Module: Business Intelligence
Lecturer: Hans-Friedrich Witschel

Analytical and Predictive Evaluation of Heating System Modes

External Challenge: Plutinsus, Challenge #2

Group 11: Joel Cherianthanathu, Preeti Goyal, Kathrin Puthiyedathu, Venkata Ramana Tella

10th of June 2025

Table of Contents

List of Figures

List of Tables

Part 0: Executive Report Summary

This project, conducted in collaboration with Plutinsus and Electrojoule, addresses the challenge of automating the classification of heating system operating modes – specifically distinguishing between normal daytime operation and nighttime setback mode. The goal is to create a model to predict the nighttime setback.

Part 2: Analytical Insights
A series of analytical questions were explored using real-time sensor data and visualized through Tableau dashboards. These questions aimed to uncover patterns in heating behavior and identify opportunities for optimization:

- **Timing of Nighttime Setback Mode**: Determining the exact hours of setback mode enables precise automation and alerting.
- **Seasonal Heating Behavior**: Understanding how heating patterns shift between fall and winter supports adaptive control strategies.
- **Supply Temperature Reduction**: Measuring temperature drops during setback validates system performance and identifies inefficiencies.
- **Temperature Relationships**: Analyzing the interplay between outside, return, and supply temperatures informs smarter, context-aware heating logic.
- **Manual Overrides**: Detecting unexpected manual interventions helps flag policy breaches and reduce unnecessary energy costs.

Part 3: Predictive Modeling
A binary classification model was developed to automate the detection of heating system states. Using features such as hour, outside temperature, return temperature, and supply temperature, several machine learning models were trained and evaluated using Orange Data Mining.

Modeling Approach
The following models were tested: Logistic Regression, Decision Tree, CN2, Gradient Boosting, Random Forest, SVM

Evaluation Method
10-fold cross-validation using AUC, Accuracy, Confusion Matrix

Key Findings

- Gradient Boosting and Logistic Regression performed best in terms of AUC and interpretability.
- Most errors occurred during nighttime, highlighting edge cases for further refinement.
- Class imbalance was addressed through oversampling and undersampling techniques.

Business Impact / Benefit for Plutinsus

- Automates detection of heating modes, reducing manual effort.
- Enables real-time alerts for anomalies or overrides.
- Lays the foundation for scalable deployment across multiple buildings.

Conclusion
The predictive model enables automated detection of heating modes, reducing manual effort, and enabling real-time alerts for anomalies. While current models perform well, future work will focus on generalizing these models across different buildings.

Part 1: Background Information

Automated Classification of Heat Distribution Operating Modes for Cost-Effective Digital Maintenance

Electrojoule, in collaboration with Plutinsus' software package, offers a cost-effective digital maintenance subscription tailored for property owners who seek to minimize heating costs and optimize heating system performance. Their advanced digital monitoring solutions leverage real-time data to enhance heating efficiency, reduce unnecessary energy consumption, and extend the operational lifespan of heating components.

A significant challenge in this process is the automated classification of heat distribution operating modes, particularly distinguishing between daytime normal mode and nighttime setback mode. Currently, this classification is managed manually, which presents limitations in efficiency. Automating the process would not only streamline system optimization but also enhance overall energy savings.

Part 2: Designing a Report

A. Analytical questions

Q1. At what times is the nighttime setback mode active?

Q2. How does the system's heating behavior vary across seasons (Fall vs. Winter)?

Q3. How much is the supply temperature reduced during nighttime hours?

Q4. What is the relationship between outside temperature, return temperature, and supply temperature during periods when setback mode is active compared to when it is inactive?

Q5. Is there any unexpected manual override during nighttime (based on system state changes)?

B. Multidimensional model in ME/R notation

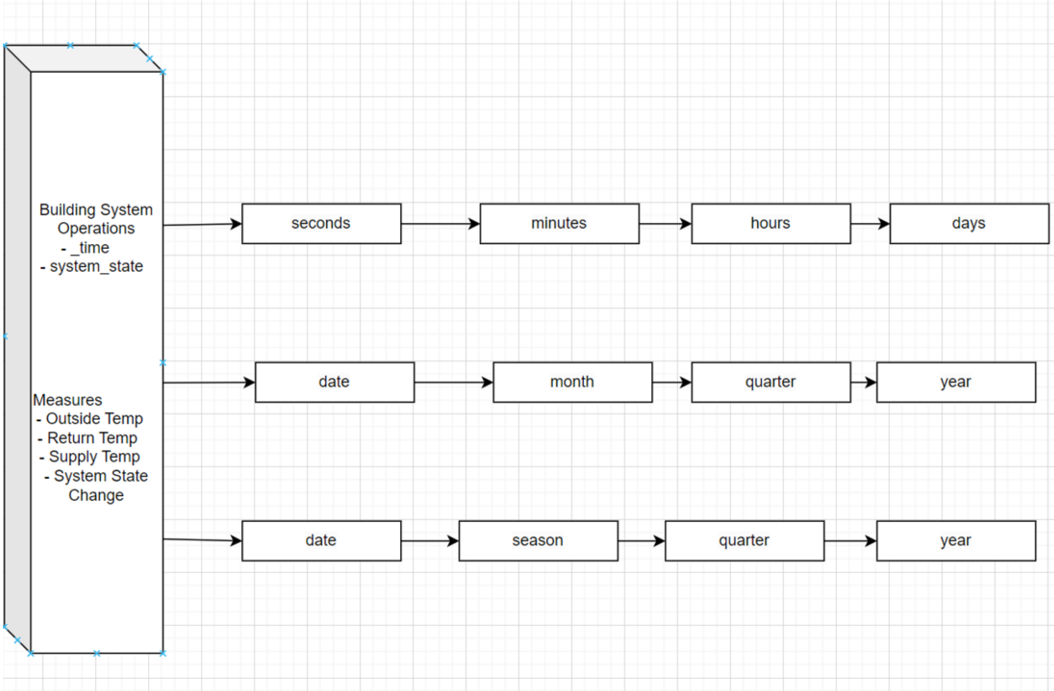

Figure 1: ME/R Model

C. Tableau Workbook

Calculated fields

Column Name	Description
Formated Time	Every min of the day, extracted from timestamp
Heating_Mode	System State descried
Mode_Time	Mode of the day like night, morning, evening, afternoon
Nightsetbackmode	Assumed Nighttime 10 pm-6 am
Seasons	Either fall or wiinter
Setback Mode Active	Ground truth is 1
Delta Temp	Difference between supply and return temperature of the water inside the heating system
Outside Temp	External environmental temperature
Return Temp	The temperature of the water flowing back from the building to the heater
Supply Temp	The temperature of the water flowing from the heater into the building
System State	Target variable (1= setback mode, 0 = normal mode) Previously manually measure state of the heating system, on/off
System State Change	When change of system is registered

