Imperial College London
Department of Earth Science and Engineering
MSc in Environmental Data Science and Machine Learning

Independent Research Project
Final Report

From Efficacy to Prediction: A Machine Learning Framework for Forecasting User Engagement in Automated Demand Response

Author:
Xiangbo Mai

Email: xm324@imperial.ac.uk
GitHub username: esemsc-xm324
Repository: https://github.com/ese-ada-lovelace-2024/irp-xm324

Supervisors:
Dr. Mirabelle Muûls
Dr. Jorge Avalos Patino
Dr. Shefali Khanna

August 2025

1. Abstract

Automated demand response (DR) programmes use smart switches and reward incentives to reduce household electricity use during peak periods. A central challenge is override behaviour, where devices remain on despite a switch-off request, which reduces actual savings. Existing studies report average effectiveness but rarely predict user-level outcomes and account for changes in engagement (fatigue) over time, limiting the operational planning and targeted design. This report develops a predictive framework for override behaviour and energy savings to inform scheduling and incentive setting in behaviour-aware DR operations. Using high-frequency smart-switch data linked with weather features, I train gradient-boosting models, a Temporal Convolutional Network (TCN) to capture recent-history patterns, and linear model as baselines. User segments from unsupervised clustering are fed back as features, and model explanations are based on SHAP value. Results show that reward rate is the dominant driver: higher rewards decrease the override risk by –3.6 percentage points across users and also drive energy savings. Demand levels matter, with higher daily energy use lowering override risk (–0.3 percentage points) but very high baseline power reducing savings. Weather conditions such as precipitation and solar radiation shape responsiveness, while fatigue effects emerge over time, with savings declining as weeks in trial increase. These findings highlight the value of adaptive incentives and varied scheduling for more reliable DR.

Acknowledgements

I would like to thank my supervisors, Dr. Mirabelle Muûls, Dr. Jorge Avalos Patino, and Dr. Shefali Khanna, for their invaluable guidance and feedback throughout this project. I am also grateful to the Department of Earth Science and Engineering at Imperial College London for providing the resources and support that made this work possible.

I would also like to thank my classmate, Daniel B. Kaupa, for helpful discussions and guidance during the course of this work.

Weather data were provided by the Copernicus Climate Change Service (C3S, ERA5), and the smart-switch trial data originate from the POWBAL randomised control trial (Khanna, Martin, & Muûls, 2025).

I used OpenAI's ChatGPT-4o (https://chat.openai.com/) and Google Gemini 2.5 Pro (https://gemini.google.com/app) to assist with debugging Python code, learn techniques for performance optimisation, and draft scaffolds for machine learning models. These generative AI tools supported my learning process; however, the submitted work is entirely my own and reflects my own understanding and effort, as declared in the Academic Integrity Declaration.

Contents

1. Abstract..... 2
2. Problem Description..... 5
2.1 Significance..... 5
2.2 Review of Existing Work..... 5
2.3 Research Aims and Objectives..... 6
2.3.1 User Profiling and Segmentation..... 6
2.3.2 Override Behaviour Prediction..... 6
2.3.3 Energy Savings and Fatigue Forecasting..... 7
2.3.4 Exploration of Incentive Strategies..... 7
3. Methodology..... 7
3.1 Data Preprocessing & Feature Engineering..... 7
3.1.1 Weather & Location Data Integration..... 7
3.1.2 Feature Selection, Imputation, & Encoding..... 9
3.1.3 Time Feature Engineering..... 10
3.1.4 Sequential Feature Engineering..... 11
3.1.5 Dataset Visualisation..... 12
3.1.6 Data Preprocess Pipeline..... 13
3.2 Customer Profiling & Feature Analysis..... 14
3.2.1 Preprocessing for PCA..... 14
3.2.2 PCA Clustering for Supervised Model Training..... 15
3.2.3 PCA Clustering for Customer Clustering & Profiling..... 15
3.3 Predictive Modeling: Override Behavior..... 16
3.3.1 Baseline Models (XGBoost, Regression)..... 17
3.3.2 LSTM Model..... 17
3.3.3 Comparison and Discussion..... 18
3.4 Machine Learning model for energy saving..... 18
3.4.1 Baseline Models (LightGBM, Regression, Linear & Ridge Regression)..... 19
3.4.2 Temporal Convolutional Network (TCN)..... 20
3.4.3 Comparison and Discussion..... 20
4. Results..... 21
4.1 Override Behaviour (XGBoost)..... 21
4.2 Energy Savings (LightGBM & TCN)..... 22
5. Discussion and Conclusions..... 23
5.1 Discussion and Conclusions..... 23
5.2 Limitations & Recommendations..... 24
Appendix..... 25
References..... 38

