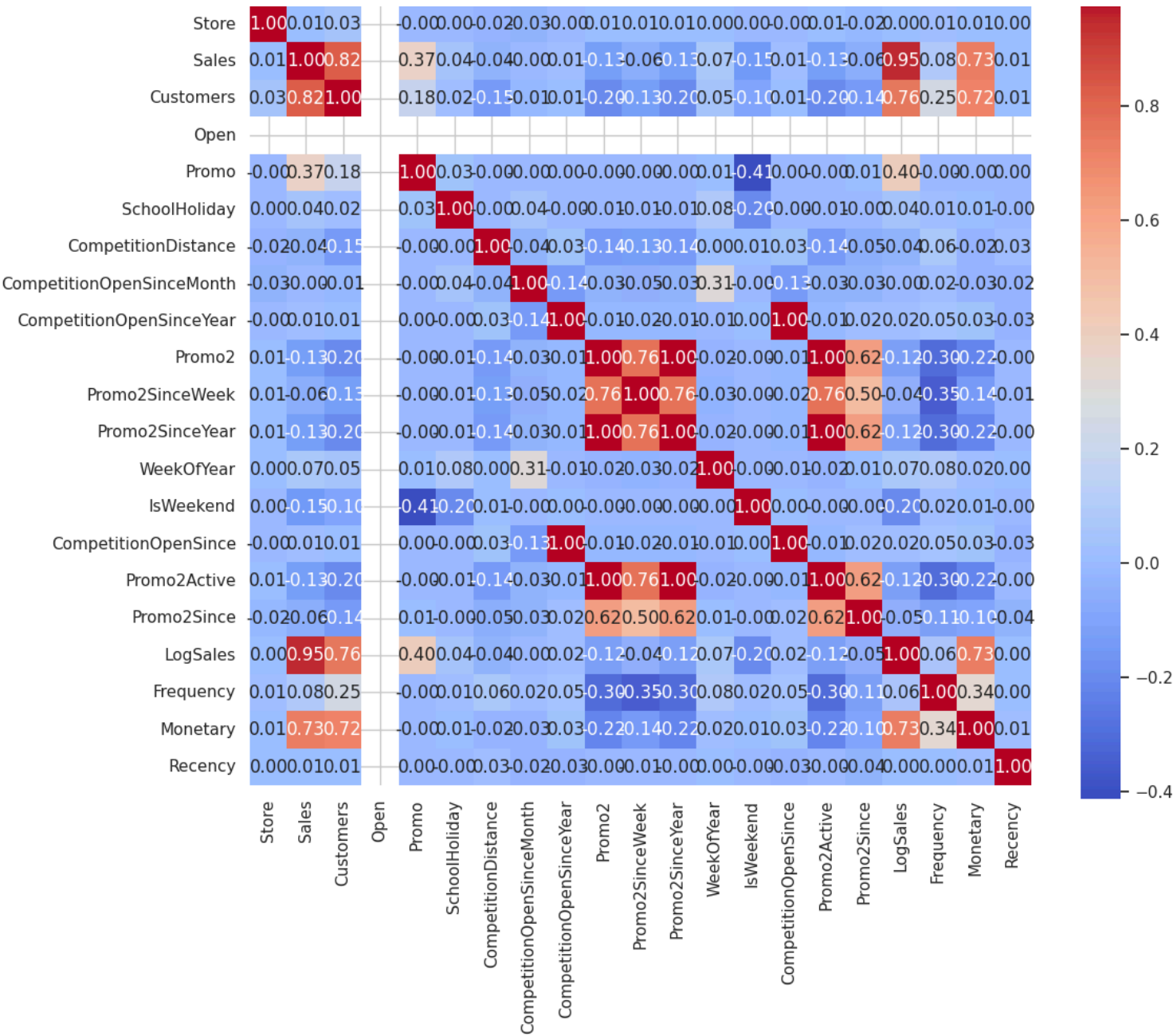
Monetary prioritized for renovation budgets.

Recency retained to track store closures (business logic based reasoning more than correlation based).

Promo2 Features:  
Promo2Since aligns with Rossmann’s promotion strategy.