Table 1: Calculated Fields

Overview all Tableau-Dashboards

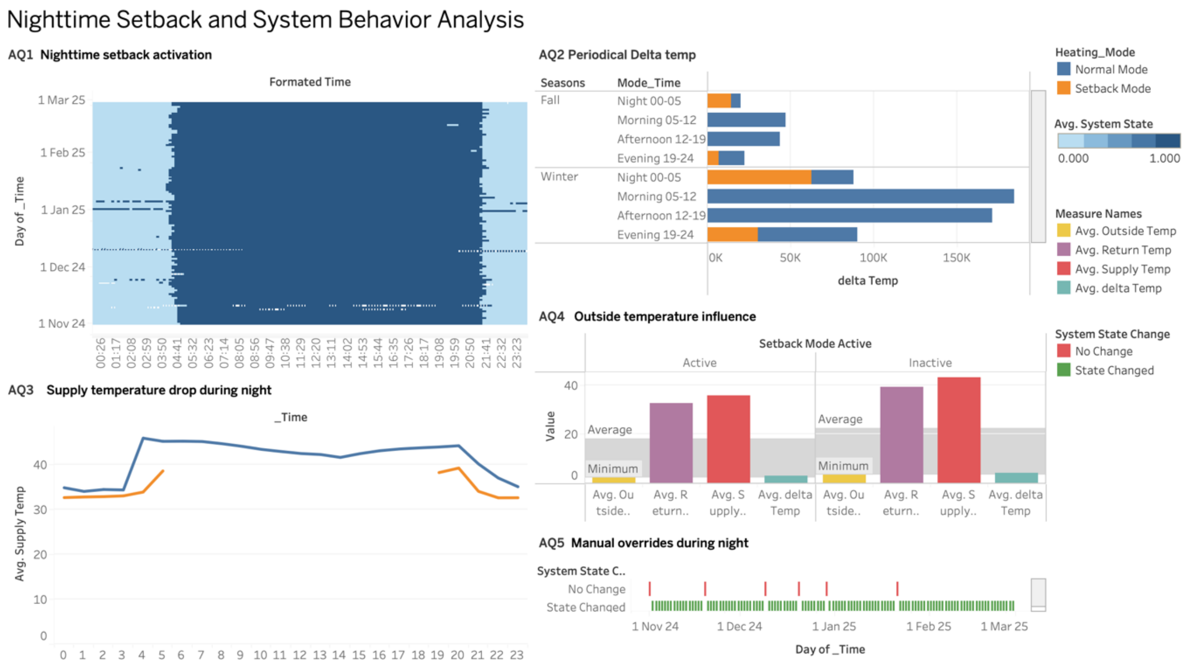

Figure 2: Overview Tableau Dashboards

Short explanation of each Dashboard

Q1: At what times is the nighttime setback mode active?

The ground truth is visualized over a 24h period. This can narrow down the starting and ending timespan of the nighttime setback: 20.30-21.30pm and 3.30-4.30am. Additionally, the outliers are visualized.

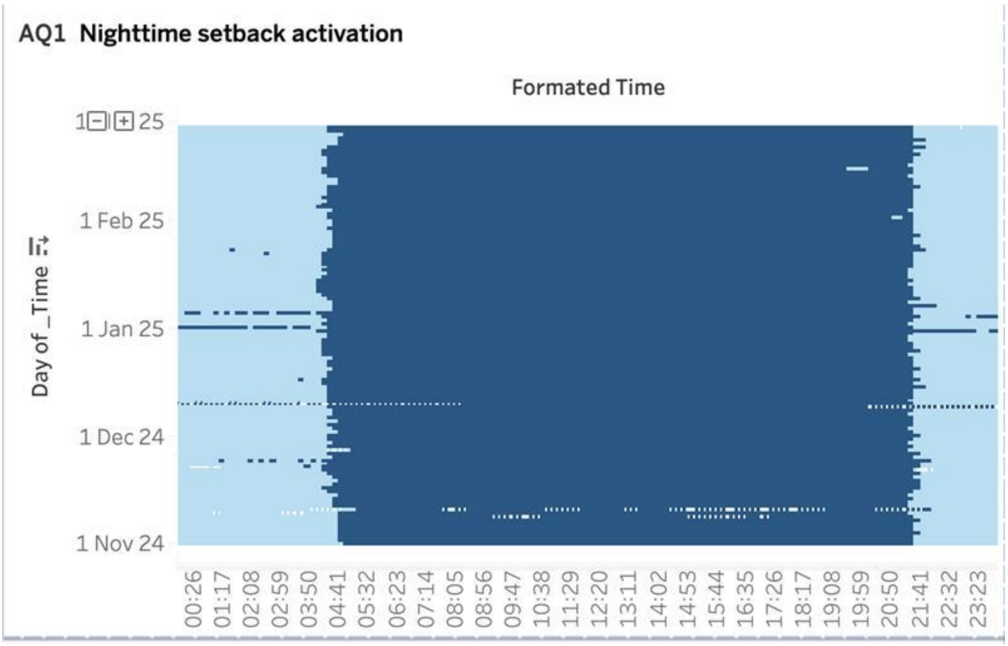

Figure 3: Tableau Dashboard for Q1

Business Value:

- Identifying the exact time windows when the heating system enters and exits setback mode allows for precise automation.
- Helps detect deviations from expected behavior, such as early or delayed transitions.

Potential Benefits for Plutinsus:

- Configure automated alerts if the system fails to enter setback mode on time.
- Provide customized recommendations to clients for optimizing heating schedules based on actual usage patterns.
- Improve energy-saving algorithms by aligning them with real-world behavior.

Q2: How does the system's heating behavior vary across seasons (Fall vs. Winter)?

Generally, the setback mode is turned on earlier and longer during colder seasons.

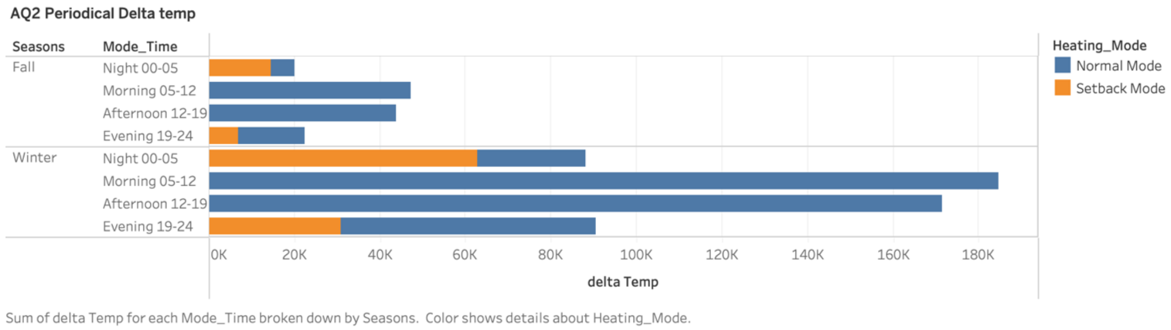

Figure 4: Tableau Dashboard for Q2

Business Value:

- Seasonal variation analysis helps in adaptive control strategies.
- Enables forecasting energy demand and adjusting heating schedules accordingly.