2. Problem Description

2.1 Significance

This report contributes to the operationalisation of automated DR by moving from efficacy to predictability. While prior studies showed that incentives reduce demand, they offered little guidance for forward-looking operations. By developing a predictive framework for override behaviour and event-level energy savings, this study equips utilities with tools to control risk, allocate incentives efficiently, and design behaviour-aware schedules.

The findings show that reward rate is the dominant driver, alongside load intensity, appliance type, and seasonal context. These insights support adaptive, segment-aware incentive schemes that balance immediate efficiency with sustained participation. More broadly, the study supports the development of Virtual Power Plants (VPPs) by embedding consumer behaviour into predictive models, and demonstrates the complementary value of gradient boosting and sequence models for energy applications. Overall, it strengthens the case for behavioural forecasting in demand-side management, offering a pathway to more flexible and cost-effective energy systems.

2.2 Review of Existing Work

The study by Khanna, Martin, and Muûls (2025) provides experimental evidence that automated, incentive-based demand response (DR) programmes reduce household electricity consumption without adverse side effects. However, their analysis is retrospective and based on a limited set of features, restricting its value for predictive or operational use.

The first limitation is the absence of long-term engagement modelling. Their regression framework examined event characteristics such as reward rate and notice period, with

user fixed effects included, but this assumes uniform user responses and does not capture time-varying effects such as behavioural fatigue. The second limitation is the treatment of incentives. The study reports no consistent evidence that higher reward rates lead to greater reductions in electricity use, suggesting diminishing marginal effects once automation is in place. Finally, while user fixed effects account for broad differences, the lack of clustering or segmentation prevents the identification of distinct behavioural archetypes. This restricts the design of targeted, adaptive interventions needed for scalable DR.

2.3 Research Aims and Objectives

The aim of this study is to develop a predictive framework for forecasting consumer behaviour in automated demand response (DR) programmes, focusing on override probability and energy savings. This framework supports forward-looking planning and identifies the main drivers of user response to switch-off events, enabling targeted interventions to reduce override risk and sustain savings.

The project is divided into two phases: (1) data preparation and segmentation, and (2) predictive modelling with feature interpretation. The objectives are:

2.3.1 User Profiling and Segmentation

Apply unsupervised clustering to capture behavioural archetypes such as high vs. low energy users or frequent vs. infrequent overriders. Clustering is conducted both with and without outcome-related features to avoid label leakage. These segments are then used as model inputs, improving predictive accuracy and enabling targeted interventions.

2.3.2 Override Behaviour Prediction

Build machine learning models to forecast override probability. Previous work measured override rates retrospectively but did not aim to predict them. Accurate forecasts allow operators to anticipate override risk in advance and understand which factors drive user responses.

2.3.3 Energy Savings and Fatigue Forecasting

Use sequence-based models to predict event-level savings and quantify behavioural fatigue. Temporal features such as time-in-trial and repeated exposure, alongside contextual factors like reward rate, are modelled to capture how savings decline over time. This provides crucial input for long-term DR scheduling.

2.3.4 Exploration of Incentive Strategies

Simulate different reward structures to assess their effect on participation and savings. Since prior research shows incentives may not have a linear impact, this objective tests how varying reward rates influence behaviour and offers guidance for optimising incentive design.

3. Methodology

This research adopts a multi-faceted, data-driven methodology to systematically deconstruct and model user behaviour. The workflow includes data preprocessing and exploratory data analysis (EDA) to ensure quality and detect patterns, PCA-based clustering for feature engagement and user profiling, and predictive modelling with training, comparison, and optimisation across different algorithms.

3.1 Data Preprocessing & Feature Engineering

3.1.1 Weather & Location Data Integration

To predict override behaviour and energy savings, the smart-switch dataset (*powbal_clean*) was enriched with weather and spatial features. Household responses in automated DR depend not only on individual preferences but also on environmental factors such as temperature, solar radiation, and precipitation, which shape heating, cooling, and lighting demand. Geographic location further determines climatic exposure, requiring spatially matched weather integration.

