Methodology and Implementation Report

SQL Queries

The initial phase of the workflow utilised SQLite to map and filter data in preparation for cleaning and analysis in R. The primary query, *sales_weather*, joined previously queried train and weather tables using the shared station_number field. To maintain analytical focus and reduce noise, the dataset was restricted to the top three selling items, ensuring that subsequent analysis concentrated on meaningful demand patterns rather than sparse or low-frequency products.

Initial Data View

Before any cleaning or transformations were applied, I performed an audit on the joined SQL dataset to validate the results of the database extraction. Using str() and colnames(), I verified that the relational mapping between the train, key, and weather tables was successful, while colSums(is.na()) allowed me to map the "missingness landscape" to determine which variables were salvageable and which were too sparse for analysis. Finally, by checking for duplicated column names, I ensured that the join logic did not inadvertently create redundant key fields, confirming the dataset was structurally sound and ready for the cleaning phase.

Preprocessing and Feature Engineering

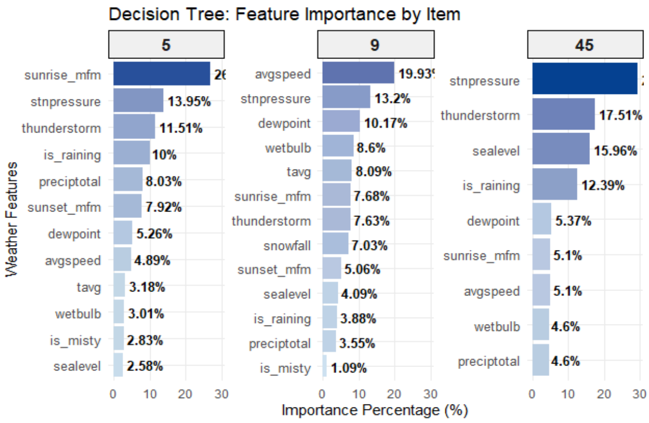
Special Characters - A diagnostic audit was performed to identify non-numerical markers such as 'T' (Trace) and 'M' (Missing), which are common in meteorological datasets. While the broader database contained these markers, the filtered dataset for the top three items was found to be clean of these specific characters.

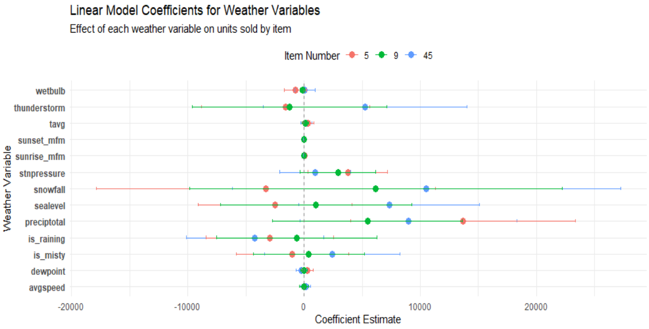
Temperature Variable Correlation - A correlation analysis of all temperature-related variables revealed extreme multicollinearity, with r = 0.98 between tavg and both tmax and tmin, and a strong inverse relationship of r = -0.92 with heat. Retaining these redundant variables would have destabilised the OLS coefficient estimates and inflated standard errors. Before dropping the two correlated columns, I first used them to salvage missing ‘tavg’ data, as it is a derived variable. As this calculation relies solely on data in the same observation, it was performed early to maximise the sample size without risk of data leakage.

Wind Variable Correlation - To ensure model stability and prevent multicollinearity, a correlation matrix was computed for all wind-related features. The results showed a near-perfect positive correlation of 0.90 between Average Wind Speed (avgspeed) and Resultant Wind Speed (resultspeed). Including both variables in the final regression would have inflated the standard errors, making it statistically difficult to isolate the true impact of wind on sales. Consequently, ‘resultspeed’ was dropped. Resultant Direction (resultdir) showed a negligible correlation with speed (0.13) and lacked a meaningful relationship with consumer purchasing behavior. By retaining only ‘avgspeed’, the model focuses on total atmospheric intensity, which is a more reliable predictor of whether a customer will travel to a store.