Potential Benefits for Plutinsus:

- Offer season-specific optimization plans to clients.
- Adjust predictive models to account for seasonal shifts, improving accuracy.
- Help clients budget heating costs more effectively by understanding seasonal trends.

Q3: How much is the supply temperature reduced during nighttime hours?

The temperature of the supply water drops significantly if the nighttime setback is turned on.

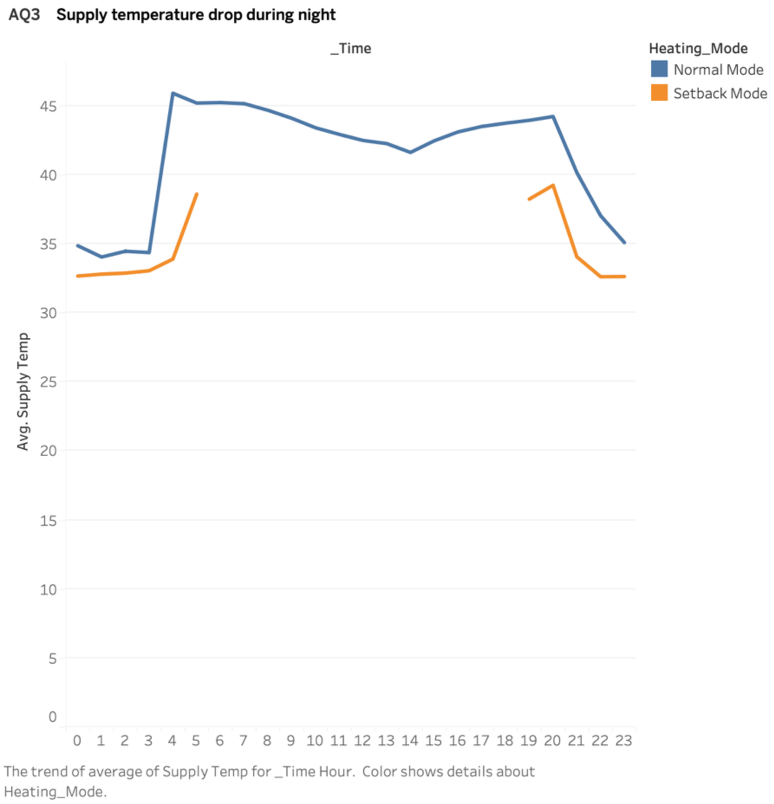

Figure 5: Tableau Dashboard for Q3

Business Value:

- Quantifying the temperature drop validates whether the setback mode is functioning as intended.
- Helps identify inefficiencies or malfunctions in the heating system.

Potential Benefits for Plutinsus:

- Use this metric to benchmark system performance.
- Remark on the graph: the orange graph indicates when the nighttime setback is active across the hours of the day.

Q4: What is the relationship between outside temperature, return temperature, and supply temperature during periods when setback mode is active compared to when it is inactive?

When the nighttime setback is active, the delta between return and supply temperature is smaller than when the nighttime setback isn't active.

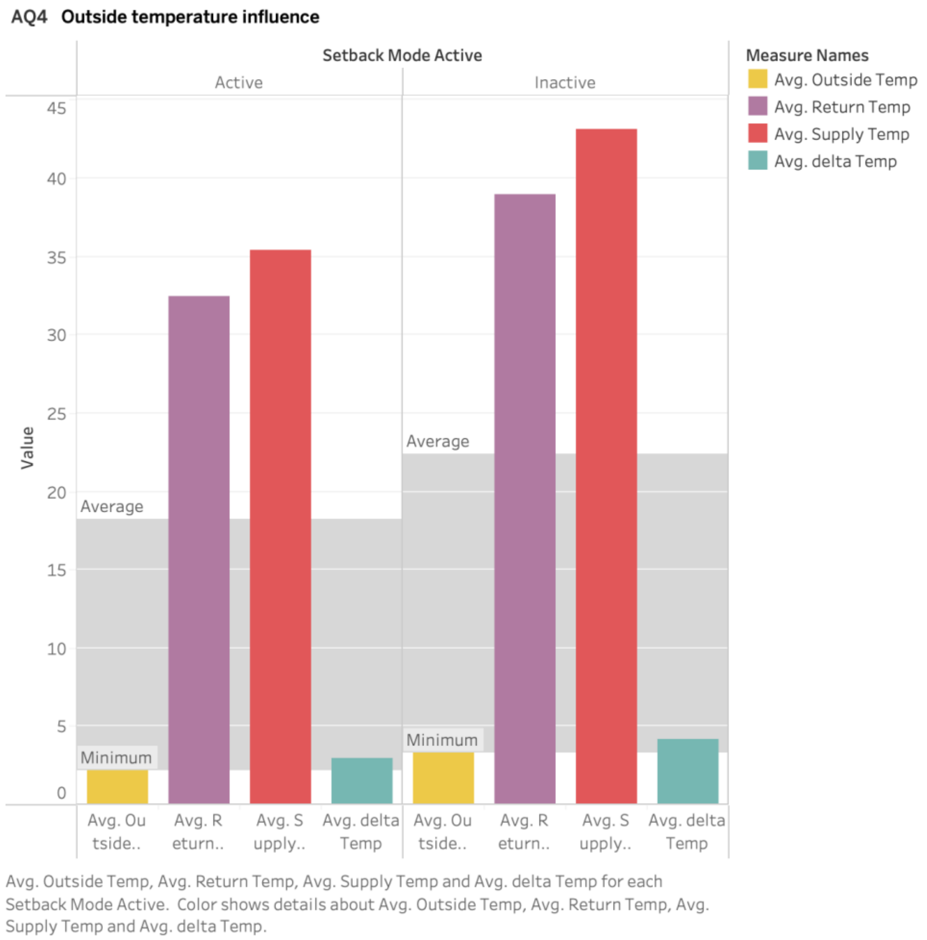

Figure 6: Tableau Dashboard for Q4

Business Value:

- Understanding these relationships helps in fine-tuning control logic.
- Enables context-aware heating strategies that respond to environmental conditions.

Potential Benefits for Plutinsus:

- Develop smarter algorithms that adjust heating based on external and internal temperatures.
- Provide data-driven insights to clients on how weather impacts their heating efficiency.
- Identify anomalies that may indicate sensor issues or system faults.

Q5: Is there any unexpected manual override during nighttime (based on system state changes)?

While most nights show correct nighttime setback operation, the red bars indicate noticeable instances of unexpected behavior (no setback) during setback hours. These events should be flagged for operational review — especially when manual heating overrides can increase energy cost or breach policy.