Hourly reanalysis data were obtained from the Copernicus Climate Change Service ERA5 single-level product (C3S, 2017). The dataset was restricted to Delhi (28–29°N, 76–78°E) and Mumbai (18–20°N, 72–73°E) for 2023–2024 to align with the trial. Records were merged with the smart-switch data using timestamp and geolocation (longitude, latitude).

The weather features were selected based on their established relationships with residential energy demand and DR participation:

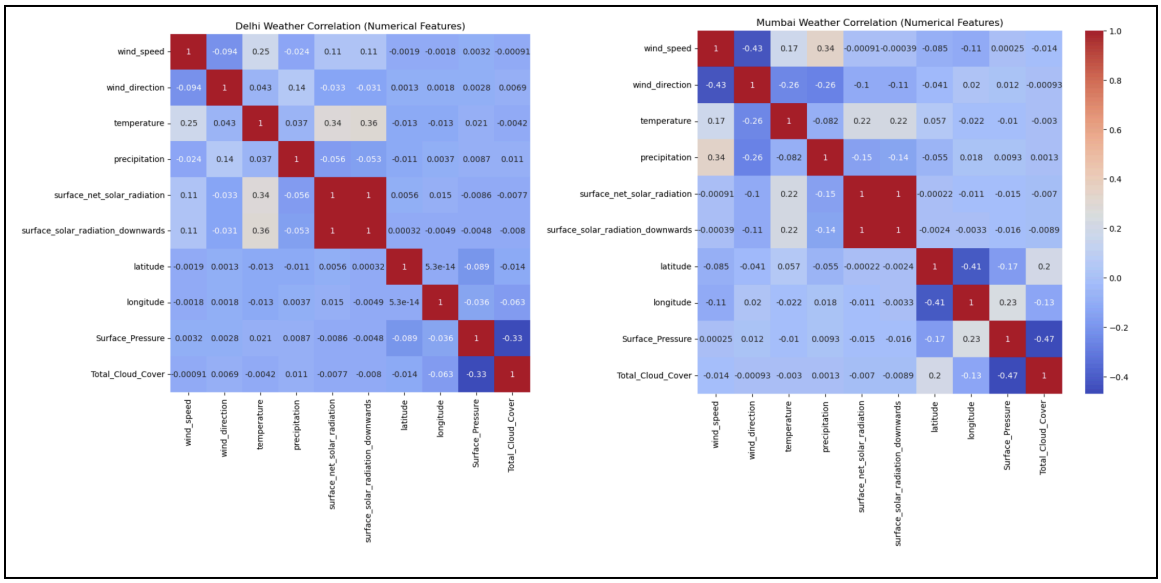
- *Air temperature* — affects heating and cooling load, as widely documented in energy demand studies (Ahmad et al., 2020; Lam et al., 2008).
- *Cloud cover and solar radiation* — influence lighting demand and solar heat gain (Siano, 2014; Gellings, 2021).
- *Wind components (U/V)* — used to compute wind direction and speed, which shape natural ventilation and cooling requirements (Lam et al., 2008).
- *Precipitation and surface pressure* — affect occupancy behaviour and ventilation patterns, impacting residential demand response (Khanna et al., 2025).

To ensure data quality, missing values were filled using linear interpolation, applied per feature and per location across time. For example, ERA5 often omits hourly values at the start of each day (e.g., *weather-delhi-2020*), and these gaps were smoothed under the assumption of continuous weather evolution. For users with missing location information, data were imputed using the average location of all users in the same city (Delhi or Mumbai), enabling every record to be spatially matched with weather data. Feature transformations included converting temperature from Kelvin to Celsius, calculating wind direction from vector components, and removing highly correlated variables. For instance, *surface_solar_radiation_downwards* was dropped due to its near-perfect correlation with *surface_net_solar_radiation* (see Figure 1).

Another challenge was the mismatch between user coordinates and available ERA5 grid points. This was addressed through spatial interpolation, which generated smoothed values to ensure that each user could be matched with a complete and continuous set of weather features. This step improved both coverage and robustness.

Integrating weather and location data in this way enhances the explanatory power of the predictive models by embedding environmental context. The final processed and interpolated dataset (*weather_full.csv*) was saved separately from the smart-switch records to maintain modularity and reusability for future analyses.

Figure 1. Correlation matrix of weather features for Delhi and Mumbai.