Splitting and Encoding Weather Codes - The *Codesum* column consisted entirely of string-based weather descriptors, often containing multiple coded events with inconsistent spacing and punctuation. As no direct numerical equivalent exists, the three most frequent weather events—Mist, Rain, and Thunderstorms—were one-hot encoded into binary indicator variables (0/1). This approach reduces noise and allows the models to isolate the distinct effects of precipitation and reduced visibility on sales outcomes.

Wet Weather Correlation - Due to the top three weather codes being largely ‘Wet Weather’ related, I decided to perform a third correlation analysis upon the newly encoded variables against ‘preciptotal’ which is my original key rainy weather indicator. All three indicators showed a moderate positive correlation; ‘is_raining’ showed the highest at 0.42. This was to be expected, as that particular code being present should have also meant precipitation (shown by preciptotal) was measured above 0 as well. The feature ‘is_misty’ showed the lowest correlation, likely due to it only loosely being related to precipitation, with the thunderstorm feature at 0.36. From this, I dropped ‘is_raining’ as it captures rainfall volume in a way that is more statistically interpretable (i.e., as preciptotal increases, unit sales increase, as opposed to ‘unit sales when it is vs isn’t raining’).





Although mist (BR) was among the most frequent weather events, model evaluation showed minimal predictive contribution. Decision Tree importance scores were consistently below 3% across products, and linear regression coefficients were not statistically significant. Given the overlap with precipitation variables and negligible impact on validation performance, mist was excluded to maintain model parsimony and interpretability. (initial importance and OLS coefficient plots seen on left)

Feature Creation (Minutes from Midnight) - The *Sunrise* and *Sunset* variables presented an additional challenge, as time-based string formats are unsuitable for direct numerical comparison. Both variables were converted into *minutes from midnight (MFM)*, producing a linear representation of daylight exposure. This transformation enables the models to quantify the effect of daylight duration on consumer purchasing behaviour.

Zero Inflation - The sales data also exhibited significant zero inflation. Quantile analysis showed that 99% of observations recorded fewer than 159 units sold, indicating that the dataset was dominated by low-volume days. To mitigate statistical noise and sparsity caused by the severe frequency of zero-sales days, I transitioned the analysis from daily to weekly units sold for each item. Unit sales were summed to capture total weekly demand, while weather variables were averaged to reveal the characteristic weather state for each specific week. This transformation stabilised the target variable and allowed the model to focus on the seasonal trends than on the daily randomness (caused by the zero-inflated data)

Outliers - To prevent extreme outliers—such as bulk purchases or data entry anomalies from disproportionately influencing model estimates, observations exceeding 1,000 units sold were excluded, as I found major outliers present and only once below 1,000 sales did the difference between the high units sold slow down. This filtering step ensured that the analysis remained focused on typical consumer behaviour.

Modeling Strategy

The Approach - A dual-model approach was employed to balance interpretability and predictive performance, utilising both Ordinary Least Squares (OLS) Linear Regression and Decision Tree models. The OLS regression model provided an interpretable baseline for estimating the direction and magnitude of individual weather effects. In contrast, the Decision Tree model was applied to capture non-linear relationships and higher-order interactions among meteorological variables, with feature importance assessed through the reduction in loss (Gini impurity or sum of squares) at each split.

Model Split - To ensure generalisability and reduce overfitting, the dataset was partitioned into training (60%), validation (20%), and testing (20%) subsets, with final performance evaluated on the independent test set. Calculations of medians and means for imputation were derived exclusively from the training set and applied to the validation and test sets via custom functions (`imp_df` and `sun_imp`) to prevent data leakage.

Model Strategy - The dual-model strategy successfully leveraged OLS Linear Regression for its transparent coefficient estimates and the Decision Tree (CART) for its ability to map non-linear "threshold" effects, yet both models carry inherent limitations. While the OLS provided a vital baseline for the directional impact of weather—such as quantifying the unit increase in sales per degree of temperature—it struggled to account for the zero-inflated nature of the retail data and assumed a strictly linear relationship that often failed to capture complex meteorological interactions. Conversely, the Decision Tree excelled at identifying specific tipping points, like the significant impact of station pressure on Item 45, but was highly sensitive to the small sample sizes of the weekly aggregates, requiring aggressive tuning of ‘minsplit’ and ‘cp’ parameters to prevent overfitting. Ultimately, while the Decision Tree generally offered higher R^2 values by accommodating these non-linearities, its "step-function" approach to predictions can sometimes lack the granular sensitivity of the OLS model when weather variables move in small, incremental ranges.

