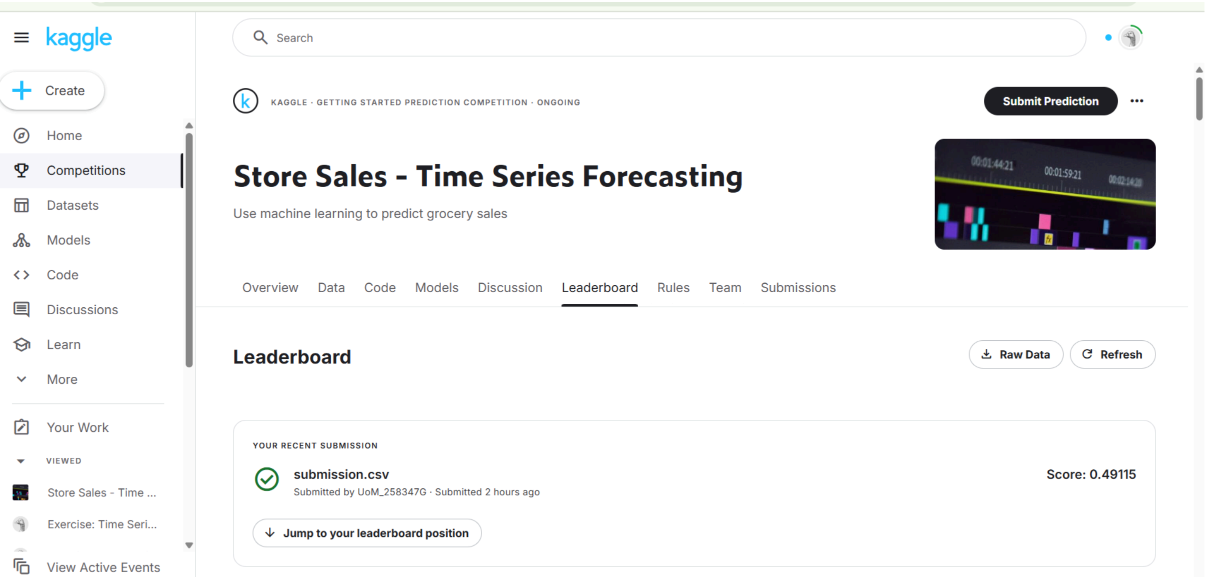
Leaderboard score



KAGGLE · GETTING STARTED PREDICTION COMPETITION · ONGOING

Store Sales - Time Series Forecasting

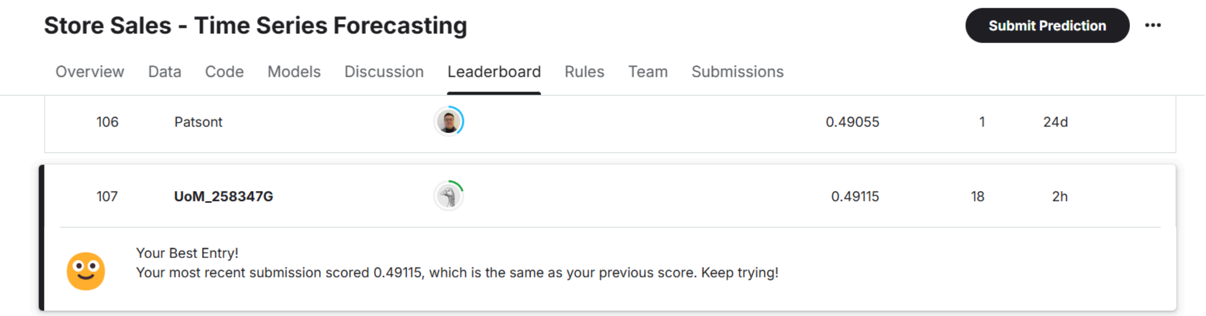
Use machine learning to predict grocery sales