3.1.2 Feature Selection, Imputation, & Encoding

After integrating weather and spatial variables, the combined dataset contained 128 features and more than seven million records. While this richness increases modelling flexibility, it also introduces redundancy, noise, and computational inefficiency. To address this, a structured feature selection process was applied to retain informative variables while removing those that were duplicate, constant, or low in variance. A full list of excluded features and justifications is provided in Table 1.

To prepare the dataset for modelling, a targeted imputation strategy was implemented, tailored to feature type and context:

- **Numerical Features with Moderate Missingness.** For usage-related variables such as *pre_switch_off_reading* and *period_duration*, missing values were

imputed with the median. This approach is robust to outliers and captures the central tendency of user habits.

- **Zero-Imputation for Operational or Derived Variables.** Engineered variables such as *power_nonzero*, *average_power_before_switchoff*, *mean_W_before_switchoff*, *avoided_energy_consumption_Wh*, and *reward* were structurally missing when no switch-off event occurred. In these cases, missing values were replaced with zero to represent non-participation without adding spurious variance.
- **Categorical Variables Imputed with Mode.** For *appliance* and *preference*, missing values were filled using the most frequent category, reflecting dominant user behaviour patterns.

Following imputation, additional highly correlated features were removed to mitigate multicollinearity, consistent with the preprocessing of weather variables. The correlation matrix (see Figure 2) illustrates the relationships, and Table 2 lists the excluded variables.

Together, these steps produced a cleaner, more interpretable dataset for model training while preserving essential variation in user behaviour and event conditions.

3.1.3 Time Feature Engineering

To capture daily and seasonal patterns in user behaviour and energy use, cyclical time features were derived from the raw timestamp variable *round_datetime*. Temporal variables such as hour of day, day of week, and month are inherently periodic and not well represented by raw numerical values. For example, 23:00 and 01:00 are numerically far apart but close in time. To address this, sine and cosine transformations were applied to map time components onto the unit circle, effectively encoding their cyclic structure (Géron, 2019).

The following temporal components were transformed: *hour* (cycle = 24), *day_of_week* (cycle = 7), *month* (cycle = 12), and *day_of_year* (cycle = 365). Each was encoded as both sine and cosine values to ensure continuity at cycle boundaries (e.g., from December and January, Sunday and Monday).

Each component was encoded using the following transformations:

$$feature_{sin} = sin(2\pi \cdot \frac{value}{cycle\ length}),\ feature_{cos} = cos(2\pi \cdot \frac{value}{cycle\ length})$$

After transformation, raw timestamp-related columns (*round_datetime*, *date*, *week*, *half_hour*) were removed so that only the derived sine–cosine features remained in the final dataset. This encoding allows models to capture intra-day (hourly) and seasonal (weekly, monthly, yearly) patterns that may be important drivers of override probability and energy savings in automated DR programmes.

3.1.4 Sequential Feature Engineering

To reduce the dimensionality of high-frequency event data while preserving essential temporal dynamics, sequential features were compressed into interpretable summary statistics using a custom transformer (*AdvancedFeatureCompressor*). This method converts rolling event windows (such as historical and forecasted switch-off signals, notice times, and reward rates) into engineered features with clear behavioural and temporal meaning.

The original dataset contained sequences of binary indicators for past and future switch-off events, labelled *switch_off_L1* to *L6* (upcoming events) and *switch_off_F1* to *F16* (recent past events). Similarly, *notice_time_F1* to *F16* and *reward_rate_F1* to *F3* represented the notice period and reward rate for scheduled future events.

From these sequences, the compressor generated the following summary features:

- *future_switch_off_count*: total number of upcoming switch-off events (sum over *switch_off_L*).
- *past_switch_off_count*: number of recent past switch-off events (sum over *switch_off_F*).
- *time_to_next_switch_off*: index of the first non-zero in *switch_off_L*, indicating how soon the next event occurs.
- *time_since_last_switch_off*: index of the first non-zero in *switch_off_F*, showing how recently the last event occurred.
- *max_future_notice*: maximum notice time among the next 16 scheduled events.
- *time_to_first_notice*: timing of the first upcoming notice with a non-zero value.
- *mean_future_reward*: average reward rate across the next 3 scheduled events.
- *std_future_reward*: standard deviation of reward rates across the same horizon, capturing variability.

