Milestone 2: Feature Engineering Summary

Executive Summary

This document summarizes all feature engineering performed across Milestone 1 and Milestone 2, documenting 91 total features with their expected impact on forecasting models. Features are categorized by type, creation methodology, and predictive value based on correlation analysis and domain expertise.

Table of Contents

---

1. Feature Overview

1.1 Summary Statistics

Category	Milestone 1	Milestone 2	Total
Original Raw Features	11	0	11
Time-Based Features	20	9	29
Lag/Rolling Features	7	15	22
Categorical Encodings	3	0	3
Normalized Features	17	0	17
Store/Dept Features	0	11	11
Promotional Features	0	4	4
Economic Interactions	0	4	4
Time Aggregations	0	5	5
TOTAL	**49**	**42**	**91**

1.2 Feature Distribution by Predictive Power

Strength	Count	Examples
Very Strong ($r > 0.90$)	5	Sales_Rolling_Mean_4, Sales_Lag1
Strong ($0.50 < r < 0.90$)	1	Sales_Rolling_Std_4
Moderate ($0.30 < r < 0.50$)	3	Size, Store_Avg_Sales
Weak ($0.10 < r < 0.30$)	8	MarkDowns, Season indicators
Very Weak ($r < 0.10$)	74	Most binary flags, interactions

---

2. Milestone 1 Features (Baseline)

2.1 Original Raw Features (11)

#	Feature	Type	Description	Missing Values
1	Store	Categorical	Store ID (1-45)	0%
2	Dept	Categorical	Department ID (1-99)	0%
3	Date	Datetime	Week ending date	0%
4	Weekly_Sales	Numerical	Target variable (train only)	0%
5	IsHoliday	Binary	Holiday week indicator	0%
6	Temperature	Numerical	Avg temperature (°F)	0%
7	Fuel_Price	Numerical	Regional fuel price	0%
8	MarkDown1-5	Numerical	Promotional markdowns	Filled with 0
9	CPI	Numerical	Consumer Price Index	0%
10	Unemployment	Numerical	Regional unemployment rate	0%
11	Size	Numerical	Store square footage	0%

2.2 Time-Based Features (20)

Basic Temporal Features (7):

Feature	Type	Range	Purpose
Year	Integer	2010-2012	Long-term trends
Month	Integer	1-12	Monthly seasonality
Day	Integer	1-31	Day-of-month patterns
Quarter	Integer	1-4	Quarterly patterns
DayOfWeek	Integer	0-6 (Mon-Sun)	Weekly patterns
WeekOfYear	Integer	1-53	Week-based seasonality
Is_Weekend	Binary	0-1	Weekend effect

Binary Temporal Flags (7):

Feature	Description	Frequency
Is_Month_Start	First day of month	~12/year
Is_Month_End	Last day of month	~12/year
Is_Quarter_Start	First day of quarter	4/year
Is_Quarter_End	Last day of quarter	4/year
Is_Year_Start	First day of year	1/year
Is_Year_End	Last day of year	1/year

Cyclical Features (6):

Feature	Transformation	Purpose
Month_Sin	$\sin(2\pi \times \text{Month}/12)$	Circular monthly encoding
Month_Cos	$\cos(2\pi \times \text{Month}/12)$	Prevents Dec→Jan discontinuity
Week_Sin	$\sin(2\pi \times \text{Week}/52)$	Circular weekly encoding
Week_Cos	$\cos(2\pi \times \text{Week}/52)$	Annual cycle continuity

Feature	Transformation	Purpose
DayOfWeek_Sin	$\sin(2\pi \times Day/7)$	Circular day encoding
DayOfWeek_Cos	$\cos(2\pi \times Day/7)$	Sun→Mon continuity

Expected Impact: HIGH - Essential for capturing seasonality and time-based patterns

2.3 Lag and Rolling Features (7)

Feature	Window	Calculation	Correlation	Expected Impact
Sales_Lag1	1 week	Previous week's sales	+0.9438	CRITICAL
Sales_Lag2	2 weeks	Sales from 2 weeks ago	+0.9260	CRITICAL
Sales_Lag4	4 weeks	Sales from 4 weeks ago	+0.9135	CRITICAL
Sales_Rolling_Mean_4	4 weeks	Moving average	+0.9758	CRITICAL
Sales_Rolling_Mean_8	8 weeks	Moving average	+0.9648	CRITICAL
Sales_Rolling_Std_4	4 weeks	Rolling std deviation	+0.4834	HIGH
Sales_Momentum	-	Lag1 - Lag2 (velocity)	Moderate	MODERATE