Leaderboard

YOUR RECENT SUBMISSION

submission.csv
Submitted by UoM_258347G · Submitted 2 hours ago

Score: 0.49115

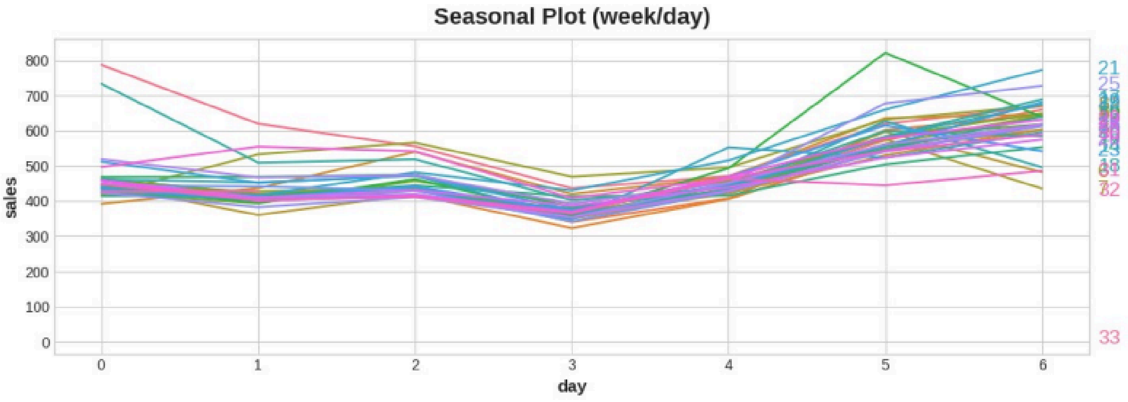


Store Sales - Time Series Forecasting

106	Patsont	0.49055	1	24d
107	UoM_258347G	0.49115	18	2h

Your Best Entry!
Your most recent submission scored 0.49115, which is the same as your previous score. Keep trying!

Exploratory Analysis

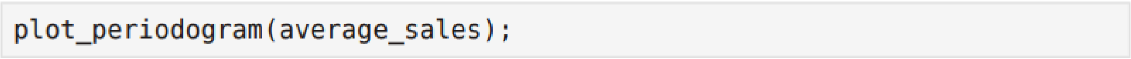


As seen in the plot, sales fluctuate predictably within the week, indicating strong weekly seasonality.

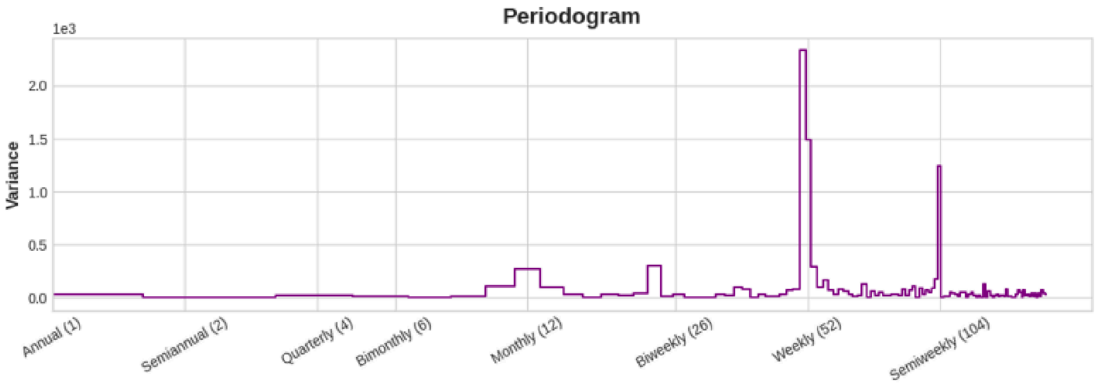
The seasonal plot reveals weekly patterns in sales data. This suggests higher sales activities toward the end of the week.

There are clear outliers with certain weeks exhibiting significantly higher peaks or deeper drops. This could be linked to promotions, holidays, or special events.