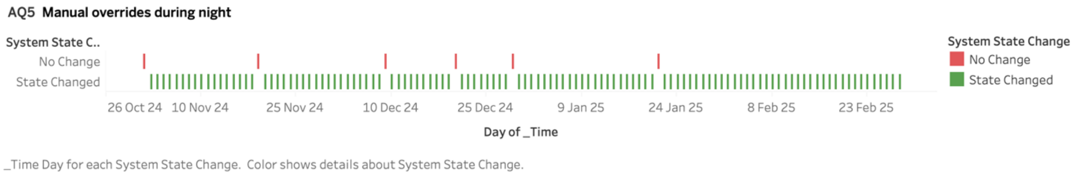

Figure 7: Tableau Dashboard for Q5

Business Value:

- Manual overrides during setback hours can lead to increased energy costs and policy violations.
- Detecting these events supports compliance monitoring and cost control.

Potential Benefits for Plutinsus:

- Alert clients when manual interventions occur outside of expected behavior.
- Help clients educate tenants or staff on energy-saving practices.
- Use override data to refine predictive models and improve system robustness.

Part 3: Learning a Predictive Model

A. Formal definition of the classification/regression task

Objective

The primary goal is to develop a binary classification model that can automatically detect whether a building's HVAC (Heating, Ventilation, and Air Conditioning) system is currently operating in normal daytime mode (system_state = 0) or in nighttime setback mode (system_state = 1).

This classification task addresses a real operational need in building management systems where:

- Improved energy efficiency
- Smarter alerting systems
- Scalable operational monitoring across buildings

Binary classification task

We selected the following **Target Variable**:

- Name/ID: system_state
- Type: Categorical (Binary)
- Classes:
  - 0: Normal Operation Mode
  - 1: Nighttime Setback Mode

Features Used:

The model uses the following 8 selected features:

Features	Description
hour	Hour of the day (0–23), derived from timestamp
outside_temp	Outside air temperature (numeric)
supply_temp	Supply water temperature (numeric)
return_temp	Return water temperature (numeric)
delta	Temperature difference: supply_temp - return_temp
supply_temp_prev_1	Supply temperature at the previous time step
supply_return_ratio	Ratio: return_temp / supply_temp
supply_temp_variation	Rolling 3-hour standard deviation of supply temperature

Table 2: Selected 8 Features for the Model

Data Preprocessing:

We focused our evaluation on buildings 3, 6, 8, 11, 12, and 16. These buildings were selected because they are the only ones in the dataset that include verified ground truth data corresponding to various setback modes. The availability of this ground truth information is critical for accurately assessing model performance. It enabled us to train and validate the model using reliable labels, thereby ensuring that the selected model aligns effectively with the real-world behavior of each building under different setback scenarios.

Data Collection and Integration

- **Source Files**: Raw sensor data (outside, supply, return temperatures) and system state logs (Result) for on Buildings 3, 6, 8, 11, 12, and 16
- **Building Consolidation**: Data from on Buildings 3, 6, 8, 11, 12, and 16was merged to create a unified dataset.
- **Time Alignment**: All data streams were aligned on a common timestamp (_time) using outer joins to retain all records.

Missing Value Handling

- **Forward Fill + Backward Fill**: Applied to sensor values to ensure temporal continuity.
- This ensures no gaps in rolling calculations or model input continuity.

Time-Based Filtering

- Filtered data to 10-minute intervals (e.g., 00:00, 00:10, etc.) to standardize time resolution.
- Helps in creating consistent rolling windows and improving time-series clarity.

Feature Formatting and Cleanup

- Removed unnecessary metadata columns: TIME, DATE, and index columns like Unnamed: 0.
- Added derived columns:
  - hour from timestamp
  - date and time strings for grouping and analysis

Feature Engineering

- delta: supply_temp - return_temp → heating/cooling activity
- supply_temp_prev_1: lag of supply_temp by one time step (10 minutes)
- temp_drop_last_30min: supply_temp(t) - supply_temp(t-3) → sharp drops in temperature
- supply_return_ratio: return_temp / supply_temp → relative efficiency
- supply_temp_variation: 3-hour

B. Orange Workflow

Workflow Description:

The Orange workflow includes the following key components:

1. File and Select Columns

- Loads the dataset (cleaned_for_orange.csv)
- Allows the selection of features and the target variable (system_state)

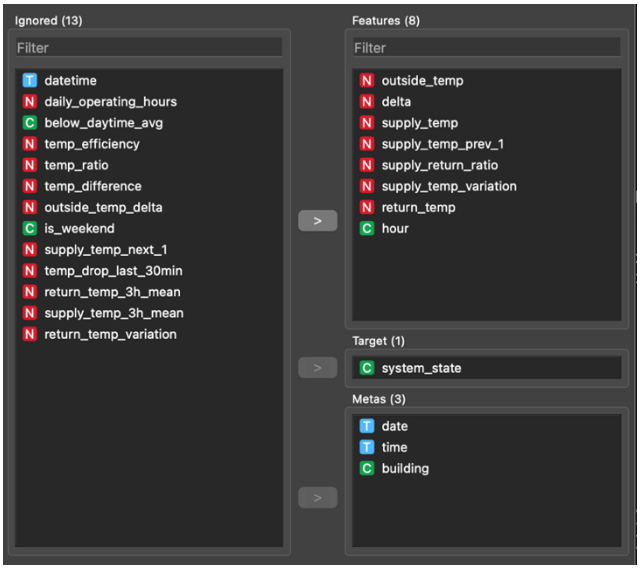

Figure 8: Selected Columns for the Workflow

2. Learner Models (connected to Test and Score)

The following models were trained and compared using Orange's Test and Score Widget:

Model	**Description & Use Case**
Decision Tree	Rule-based model splitting on feature thresholds. Useful for interpretable, rule-driven decisions.
CN2 Rule Induction	Generates interpretable "if-then" rules, ideal for integration into expert systems.
Logistic Regression	Estimates the probability of binary outcomes; shows influence of each feature. Excellent transparent, low-complexity models.
Gradient Boosting	Ensemble method that boosts weak learners to improve accuracy. High-performing but less interpretable.
Random Forest	Combines multiple decision trees using bagging. Balances performance and interpretability; robust to overfitting.
SVM (Support Vector Machine)	Use hyperplanes for class separation in high-dimensional space. Effective for complex boundary separation but harder to interpret.