Calculation Level: Per Store-Department combination
Missing Values: Forward-filled for first few weeks
Expected Impact: CRITICAL - Strongest predictors identified

2.4 Categorical Encoding (3)

One-Hot Encoding for Store Type:

Feature	Values	Count	Purpose
Type_A	0/1	22 stores	Large supercenters
Type_B	0/1	17 stores	Mid-size stores
Type_C	0/1	6 stores	Small format stores

Expected Impact: MODERATE - Captures store format differences

2.5 Normalized Features (17)

Method: Z-score normalization

Normalized Features: - Size, Temperature, Fuel_Price, CPI, Unemployment - MarkDown1-5 (after filling) - Sales_Lag1-4, Sales_Rolling_Mean_4/8, Sales_Rolling_Std_4

Parameters Saved: normalization_params.json (train statistics only)

Expected Impact: HIGH - Essential for gradient-based models, helps prevent feature domination

---

3. Milestone 2 Enhanced Features

3.1 Advanced Rolling Statistics (9)

Feature	Type	Window	Description	Expected Impact
Sales_EMA_4	EMA	4 weeks	Exponential moving avg	HIGH
Sales_EMA_8	EMA	8 weeks	Exponential moving avg	HIGH
Sales_EMA_12	EMA	12 weeks	Exponential moving avg	HIGH

Feature	Type	Window	Description	Expected Impact
Sales_Rolling_Min_4	Min	4 weeks	Minimum sales in window	MODERATE
Sales_Rolling_Max_4	Max	4 weeks	Maximum sales in window	MODERATE
Sales_Rolling_Range_4	Range	4 weeks	Max - Min	MODERATE
Sales_Trend	Trend	-	EMA_4 - EMA_12	HIGH
Sales_CV_4	Volatility	4 weeks	Coefficient of variation	MODERATE
Sales_Acceleration	Derivative	-	Change in momentum	LOW

Rationale: - **EMAs** give more weight to recent observations → better for trending data - **Min/Max/Range** capture volatility and price boundaries - **Trend** identifies upward/downward movement - **CV** normalizes volatility by mean (scale-invariant) - **Acceleration** captures rate of change in trend

3.2 Seasonal Features (9)

Feature	Type	Description	Expected Impact
Is_Holiday_Season	Binary	Nov-Dec flag	HIGH
Is_BackToSchool_Season	Binary	Jul-Aug flag	HIGH
Is_SuperBowl_Week	Binary	Early Feb flag	MODERATE
Days_To_Thanksgiving	Integer	Days until Thanksgiving	HIGH
Days_To_Christmas	Integer	Days until Christmas	HIGH
Season_Winter	Binary	Dec-Feb	MODERATE
Season_Spring	Binary	Mar-May	LOW
Season_Summer	Binary	Jun-Aug	MODERATE
Season_Fall	Binary	Sep-Nov	HIGH

Rationale: - **Holiday Season** captures Black Friday/Christmas surge - **Back to School** targets department-specific spikes (clothing, electronics) - **Days_To_Christmas** captures pre-holiday ramp-up (non-linear relationship) - **Meteorological Seasons** align with retail calendar

3.3 Store Performance Features (11)

Feature	Granularity	Statistic	Expected Impact
Store_Avg_Sales	Store	Mean	HIGH
Store_Std_Sales	Store	Std Dev	MODERATE
Store_Min_Sales	Store	Min	LOW
Store_Max_Sales	Store	Max	LOW
Dept_Avg_Sales	Department	Mean	HIGH
Dept_Std_Sales	Department	Std Dev	MODERATE
StoreDept_Avg_Sales	Store-Dept	Mean	HIGH
StoreDept_Std_Sales	Store-Dept	Std Dev	MODERATE
Sales_Deviation_From_Store_Avg	Store	Deviation	MODERATE
Sales_Deviation_From_Dept_Avg	Department	Deviation	MODERATE
Sales_Deviation_From_StoreDept_Avg	Store-Dept	Deviation	HIGH