The spread of sales across weeks indicates some variance, possibly influenced by seasonal patterns, external factors like holidays, or regional sales differences.



```
plot_periodogram(average_sales);
```

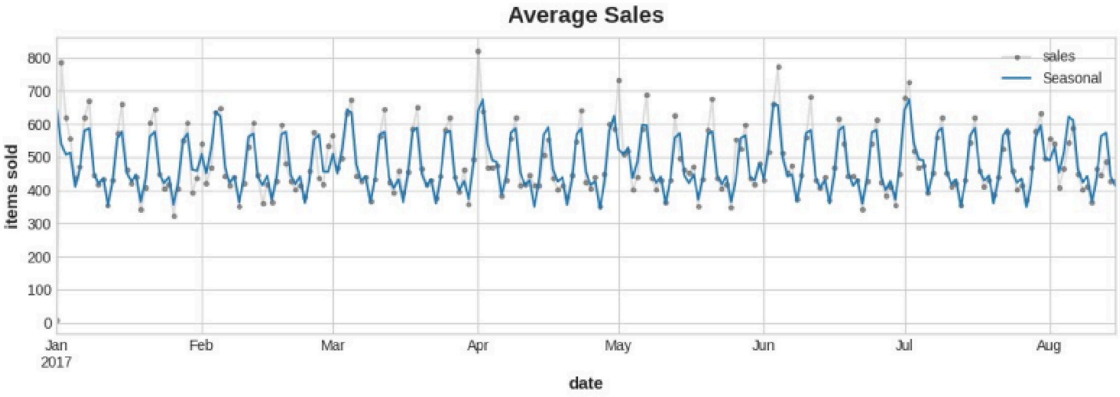


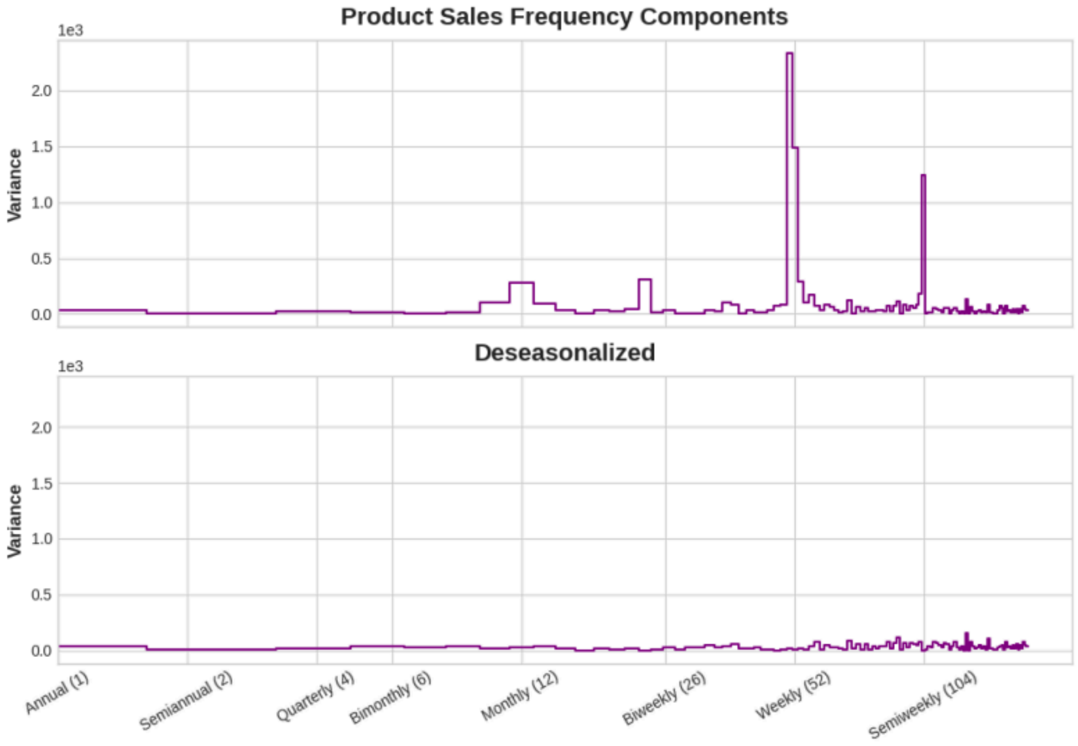
The strongest peak is observed at Weekly (52), confirming strong weekly seasonality in the sales data.

