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.

Preprocessing and Feature Engineering

The raw meteorological dataset contained several non-numerical variables that required transformation prior to modeling. Weather data frequently includes character-based entries, with observations marked as “T” (Trace) and “M” (Missing). These values were addressed using median imputation in combination with replace_na. Removing these rows would have resulted in substantial data loss, particularly during extreme weather conditions. Median imputation preserves the underlying seasonal distribution while minimizing distortion from outliers.

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 behavior.

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.

Prior to modeling, a correlation analysis was conducted on wind-related variables (avgspeed, resultspeed, and resultdir). High correlation among independent variables can lead to unstable coefficient estimates in Ordinary Least Squares (OLS) regression. The resulting correlation matrix confirmed which variables provided overlapping information. Based on these findings, resultdir was excluded from the final models. Wind direction is measured in circular degrees, which can introduce mathematical bias, and it demonstrated a weaker relationship with units sold relative

to wind intensity. Its removal improved model interpretability without compromising predictive performance.

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 prevent extreme outliers—such as bulk purchases or data entry anomalies—from disproportionately influencing model estimates, observations exceeding 1,000 units sold were excluded. This filtering step ensured that the analysis remained focused on typical consumer behavior.

Modeling Strategy

A dual-model approach was employed to balance interpretability and predictive performance, utilising both Ordinary Least Squares (OLS) Linear Regression and Random Forest models. The OLS regression model provided an interpretable baseline for estimating the direction and magnitude of individual weather effects. In contrast, the Random Forest model was applied to capture non-linear relationships and higher-order interactions among meteorological variables, with feature importance assessed using *IncNodePurity*.

To ensure generalizability 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.

The OLS model relies on standard assumptions, including linearity, independence of errors, homoscedasticity, and approximate normality of residuals. While these assumptions facilitate clear and interpretable coefficient estimates, real-world consumer demand and weather relationships may deviate from strict linearity. The Random Forest model mitigates this limitation by capturing non-linear effects and interaction terms without requiring parametric assumptions, though this comes at the cost of reduced interpretability.

Weather variables were derived from station-level observations and therefore serve as proxies for actual consumer exposure. Microclimatic variation and individual-level behavioral factors—such as promotions, holidays, or stock availability—were not explicitly modeled and may contribute to unexplained variance.

Random Forest hyperparameters were selected using validation-set performance to balance predictive accuracy and model stability. Parameters such as the number of trees (ntree) and the number of variables considered at each split were evaluated with respect to RMSE, with final values chosen to minimize error while avoiding overfitting. This approach ensured that the models remained robust and generalizable across different product demand profiles. I chose the number of trees used in the Random Forest model by incorporating the “optRF” library, this package inside R worked to find the optimal number of trees by testing the model with multiple

different “ntree” values. Once ran, the package found 1000 trees to yield the best results for the set.

Creating the models with three item numbers in mind, I initially fitted each model to each item individually. I later decided to remove the initial hardcoding and instead use a “for” loop to filter each item and create only the OLS and RandomForest model. This meant I’d have filter when using the train/ validation set but became increasingly useful and time efficient during the visualisation and performance analysis.

OLS coefficient interpretability was preserved by retaining variables in their original units. Random Forest models are inherently scale-invariant and therefore unaffected by differences in variable magnitude. As a result, normalization or standardization was unnecessary for either modeling approach. Standardizing the data would have converted coefficients into standard deviation units, reducing interpretability for retail stakeholders seeking to understand the physical impact of weather conditions on sales. Furthermore, normalization would have provided no predictive advantage in this context.

To prevent data leakage, all preprocessing steps—including imputation, feature transformations, and filtering thresholds—were derived exclusively from the training dataset and then applied to the validation and test sets. This ensured that performance metrics reflected true out-of-sample generalization rather than information leakage.

Within the modelling step, due to the different splits, I incorporated functions and ‘for’ loops to remove many instances of ‘hard coding’. One example of a function I used is the “mod_metrics” function, which was used to feedback the model metrics such as MAE.

Evaluation Metrics

Model performance was assessed using Root Mean Square Error (RMSE) and R-squared values computed on the validation dataset via the postResample function. This evaluation framework enabled a statistically grounded comparison between the Linear Regression and Random Forest models for each product. While the linear model offered superior transparency regarding the direction of weather impacts, the Random Forest model frequently achieved improved predictive accuracy for products exhibiting non-linear or interaction-driven demand. Evaluating both models ensured that feature importance rankings were derived from the most appropriate model for each specific item.

Visualisation

Model validity and interpretability were further assessed using a suite of diagnostic visualizations. Random Forest feature importance plots ranked weather variables by their overall predictive contribution, while coefficient forest plots from the linear models illustrated effect sizes and confidence intervals. These were complemented by *Actual vs. Predicted* scatter plots, benchmarked against a 45-degree reference line, to identify systematic over- or under-prediction across sales volumes.

For the analysis of weather-driven demand patterns, continuous meteorological variables were transformed into *Binned Impact Charts*. Variables such as temperature and wind speed were grouped into discrete bins, with a minimum threshold of three historical observations per bin. This approach filtered out statistically sparse regions and ensured that observed trends—such as the relationship between sunset timing and unit sales—were supported by repeatable patterns rather than isolated events. Collectively, this layered visualization strategy captures not only the statistical importance of weather variables but also their practical, observable impact on units sold.

Another variable I believed to be of use to management is the predicted sales over a yearly period, due to the interpretations of the visualisations thus far, It became clear that an explanation for the correlation of weather and units sold could be illustrated by seasonal norms.

Finally, once the top 5 weather variables had been identified, I created a heatmap, showing the correlation between Units and Weather in an easily interpretable way for viewers who want an easier way to see “Does weather have a significant Impact on sales?”. For this plot I used the train_set as opposed to the final data frame, due to all the null values having already been imputated onto the train set, which saved another layer of preprocessing.