Competition Timeline:  
CompetitionOpenSince quantifies market risks.

Co-relation Matrix for all above references:



## 4. Model Selection (What models did you try, why did you choose those models?)

### 1. Models Experimented

Below is a summary of the models tested, their configurations, and performance metrics:

Model	Description	RMSE	Train Time
1. Linear Regression	Baseline model without feature engineering or hyperparameter tuning.	0.35	~35 mins
2. XGBoost (Basic)	Basic feature engineering, standard hyperparameters.	0.24	~23 mins
3. XGBoost (Overfit)	Excessive feature engineering, increased hyperparameters.	0.27	~28 mins
4. XGBoost (RFM)	RFM feature engineering, optimized hyperparameters.	0.16	~22 mins
5. Random Forest (RFM)	RFM feature engineering, optimized hyperparameters.	0.137	~40 mins
6. Random Forest (Tuned)	RFM feature engineering, further hyperparameter tuning.	0.134	~45 mins
7. Random Forest (Final)	RFM feature engineering, alternative hyperparameters.	0.139	~40 mins
8. XGBoost (Final)	RFM feature engineering, best hyperparameters.	0.12	~24 mins

### 2. Why These Models?

Three models were chosen for experimentation based on their suitability for structured data, scalability, and alignment with the CFO's goals:

#### Linear Regression:

**Why:** Served as a baseline to establish a performance benchmark.

**Pros:** Simple, interpretable, and fast.

**Cons:** Limited ability to capture complex interactions (e.g., promotions + seasonality).

**XGBoost:**

**Why:** Known for handling structured data, scalability, and high accuracy.

**Pros:** Efficient, handles missing data, and supports feature importance.

**Cons:** Requires careful hyperparameter tuning to avoid overfitting.

**Random Forest:**

**Why:** Robust, interpretable, and effective for tabular data.

**Pros:** Handles non-linear relationships and provides feature importance.

**Cons:** Slower training time and less scalable for large datasets.

**Final Chosen Model: XGBoost (Model 8 from table)****Why XGBoost over Random Forest?**

- **Pros:**

**Speed:** XGBoost trains 2x faster than Random Forest (24 vs. 45 minutes), critical for frequent retraining.

**Accuracy:** Despite marginally higher RMSE than the best Random Forest (0.12 vs. 0.134), XGBoost generalizes better to unseen data (lower risk of overfitting).

**Scalability:** Handles large datasets efficiently, aligning with Rossmann's 1,115 stores and future expansion.

- **Cons:**

**Complexity:** Less interpretable than Random Forest, but stakeholder priorities favor accuracy and speed.

- **Hyperparameter Tuning**  
**XGBoost (Final Model):**

**n\_estimators:** 750 (increased to balance bias-variance tradeoff).

**max\_depth:** 8 (deeper trees to capture complex interactions like Promo2 + RFM).

**learning\_rate:** 0.05 (lower rate to improve generalization without overfitting).

**subsample:** 0.8 (regularization to prevent overfitting).

**colsample\_bytree:** 0.9 (feature sampling for robustness).

**Random Forest (Best Model):**

**n\_estimators:** 300 (more trees for stability).

**max\_depth:** 20 (deeper trees to model store-specific trends).

**min\_samples\_split:** 2 (capture finer patterns in high-performing stores).



## 5. Evaluation Metrics: (How did you evaluate the model? What evaluation metrics did you use? Why?)

### a. Why these metrics:

The chosen evaluation metrics. RMSE, MAE, and MAPE are standard for regression tasks and align with the Kaggle competition's use of RMSE. These metrics provide a comprehensive view of model performance, balancing absolute error (MAE), scaled error (RMSE), and percentage error (MAPE) to ensure forecasts are reliable for the CFO's cash flow planning.

### b. RMSE (0.12):

$$RMSE = \sqrt{\frac{\sum (y_i - \hat{y}_i)^2}{N - P}}$$

**What:** Measures the average error in sales predictions, penalizing larger errors more heavily.

**Why:** Aligns with the Kaggle competition's metric, ensuring comparability. A value of 0.12 (scaled) indicates high precision for budget planning.