Rationale: - **Hierarchical Statistics** provide context at store, department, and store-dept levels - **Deviations** help identify anomalies and outliers - **Store-Dept Averages** are highly predictive (store+dept baseline)

Note: Calculated from training data only (to prevent data leakage)

3.4 Promotional Intensity Metrics (4)

Feature	Formula	Description	Expected Impact
Total_MarkDown	Sum(MarkDown1-5)	Total promotional discount	HIGH
Num_Active_MarkDowns	Sum(Has_MarkDownX)	Number of active promotions	MODERATE
Promo_Intensity	Total_MarkDown / Size	Promotion density	MODERATE
Total_MarkDown_Rolling_4	Rolling(Total_MarkDown, 4)	Recent promo trend	MODERATE

Rationale: - **Total_MarkDown** aggregates all promotional activity - **Num_Active_MarkDowns** captures breadth of promotions - **Promo_Intensity** normalizes by store size - **Rolling average** captures sustained promotional periods

3.5 Economic Indicator Interactions (4)

Feature	Formula	Interpretation	Expected Impact
Economic_Stress	CPI × Unemployment	Consumer hardship indicator	MODERATE
Holiday_Temperature	Temperature × IsHoliday	Holiday weather effect	LOW
Spending_Power	Fuel_Price × Unemployment	Disposable income proxy	LOW
Store_Purchasing_Power	Size × CPI	Store-level economic context	LOW

Rationale: - Individual economic indicators show weak correlations (< 0.03) - **Interaction features** may capture non-linear relationships - Especially valuable for tree-based models (XGBoost, Random Forest)

3.6 Time-Based Aggregations (5)

Feature	Granularity	Aggregation	Expected Impact
Month_Store_Avg_Sales	Store-Month	Mean	HIGH
Month_Store_Total_Sales	Store-Month	Sum	MODERATE
Quarter_Store_Avg_Sales	Store-Quarter	Mean	HIGH
Quarter_Store_Total_Sales	Store-Quarter	Sum	MODERATE
Store_Sales_YoY_Growth	Year-over-Year	Pct Change	MODERATE

Rationale: - **Monthly/Quarterly aggregations** smooth out weekly noise - **YoY Growth** captures long-term trends - Helpful for models that benefit from multi-scale temporal features

---

4. Feature Impact Assessment

4.1 Critical Features (Top Priority)

Must-have for any model:

Feature	Correlation	Rationale
Sales_Rolling_Mean_4	+0.9758	Strongest single predictor
Sales_Rolling_Mean_8	+0.9648	Captures medium-term trends
Sales_Lag1	+0.9438	Immediate past dependency
Sales_Lag2	+0.9260	2-week lookback
Sales_Lag4	+0.9135	Monthly cycle (4-week lag)
StoreDept_Avg_Sales	High	Store-Dept baseline

Expected Model Impact: CRITICAL
Removal Risk: Model performance would degrade by **30-50%**

4.2 High-Value Features

Strongly recommended:

Feature Group	Count	Impact
Time-Based (Month, Quarter, Cyclical)	13	HIGH
Seasonal Indicators (Holiday Season, Days_To_Christmas)	5	HIGH
EMAs (4, 8, 12)	3	HIGH
Store/Dept Statistics	6	HIGH
Promotional Features (Total_MarkDown)	2	HIGH

Expected Model Impact: HIGH
Removal Risk: Model performance would degrade by **10-20%**

4.3 Moderate-Value Features

Useful for ensemble models:

Feature Group	Count	Impact
Rolling Min/Max/Range	3	MODERATE
Sales Deviations	3	MODERATE
Economic Interactions	4	MODERATE
Secondary Seasonal Flags	4	MODERATE

Expected Model Impact: (Moderate)
Removal Risk: Model performance would degrade by **2-5%**

4.4 Experimental Features

Test in advanced models:

Feature Group	Count	Impact
Individual Economic Indicators	4	Low-Moderate
Sales_Acceleration	1	Low-Moderate
Promo_Intensity	1	Low-Moderate
Temperature (raw)	1	Low-Moderate

Expected Model Impact: (Low-Moderate)
Removal Risk: Minimal (< 2%)