Another significant peak appears at Semiweekly (104), indicating potential bi-weekly patterns, possibly linked to payday effects or recurring promotional campaigns.

There are smaller peaks around Monthly (12) and Biweekly (26), suggesting mild monthly patterns.

Minimal variance is seen in longer-term cycles like Annual (1) or Semiannual (2), indicating that annual trends may not play a strong role in this dataset.

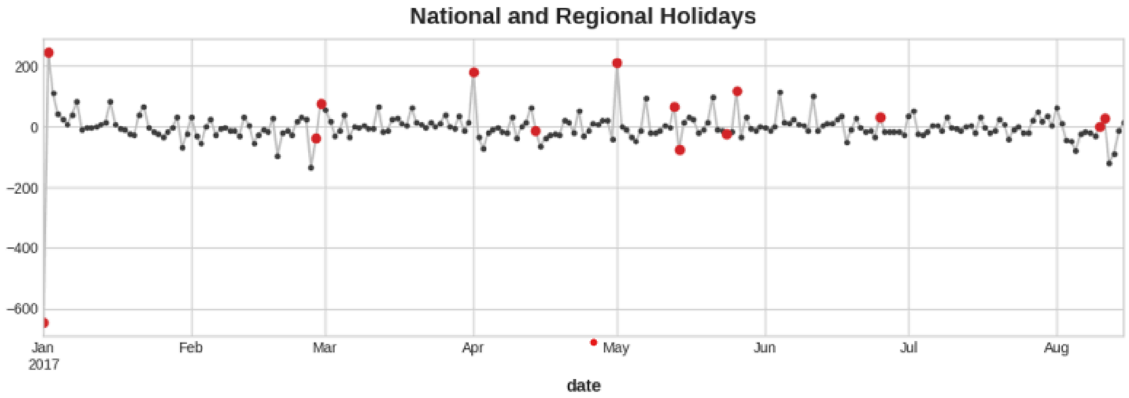




The original data has strong weekly and semiweekly seasonality that should be modeled with Fourier terms or seasonal indicators.

The successful reduction of these peaks in the deseasonalized series indicates that the applied seasonality modeling effectively captured those patterns.

Future modeling efforts should focus on enhancing the model with additional features for promotions, holidays, and potential trend changes.



Significant spikes and drops are evident on these holidays, confirming their substantial impact on sales.

Some holidays correlate with noticeable positive spikes, likely driven by increased consumer demand (e.g., festive shopping, payday effects).

Conversely, certain holidays coincide with sharp declines, possibly linked to store closures, reduced business hours, or shifts in customer behavior.

Creating binary indicators for key holidays (e.g., major national holidays, paydays) can enhance the model's predictive power.

Solution Overview

This section outlines three distinct approaches developed to forecast store sales in the Kaggle Store Sales Time Series Forecasting competition. Each submission builds upon different feature engineering strategies and evaluates performance using RMSLE (the competition metric), MAE, and R² Score on a validation set (Aug 1–15, 2017, with training from Jan 1–Jul 31, 2017). Among these, the second submission proved most accurate and achieved the highest leaderboard ranking, making it the final submission.

Approch 1: Baseline Trend and Seasonality Model

- **Model Selection**
  - Ridge regression with alpha=1.0.
  - A regularized linear model to prevent overfitting.
- **Data Preparation**
  - Set multi-index (store_nbr, family, date), unstacked y for 2017 sales, preserved test id column.
  - Structured data for multi-target time series modeling; preserved test IDs for submission completeness.
- **Feature Engineering**
  - Linear trend (order=1), weekly seasonality (seasonal=True), monthly Fourier terms (order=4, freq='M'), New Year indicator (X['NewYear']).
  - Captures core time series patterns (trend, weekly/monthly cycles, New Year anomaly) with a simple, focused feature set.
- **Training and Evaluation**
  - Trained on X_train, y_train; evaluated on X_val, y_val with RMSLE, MAE, R². Refit on full 2017 data for submission.
- **Test Prediction**
  - Generated X_test with matching features, predicted, merged with test IDs, filled NaNs with 0.
  - Ensures feature consistency and valid submission format.
- **Validation Metrics**
  - RMSLE: 0.5092
  - MAE: 81.9599
  - R² Score: 0.9493
- **Assessment**: This baseline approach achieved decent performance, leveraging trend and seasonality. However, its simplicity limits accuracy compared to more feature-rich models.