Table 3: Tested Models

Training Path for the Models:

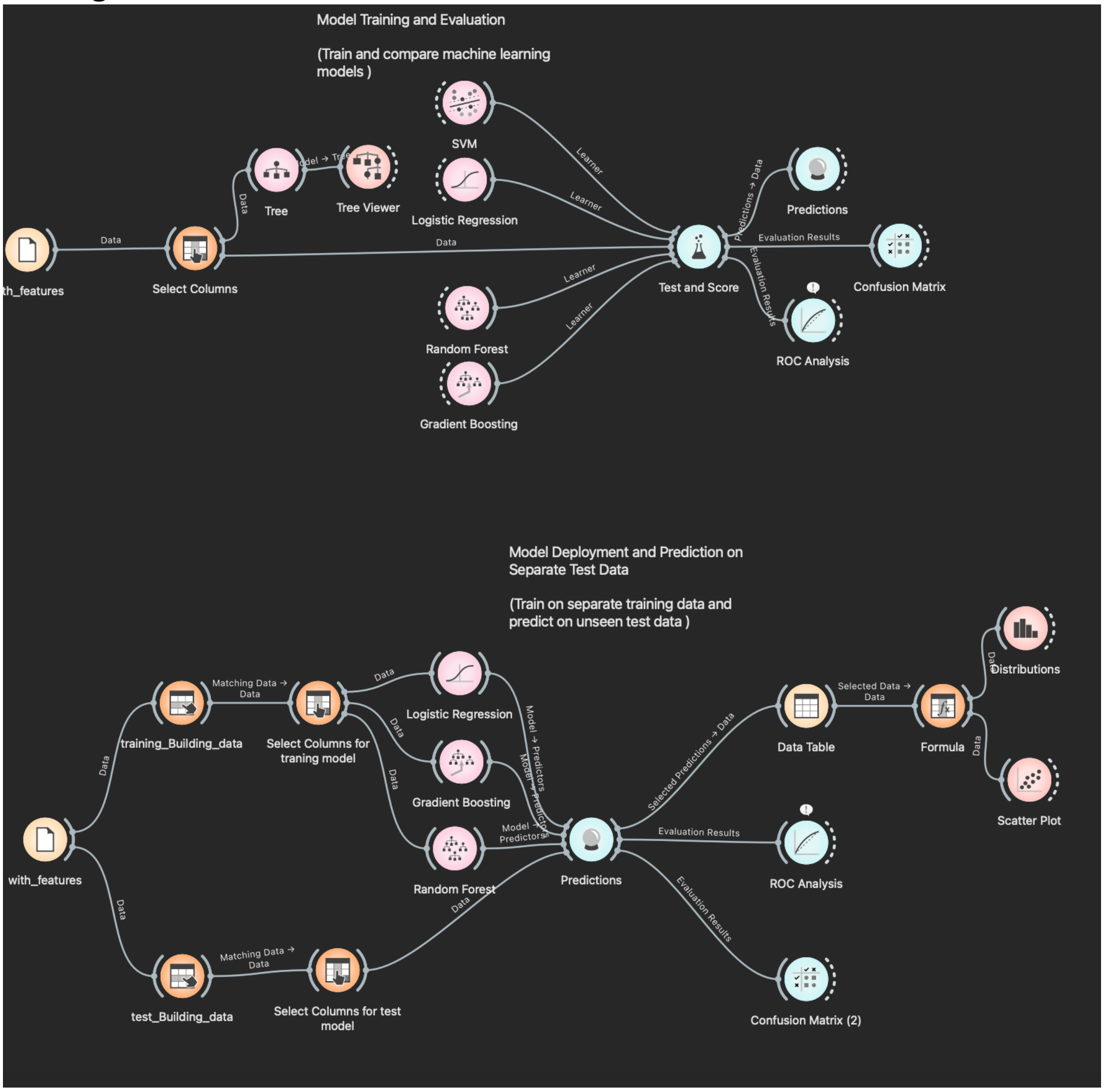

Figure 9: Full Orange Workflow

In Figure 9, the entire workflow is depicted. It is divided into two main phases: The Training & Evaluation above, and the Predictions below. In Training & Evaluation, multiple models were trained in parallel:

- Decision Tree
- SVM (Support Vector Machine)
- Logistic Regression
- Random Forest
- Gradient Boosting

Further, the Tree Viewer helped in decide on the features.

All 4 models were tested and cross-validated with the “Test and Score”-Widget, at the top right. The data was split into two: 1 dataset without building number 3, and one is with building number 3 – the tests were conducted accordingly. The output of the testing is sent to a prediction viewer, a confusion matrix and a ROC-analysis and evaluated.

C. Evaluation Report

Evaluation Method

We used 10-fold cross-validation through the *Test & Score* widget in Orange. The dataset was partitioned into 10 equal segments. Each fold used 9 segments for training and 1 for testing. Additionally, we applied random sampling by the 'building' feature and repeated training/testing 20 times with a 60% training set size. This robust method ensures general performance estimates, particularly for large and imbalanced datasets.

Evaluation Tools Used

Tool	Purpose
Test & Score	Cross-validation with multiple performance metrics
Confusion Matrix	Displays true vs. false predictions across actual and predicted classes
ROC Analysis	Evaluates model sensitivity vs. specificity across thresholds
Predictions Viewer	Shows predicted probabilities and classification errors
Scatter Plot / Box Plot	Highlights class separation, feature distributions, and outliers
Distribution Viewer	Examines class imbalance across feature values (e.g., time of day)

Table 4: Evaluation Methods

Evaluation Metrics

To address the dataset's imbalance, we optimized AUC and included other metrics for a more detailed view.

Metric	Purpose
AUC	Measure's ability to separate the setback (1) and normal (0) modes robustly
Accuracy (CA)	Indicates overall correctness, though biased in imbalanced datasets
F1 Score	Balances precision and recall, highlighting model robustness on the minority class
Precision & Recall	Precision reveals false alert rate; Recall highlights missed setbacks
MCC (Matthews Corr. Coeff.)	Captures balance between classes, good for binary classification