---

5. Feature Selection Recommendations

5.1 Baseline Model (Simple)

Use for: ARIMA, Prophet, Linear Regression

Features (15): - Lag features: Lag1, Lag2, Lag4 - Rolling means: Rolling_Mean_4, Rolling_Mean_8 - Time: Month, Quarter, Year - Seasonal: Is_Holiday_Season, Days_To_Christmas - Store: Store_Avg_Sales, StoreDept_Avg_Sales - Categorical: Type_A, Type_B, Type_C

Expected R^2: 0.85-0.90

5.2 Intermediate Model (Tree-Based)

Use for: Random Forest, XGBoost

Features (40): - All Baseline features - Additional lags: All EMAs - All seasonal indicators - Store/Dept statistics (all) - Promotional features - Economic interactions - Time aggregations

Expected R^2: 0.92-0.95

5.3 Advanced Model (Deep Learning)

Use for: LSTM, Transformer

Features (All 91): - Include all engineered features - LSTM will learn feature importance - Use attention mechanism to weight features - Benefit from redundancy and interactions

Expected R^2: 0.94-0.97

5.4 Feature Elimination Strategy

Step 1: Remove features with VIF > 10 (multicollinearity)
Step 2: Use SHAP values or permutation importance
Step 3: Iterative backward elimination (remove lowest importance)
Step 4: Cross-validation to confirm impact

---

6. Implementation Details

6.1 Data Consistency

Train-Test Alignment: - All features calculated identically for train and test - Normalization uses **train statistics only** (no data leakage) - Store/Dept statistics calculated from train only - Lag features properly handled at train-test boundary

6.2 Memory Optimization

Techniques Applied: - Manual one-hot encoding (vs pd.get_dummies) - In-place operations where possible - Chunked processing for large datasets - Efficient data types (int8 for binary, float32 for numerical)

Result: Successfully processed 421K rows × 91 features without memory errors

6.3 Code Organization

Script	Purpose	Output
step_1_3_1_time_features.py	Time-based features	Stage1.3.1
step_1_3_2_lag_features.py	Lag/rolling features	Stage1.3.2
step_1_3_3_encode_categorical.py	One-hot encoding	Stage1.3.3
step_1_3_4_normalize_features_final.py	Z-score normalization	Final (M1)
step_2_2_feature_engineering.py	Enhanced features	Stage2/enhanced_features

6.4 Feature Documentation

Metadata Saved: - feature_summary.json: Complete feature list with categories - normalization_params.json: Mean/std for each normalized feature - correlation_matrix.csv: Full correlation matrix - sales_correlations.csv: Sorted correlations with Weekly_Sales

---

7. Expected Impact on Model Performance

7.1 Baseline (No Feature Engineering)

Features: Raw data only (11 features)
Expected WMAE: ~$4000-5000
Expected R²: 0.70-0.75

7.2 Milestone 1 Features

Features: + Time + Lag + Encoding + Normalization (49 features)
Expected WMAE: ~$2500-3000
Expected R²: 0.85-0.90
Improvement: 30-40% reduction in error

7.3 Milestone 2 Enhanced Features

Features: + Advanced rolling + Seasonal + Store stats + Interactions (91 features)
Expected WMAE: ~$1500-2000
Expected R²: 0.92-0.95
Improvement: 50-60% reduction in error vs baseline

---

8. Conclusion

8.1 Summary

Total Features Created: 91
Critical Features: 6 (lag + rolling means)
High-Value Features: 29 (time + seasonal + EMAs)
Moderate-Value Features: 30 (interactions + statistics)
Experimental Features: 26 (to test in advanced models)

8.2 Key Achievements

- **Comprehensive Feature Set:** Covers temporal, lag, categorical, and interaction features
- **No Data Leakage:** Strict train-test separation maintained
- **Scalable Pipeline:** Automated feature generation for both datasets
- **Memory Efficient:** Handled large dataset without errors
- **Well-Documented:** Detailed rationale for each feature category

8.3 Next Steps

1. **Feature Selection:** Use SHAP/permutation importance in model training

2. **Model Development:** Test baseline → intermediate → advanced models
3. **Hyperparameter Tuning:** Optimize models with best feature sets
4. **Ensemble:** Combine models for robust forecasting
5. **Deployment:** Select final feature set for production