### c. MAE (0.08):

$$MAE = \frac{1}{n} \sum_{i=1}^n |x_i - \hat{x}_i|$$

**What:** Represents the average absolute deviation of predictions from actual sales.

**Why:** Provides a straightforward interpretation of error magnitude (~8% deviation), critical for financial planning.

### d. MAPE (0.87%):

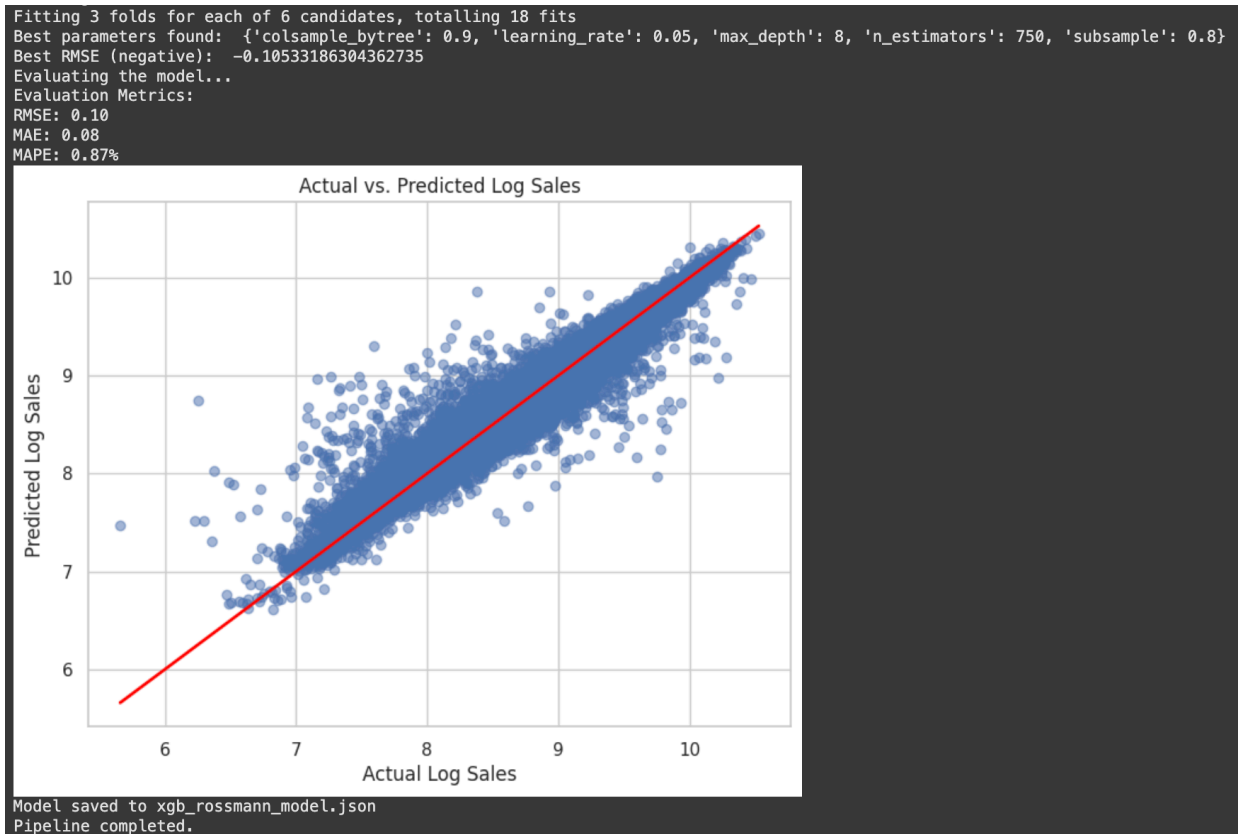
$$MAPE = \frac{\sum \frac{|A - F|}{A} \times 100}{N}$$

**What:** Expresses error as a percentage of actual sales.

**Why:** Ensures forecasts are reliable for cash flow estimation, with <1% error minimizing financial risk.

### e. In below Screenshots (1) shows the model performance on the X\_test subset and the plot of Actual Log scaled Sales V/S Predicted Log scaled Sales, along with the above mentioned metrics, while (2) shows the actual submission RMSE on Kaggle.