Approach 2 (Final Submission): Enhanced Feature Engineering with External Drivers

- **Model Selection**
  - Ridge regression with alpha=1.0
- **Feature Engineering**

        - Linear trend (order=1).
        - Weekly seasonality (seasonal=True) and monthly Fourier terms (order=4, freq='ME').
        - New Year indicator (X['NewYear']).
        - Holiday dummies from national/regional events (e.g., Primer Grito de Independencia).
        - Oil prices (dcoilwtico) as a continuous feature.
        - No lagged sales included.
    - **This**:
        - Captures long-term trends, weekly/monthly cycles, and holiday effects for comprehensive temporal modeling.
        - Oil prices reflect economic influences on consumer behavior in Ecuador.
        - New Year indicator addresses a significant sales spike.
        - Omitting lagged sales simplifies the model while maintaining strong predictive power.
- **Train-Validation Split**
    - Training: Jan 1–Jul 31, 2017; Validation: Aug 1–Aug 15, 2017.
- **Training and Evaluation**
    - Trained on X_train, y_train; evaluated on X_val, y_val with RMSLE, MAE, R^2. Refit on full 2017 data for submission.
    - Ensures competition-relevant accuracy (RMSLE) with additional insights (MAE, R^2).
- **Test Prediction**
    - Generated X_test with matching features (trend, seasonality, New Year, holidays, oil), predicted, merged with test IDs, filled NaNs with 0.
- **Validation Metrics**
    - RMSLE: 0.4911
    - MAE: 77.6624
    - R^2 Score: 0.9545
    - This submission outperformed others with an RMSLE of 0.4911, MAE of 77.6624, and R^2 of 0.9545, demonstrating better accuracy across all metrics. The inclusion of holidays and oil prices enhanced predictive power, capturing key external influences on sales. After trying to add lag values (as in 3rd approach) based on the results obtained, this was selected as the final submission.

Approach 3: Extended Model with Lagged Sales

- **Feature Engineering**
        - Linear trend (order=1).
        - Weekly seasonality (seasonal=True) and monthly Fourier terms (order=4, freq='ME').
        - New Year indicator (X['NewYear']).
        - Holiday dummies from national/regional events.

        - Oil prices (dcoilwtico).
        - Lagged sales (1-day, 2-day, 7-day averages across all stores/families).
        - Adds autoregressive power via lagged sales to capture short-term trends.
- **Validation Metrics**
    - RMSLE: 0.5549
    - MAE: 84.4390
    - R² Score: 0.9472
- **Assessment**:
    - Despite adding lagged sales for short-term trend capture, this submission underperformed Approach 2.

Note: Approach A (no lags) unexpectedly outperforms Approach (with lags) in accuracy. The lagged sales feature, while theoretically sound, didn't enhance accuracy here, likely due to its average-based implementation. Maybe adding store specific lags improves the performance which can be tried in next steps.

Data cleaning and preprocessing steps in final best solution

Data Loading and Type changing

- Loaded train.csv with only store_nbr, family, date, and sales, using category types for store_nbr and family, float32 for sales, and parsed date as datetime to reduce memory usage.
- Loaded holidays_events.csv with category types for type, locale, locale_name, description, and bool for transferred, parsing date as datetime for efficient memory use.
- Loaded oil.csv, parsed date as datetime, and set it as a period index ('D') to prepare for time series alignment.
- Loaded test.csv with category types for store_nbr and family, parsed date as datetime, and preserved the id column separately for submission integrity.

Date Formatting and Indexing

- Converted date in train.csv to daily period format ('D') and set a sorted multi-index (store_nbr, family, date) for structured store-family-specific sales data.
- Converted holiday date to period format ('D') to align with training and test data.
- Converted test date to period format ('D') and set a sorted multi-index (store_nbr, family, date) to match training data structure.