Table 5: Evaluation Metrics

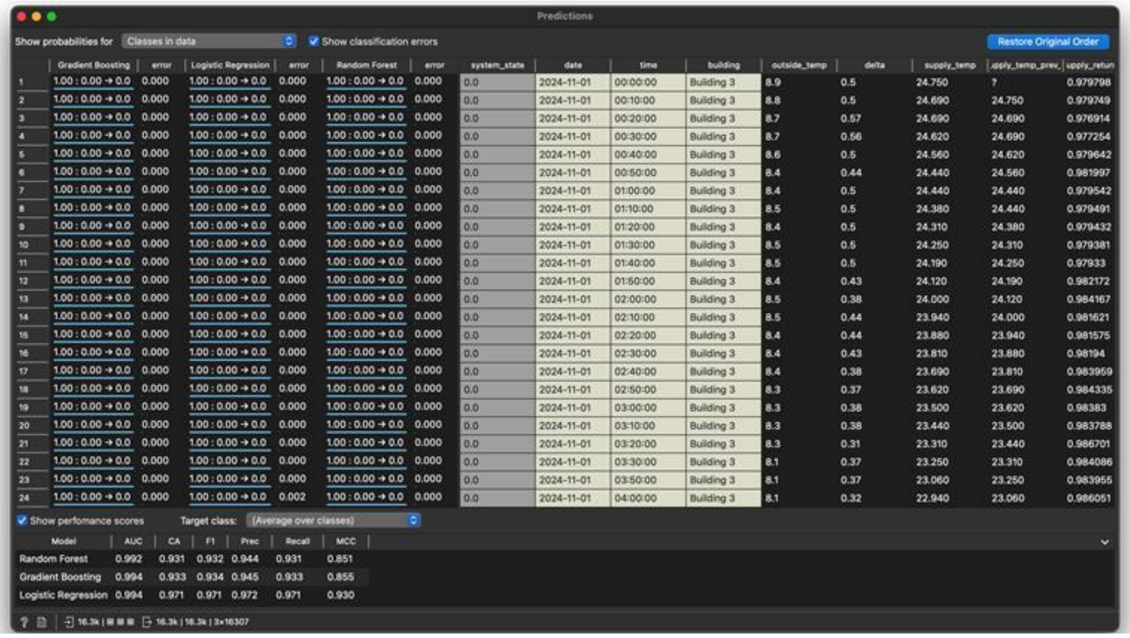

Figure 10: Prediction and Performance Score

Figure 10 displays the prediction and performance score of the test data. All 3 models (Random Forest, Gradient Boosting as well as Logistic Regression) indicate a high prediction and performance score.

Model Interpretation & Comparison

We focused on the following four models:

Model	**AUC**	**Accuracy**	**F1**	**Recall**	**MCC**	**Interpretation Notes**
Gradient Boosting	0.995	0.823	0.843	0.895	0.671	High AUC and robust recall. Handles imbalance and nonlinearities well.
Random Forest	0.995	0.960	0.959	0.961	0.893	Excellent accuracy and recall. Best overall performer in confusion matrix and ROC.
Logistic Regression	0.891	0.745	0.635	0.554	0.500	Fast and interpretable, but underperforms on imbalance — poor recall, misses setbacks.
SVM (upper view)	0.709	0.712	0.718	0.712	0.340	Lower performance and slow training time. Not selected for final evaluation.

Table 6: Comparison of Best Performing Models

Confusion Matrix Analysis

Gradient Boosting:

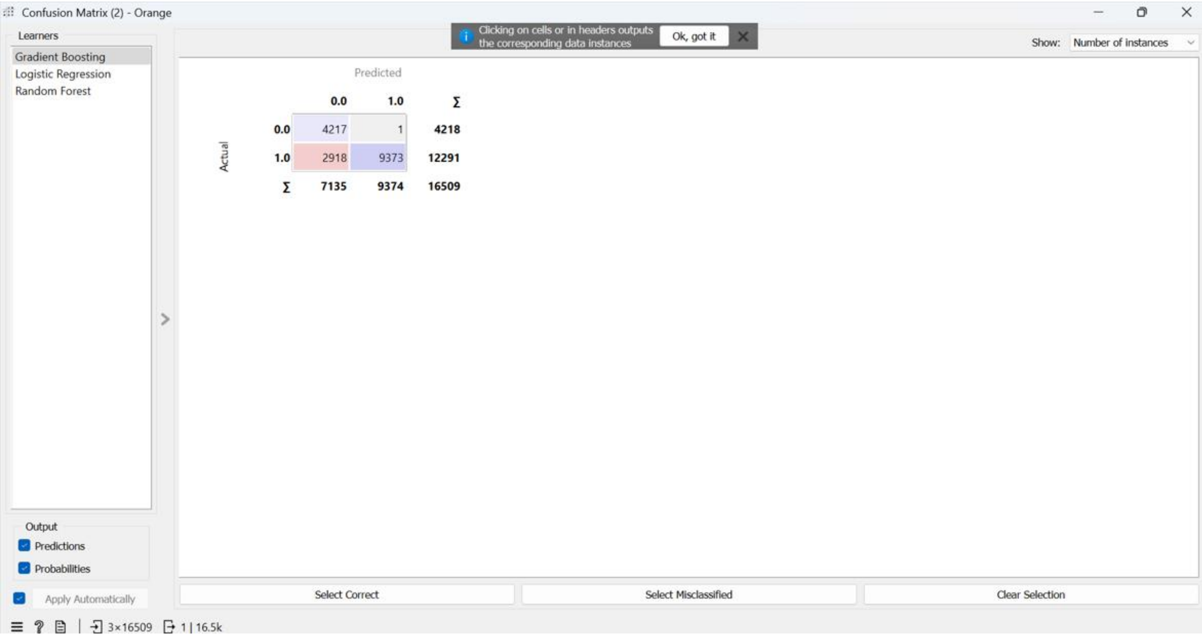

Figure 11: Gradient Boosting - Confusion Matrix Analysis

- False Negatives: 2918 setback missed → Energy Waste / Inefficiency
- False Positives: 1 harmless alert

Logistic Regression:

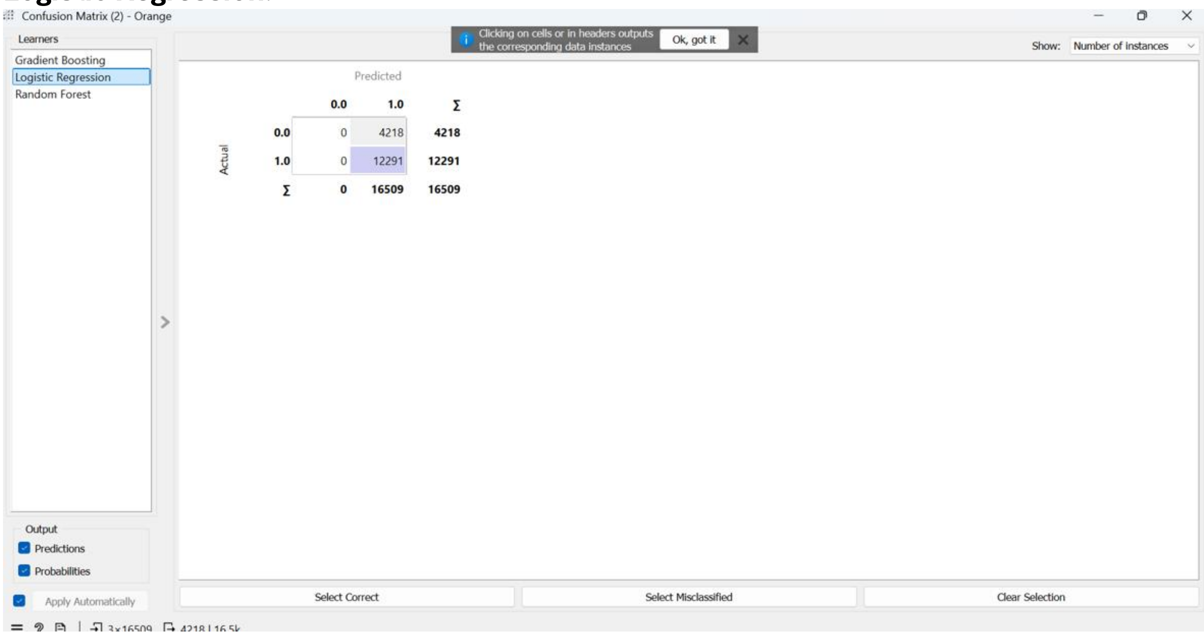

Figure 12: Logistic Regression - Confusion Matrix Analysis

- Predicts all samples as setback (1) → zero true negatives, many false positives → not usable in practice