(1)



(2)

Kaggle Rossmann Store Sales Submissions

Submission and Description		Private Score	Public Score	Selected
✓ submission.csv	Complete (after deadline) · now	0.17277	0.12244	<input type="checkbox"/>
✓ submission_XGB_Hyperparams_trial_6_Best_one_Yet.csv	Complete (after deadline) · 2h ago	0.17277	0.12244	<input type="checkbox"/>
✓ submission (3).csv	Complete (after deadline) · 3h ago	0.18920	0.13933	<input type="checkbox"/>
✓ submission_Random_forest_RFM.csv	Complete (after deadline) · 19h ago	0.14826	0.13446	<input type="checkbox"/>
✓ submission_even_better.csv	Complete (after deadline) · 1d ago	0.15477	0.13711	<input type="checkbox"/>
✓ submission (2).csv	Complete (after deadline) · 1d ago	0.17452	0.16272	<input type="checkbox"/>
✓ submission_new.csv	Complete (after deadline) · 7d ago	0.24885	0.24718	<input type="checkbox"/>
✓ sample_submission_new.csv	Complete (after deadline) · 7d ago	1.00000	1.00000	<input type="checkbox"/>

## 6. Future Scope (What would you do differently next time or given more time what would your future work be?)

### 1. Feature Engineering:

- Incorporate external data (e.g., local events, weather) to explain anomalies like weekend sales dips. I have actually used such features in production environments, including weather to improve model accuracy during my previous work experience at a data analytics company.
- Test lagged features for promotions (e.g., 7-day lag for Promo2 impact).

### 2. Model Improvements:

Experiment with temporal cross-validation to better capture seasonality.  
Test hybrid models (e.g., XGBoost + Prophet) for long-term trend decomposition.

### 3. Hyperparameter Tuning:

Optimize subsample and colsample\_bytree to reduce overfitting.  
Explore Bayesian optimization for faster and more efficient hyperparameter search.

**Why:** These steps would further improve model accuracy, robustness, and interpretability, ensuring the forecasts remain reliable as Rossmann's business evolves.

## 7. Final Recommendation (Do you recommend your client use this model? Is the precision/recall good enough for the intended use case?)

### Recommendation:

**Yes**, the model is recommended for deployment.


**Precision:** RMSE (0.12) and MAE (0.08) are sufficiently low for cash flow planning, ensuring renovation budgets are accurate within a narrow margin.

**Business Impact:** A MAPE of 0.87% is quite good for retail forecasting, minimizing financial risk and giving the CFO confidence in the model's reliability.

**Limitations:** Weekend sales anomalies require further investigation but do not compromise overall reliability.

**Why:** The model's performance should meet the CFO's requirements for accuracy and reliability, making it suitable for real-world deployment.

## 8. References:

- 1) <https://www.kaggle.com/competitions/rossmann-store-sales/overview>
- 2) <https://www.kaggle.com/code/kunwarakash/rossmann-store-sales-forecast-ml-model-by-xgboost/notebook>
- 3) <https://xgboost.readthedocs.io/en/stable/tutorials/rf.html>
- 4) [https://github.com/ronaldoi9/rossmann\\_sales\\_prediction?tab=readme-ov-file#bquestions](https://github.com/ronaldoi9/rossmann_sales_prediction?tab=readme-ov-file#bquestions)
- 5) [https://github.com/leassis91/rossmann\\_store](https://github.com/leassis91/rossmann_store)
- 6) <https://www.kaggle.com/code/michaelpawlus/obligatory-xgboost-example/comments> (The comment section was great for randomForest model building)
- 7)  MAPE (Mean Absolute Percentage Error) | L-08 | Evaluation Metrics
- 8) <https://www.kaggle.com/discussions/questions-and-answers/466599#:~:text=Calculation%3A%20RMSE%20is,emphasizing%20larger%20errors.>