Null Value Removal and Imputation

- Filled missing dcoilwtico values in oil.csv with forward fill then backward fill to ensure a complete economic indicator series.
- Filled NaNs in holiday dummies joined to X with 0 to assume no effect on non-holiday days.
- Applied forward-backward fill to oil prices (dcoilwtico) after joining to X to eliminate NaNs and maintain continuity.
- Filled missing columns in X_test with 0 to ensure feature consistency with training data.
- Filled NaN sales in y_submit with 0 after merging with test IDs to handle unmatched rows for submission.
- Verified no NaNs remained in X and X_test to prevent model errors during training and prediction.

Data Transformation and Structuring

- Unstacked sales from store_sales into a wide-format target y with store-family columns, filtered to 2017, for multi-target regression aligned with the test period.
- Filtered holidays_events for national and regional holidays in 2017 up to Aug 31, kept only description, and removed unused categories to focus on relevant events.
- Created base training features (X) with a linear trend, weekly seasonality, and monthly Fourier terms (order 4) using DeterministicProcess to capture core time series patterns.
- Generated test features (X_test) with trend, seasonality, and matching additional features using dp.out_of_sample for prediction consistency.

Feature Engineering and Enhancement

- Added a New Year indicator to X and X_test based on dayofyear == 1 to flag a significant sales event.
- Generated holiday dummies from filtered holidays and joined them to X and X_test to encode event impacts.
- Joined oil prices (dcoilwtico) to X and X_test as a continuous economic feature influencing sales.
- Aligned X_test column order with X to ensure model compatibility.

Additional Evaluation Metrics Suitable for Validation Data

RMSLE, MAE, R^2 were used. Someother evaluation metrics suitable for validation data can be: MAPE, MedAE
Mean Absolute Percentage Error (MAPE):

- Measures relative error in percentage terms, complementing RMSLE’s logarithmic focus and MAE’s absolute focus.
- Can highlights prediction accuracy across diverse sales scales

Median Absolute Error (MedAE):

- Robust to outliers, unlike MAE, which averages all errors.

These metrics can enhance the evaluation by addressing relative errors (MAPE) and robustness (MedAE).

Alternative Solution

Hybrid Model with Store-Family-Specific Lags and Gradient Boosting

- Replace Ridge regression with a gradient boosting model (e.g., XGBoost or LightGBM) to capture non-linear patterns. Because, gradient boosting excels at modeling complex, non-linear relationships and interactions (e.g., holiday effects varying by family).
- Add store-family-specific lagged sales (e.g., 1-day, 7-day lags per store-family pair) instead of averages, alongside existing features (trend, seasonality, holidays, oil prices, New Year).

Try prophet model

Issues with Submission (the best one among all our submissions)

- Lack of Autoregressive Features:
  - Approach 2 omits lagged sales, missing short-term dependencies. This likely limits its ability to capture abrupt changes in the test period.
  - Approach 3 (with lags) aimed to address this but underperformed, suggesting potential if implemented better.
  - Improvement:
    Add store-family-specific lags (e.g., y.shift(1) per target) instead of averages. Fill NaNs with family-specific means or rolling averages to maintain data integrity.
- Limited Non-Linearity:

  - Ridge regression assumes linear relationships, potentially underfitting complex interactions (e.g., holiday effects varying by store or family).
  - Improvement:
    - Use a tree-based model (e.g., LightGBM) to capture non-linear patterns and interactions.
    - Tune hyperparameters to optimize RMSLE.
- Potential Over-Simplification of Holidays:
  - Holiday dummies treat all events uniformly (0 or 1), ignoring varying impacts (e.g., Navidad vs. minor regional holidays).
  - Improvement:
    - Weight holidays by historical sales impact (e.g., average sales lift per event from 2016 data) instead of binary encoding.
- Fixed Regularization Parameter:
  - Problem: Ridge uses a fixed alpha=1.0, which may not be optimal for all store-family combinations.
  - Strong R^2 but potential overfitting or underfitting for specific targets is showing that.
  - Improvement:
    - Perform grid search on alpha using cross-validation on the training set to minimize RMSLE.