The feature set was validated using both OLS coefficients and Decision Tree importance scores. The Feature Importance Table provides the empirical justification for the final variable set, showing that Station Pressure (stnpressure) is a primary driver for Item 45 and Item 9, while Item 5 is uniquely sensitive to Thunderstorms. These findings were corroborated by the OLS Coefficient Analysis, which confirmed the mathematical "direction" of these relationships (e.g., the positive impact of temperature on Item 5) while confirming statistical significance via p-values.

Evaluation Metrics

To evaluate model success, a custom function mod_metrics was implemented to systematically calculate RMSE, MAE, and R^2 across all model-item combinations. By avoiding hard-coding and utilising tidy evaluation via broom and jtools, the strategy ensures a consistent and reproducible comparison. I utilised this approach to remove repetition, allowing for the simultaneous evaluation of multiple products without the risk of manual calculation errors. This modular design proved essential because retail demand often reacts to weather in a "threshold-based" manner—where sales don't just increase linearly with temperature but rather spike or drop once a specific meteorological limit is reached. For most items, the Decision Tree was selected as the superior model for capturing this complex, non-linear nature of weather-driven demand, as it naturally partitions data into these distinct "high" and "low" sales regimes based on environmental triggers.

Visualisation

Decision Tree Importance - Model validity and interpretability were assessed using a host of diagnostic visualisations. Decision Tree feature importance plots ranked weather variables by their overall predictive contribution (calculated via Gini importance), while coefficient plots from the linear models illustrate the effect sizes and confidence intervals of each meteorological factor. These were complemented by actual vs. predicted bar charts, which allowed for a direct comparison of model performance across varying weather conditions.

OLS Coefficient Plot - The Linear Model Coefficients for Weather Variables plot serves as a diagnostic tool to quantify the magnitude, direction, and statistical significance of weather predictors on unit sales. The horizontal axis measures the coefficient estimate, representing the specific unit change in sales for every one-unit increase in a weather factor. By utilising colour-coding for Items 5, 9, and 45, the plot allows for a simultaneous comparison of how different products react to the same environmental triggers, such as the positive impact of temperature on Item 5. The horizontal error bars represent 95% confidence intervals; if these whiskers cross the vertical dashed zero-line, the variable is considered statistically insignificant, which provides the empirical justification for excluding low-impact features like "mist" from the final OLS model.

Predicted vs Actual - For the analysis of weather-driven demand patterns, continuous meteorological variables were transformed into binned impact charts. Key variables—specifically Sunrise/Sunset variables, Average Temperature, Wind Speed, and Station Pressure—were grouped into discrete bins. This approach filtered out noise and ensured that observed trends, such as the relationship between sunset timing and unit sales, were supported by repeatable patterns rather than isolated outliers. This layered visualisation strategy captures both the statistical weight of weather variables and their practical, observable impact on weekly units.

Correlation Heatmap - Once the top weather variables were identified, I generated a correlation heatmap. This provides a high-level, "at-a-glance" answer to the question: “Does weather have a significant impact on sales?" This plot utilised the train_set because it had already undergone full preprocessing and imputation, ensuring the correlation coefficients were calculated on a robust, complete data foundation.

Station Pressure Effect Chart - To further investigate the relationship between barometric pressure and product-level sales performance, station pressure (stnpressure) was categorised into interpretable weather regimes and analysed across the three selected products (Items 5, 9, and 45).

This visualisation supports earlier model findings indicating that station pressure was among the most influential predictors in both Linear Regression coefficients and Decision Tree feature importance.

Thresholds - <28.5 - Low Pressure - Associated with storms.

28.5-28.9 - Stable weather - Neutral atmospheric importance.

>28.9 - Optimal Conditions - Clear and dry weather.

These thresholds were selected to:

- Create meaningful atmospheric regimes
- Improve interpretability for business stakeholders
- Align with meteorological conventions where high-pressure systems often indicate stable consumer conditions

The factor levels were explicitly ordered: Low Pressure → Stable Weather → Optimal Conditions. This ensures logical progression in the visual output.