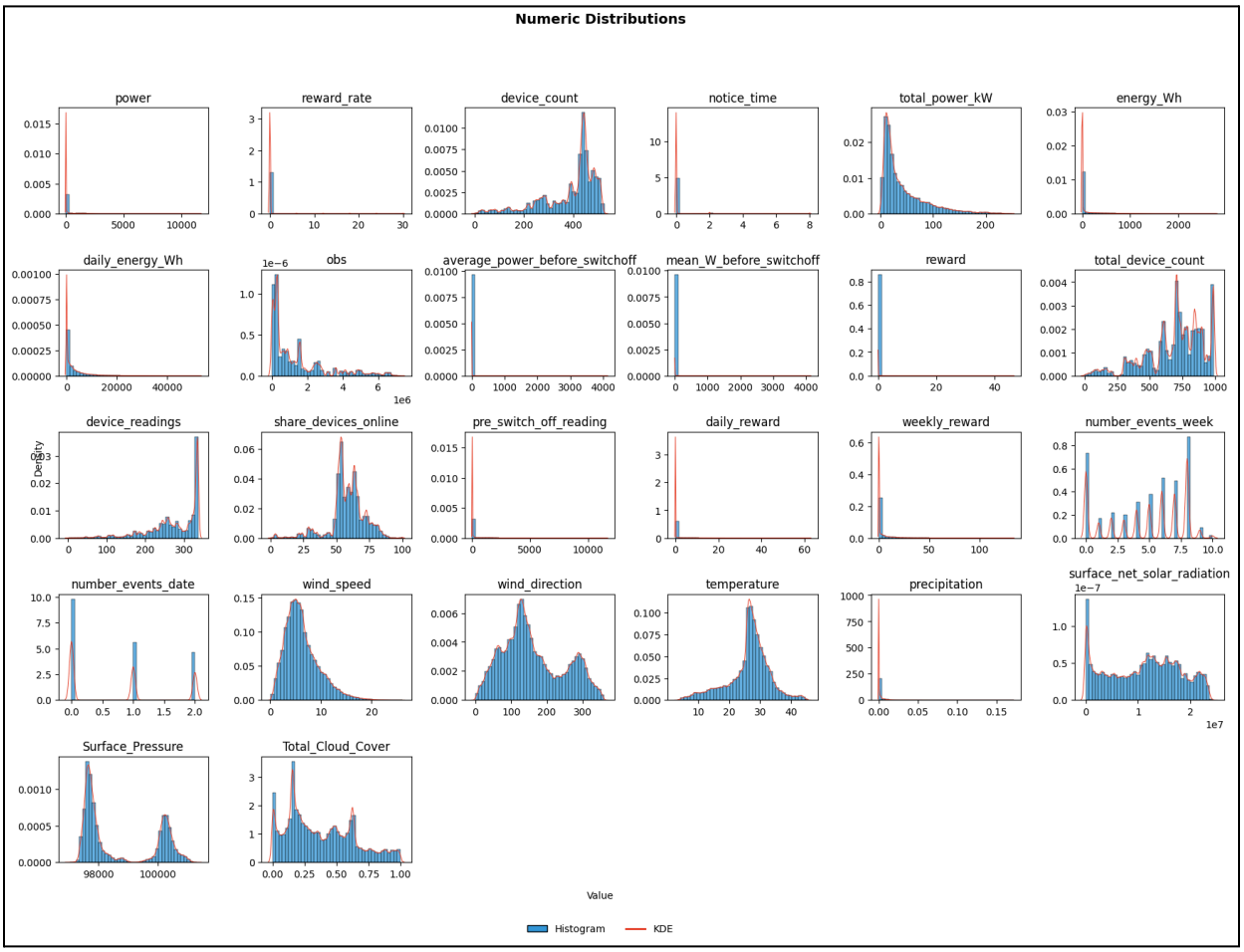
After these transformations, the original sequence columns were removed to reduce redundancy. This approach preserves temporal richness (e.g., exposure frequency, timing, and stimulus variation) while compressing the data into features that are more suitable for predictive modelling.

3.1.5 Dataset Visualisation

The plot (see Figure 3) presents the distribution of all numeric features in the integrated user dataset (*powbal_clean* with weather features). Each subplot shows a histogram (blue) overlaid with a Kernel Density Estimate (KDE) curve (red), providing insight into the shape, spread, and skewness of each variable.