Random Forest:

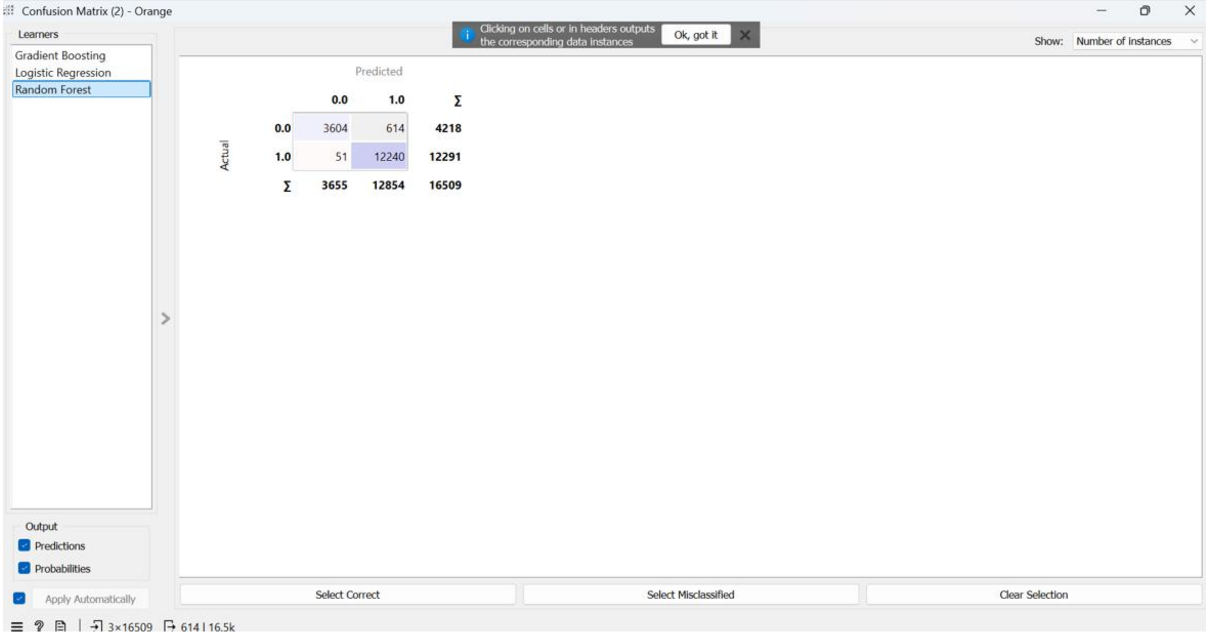

Figure 13: Random Forest - Confusion Matrix Analysis

- False Negatives: 51 very low → strong reliability
- False Positives: 614 manageable → better than Logistic Regression

ROC-Curve Comparison

The ROC Analysis confirms that Gradient Boosting and Random Forest outperform Logistic Regression, which follows a less steep curve and fails to distinguish between classes effectively. Both Gradient Boosting and Random Forest achieve near-perfect separation (AUC ~0.995).

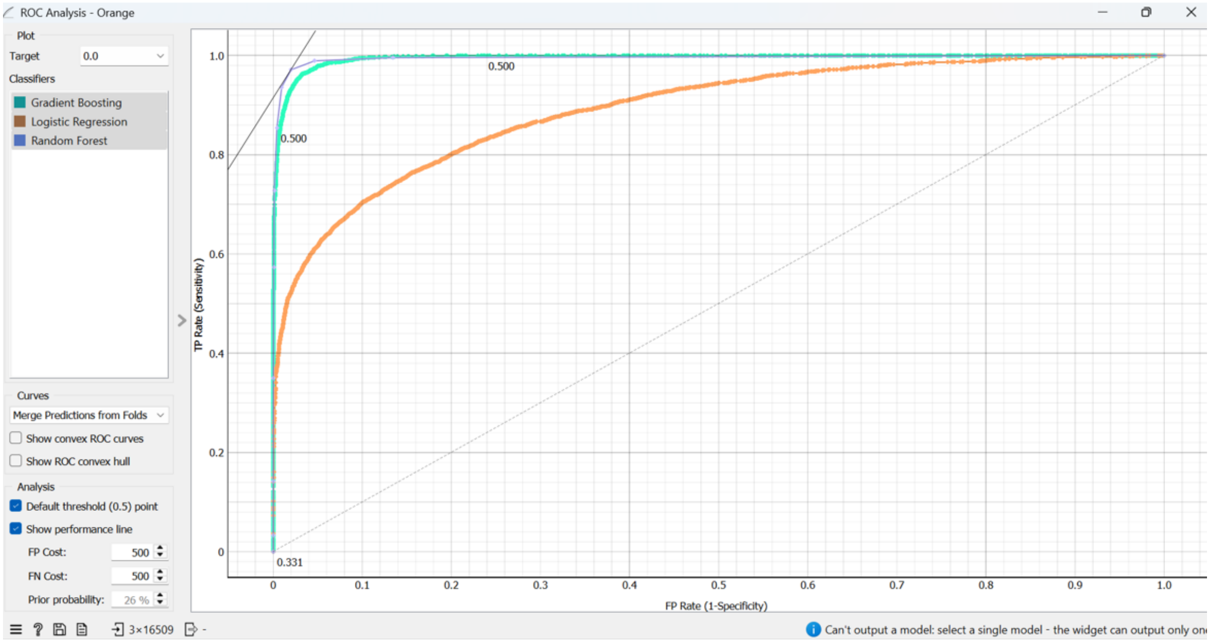

Figure 14: ROC-Curve Comparison

Prediction Error Visualization

The *Predictions* view shows:

- **Random Forest** has the lowest error values.
- **Gradient Boosting** achieves good probabilities (close to 0 or 1) with minimal uncertainty.
- **Logistic Regression** consistently misclassifies class 0 samples, suggesting failure in threshold adaptation for imbalanced data.

Bias and Variance Check

Model Type	Complexity	Variance/Bias Insights
Logistic Regression	Low	High bias — underfits the data, especially for setback detection
Gradient Boosting	High	Solves bias; some signs of overfitting in Confusion Matrix FN count
Random Forest	Moderate	Balanced performance, good generalization

Table 7: Bias and Variance for each Model

Error Implications

- False Negatives (FN): Missed setbacks → wasted heating energy.
- False Positives (FP): Incorrect setback alerts → minor inconvenience, low cost

Thus, minimizing FN is more critical to reduce energy waste, aligning well with AUC and Recall as optimization metrics.

Handling Imbalanced Data

We observed:

- ~26% of data belongs to class 0 (normal mode), ~74% to class 1 (setback)
- Random Forest and Gradient Boosting effectively manage imbalance without explicit resampling, thanks to their structure.
- Logistic Regression fails without balancing (zero true negatives)

We used:

- AUC as robust metric
- ROC & Confusion Matrix to validate minority class detection.

Visualization Insights

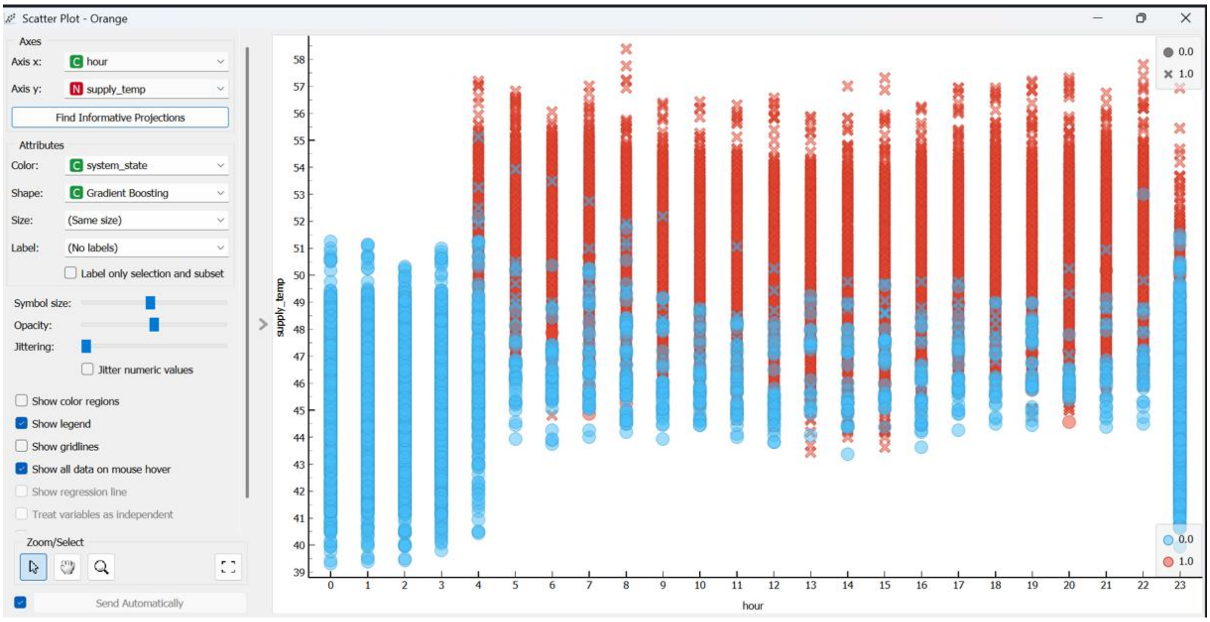

Figure 15: Scatter plots (Supply Temp vs. Hour)

- Show separation between system states.
- Class 1 (red) dominates most hours except early morning (00:00–04:00) where class 0 (blue) is more frequent.
- Gradient Boosting and Random Forest capture these patterns better than Logistic Regression

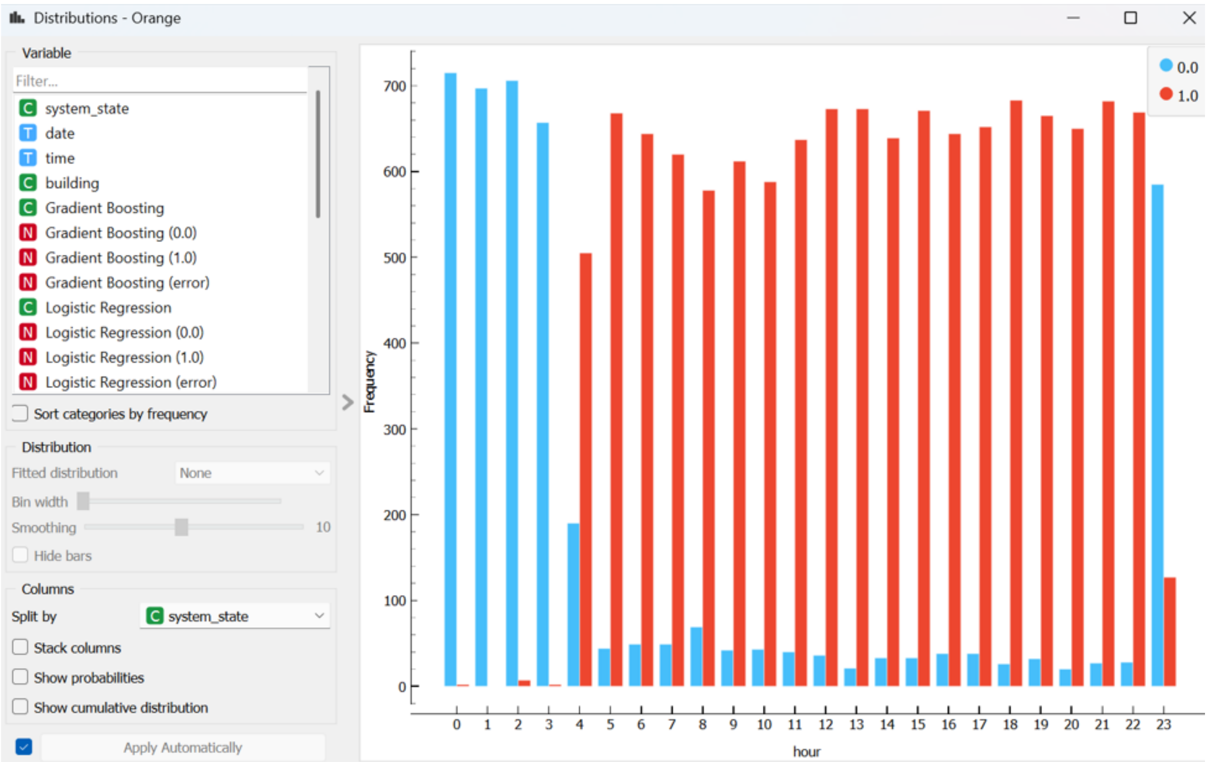

Figure 16: Distributions

- Confirm class imbalance by hour – setback mode is dominant throughout most of the day except during the night.

Conclusion of the Predictive Modelling

- Automated detection of nighttime setback mode becomes feasible with Gradient Boosting or Random Forest.
- Manual monitoring can be replaced with real-time predictive analytics.
- The models can be used to enable alert systems that reduce energy waste by identifying when heating fails to enter setback mode.

The predictive model enables automated detection of heating modes, reducing manual effort and enabling real-time alerts for anomalies. While current models perform well, future work will focus on generalizing these models across different buildings.