The visualisations indicate that most features fall within reasonable ranges, with no extreme outliers likely to distort model training. Overall, the distributions confirm that the dataset is suitable for standardisation and predictive modelling.

Figure 3. Distribution of Numeric Features in the Integrated Dataset (powbal_clean with weather features).



3.1.6 Data Preprocess Pipeline

The preprocessing pipeline combines both time-based and sequential feature transformations to capture temporal and event-history dynamics. Following these steps, standard preprocessing is applied: numerical features are standardized, categorical features are one-hot encoded, and binary or identifier variables are passed through unchanged.

3.2 Customer Profiling & Feature Analysis

This section applies two Principal Component Analysis (PCA)-based strategies to uncover patterns in user behaviour: one for exploratory customer segmentation and the other for predictive model training. In both cases, PCA is used for dimensionality reduction, followed by K-means clustering. Results are compared with Density-Based Spatial Clustering of Applications with Noise (DBSCAN) to test robustness. The optimal number of clusters (*K*) is determined using the silhouette score, providing a balance between model compactness and clear cluster separation.

3.2.1 Preprocessing for PCA

To generate customer-level inputs for clustering, raw event-level features were aggregated by user. Binary and one-hot categorical features were summarised using their means and standard deviations, while numeric and weather variables were aggregated using both mean and standard deviation. Time-of-use behaviour was encoded with circular transformations (e.g., *hour_sin*, *hour_cos*) to preserve periodicity. From these encodings, features such as *typical_hour*, *hour_concentration*, *typical_dow*, and *dow_concentration* were derived to capture both average activity and consistency across the day or week.

The standard deviations of cyclical encodings were dropped, as they lack consistent physical interpretation. For example, the standard deviation of *hour_sin* and *hour_cos* can misrepresent variability when activity spans midnight, producing spurious variance due to circular discontinuity (Géron, 2019; Jammalamadaka & SenGupta, 2001). Removing these features reduces artificial noise and improves clustering quality.

Additional user-level features were also computed, including weather sensitivity indicators (e.g., correlations between power and temperature, solar radiation, or wind) and event-level summaries such as average future reward and time-to-switch-off. To ensure robust modelling, the predictive strategy excluded any features containing future knowledge or post-outcome signals (e.g., actual reward received, post-event device status), thereby preventing label leakage.

3.2.2 PCA Clustering for Supervised Model Training

This strategy applies clustering on leak-safe features to create reliable and interpretable inputs for supervised learning. PCA reduced dimensionality while retaining 95% of variance, resulting in 18 principal components. K-means clustering identified an optimal solution at $K = 2$ (silhouette = 0.36), producing a relatively meaningful separation of behavioural patterns (see Table 3, Subplot 1). In contrast, DBSCAN generated highly imbalanced clusters, with over 99% of users grouped into a single cluster (see Table 3, Subplot 2), confirming K-means as the more effective method for predictive feature creation.

A heatmap of z-scored cluster means (see Table 3, Subplot 3) highlights the key discriminative features: reward_rate, number_events_date, notice_time, and circular time-of-use features such as typical_hour. These results suggest that override behaviour might be shaped comprehensively by incentive structures and scheduling patterns. Incorporating these clusters as inputs to supervised models provides a structured way to capture user diversity, with potential to improve predictive accuracy.

3.2.3 PCA Clustering for Customer Clustering & Profiling

This strategy uses the full feature set, including post-event behaviours and realised weather, to generate a comprehensive view of user behaviour. PCA reduced dimensionality while retaining 99% of variance, resulting in 24 principal components. K-means clustering with $K = 3$ (silhouette = 0.20) captured diverse engagement and response patterns (Table 3, Subplot 4). By contrast, DBSCAN again produced excessive noise (Table 3, Subplot 5), reinforcing K-means as the preferred method for customer profiling.

The heatmap of z-scored cluster means (Table 3, Subplot 6) highlights clusters differentiated by energy intensity, reward sensitivity, and override frequency. These segments can be interpreted as behavioral patterns, such as frequent overrider,

low-energy users, and reward-focused participants. Such segmentation provides actionable insights for designing targeted interventions and engagement strategies. Overall, this approach demonstrates strong potential for advancing customer profiling and supporting personalised demand response programmes.

3.3 Predictive Modeling: Override Behavior

This section evaluates supervised learning models for predicting override behaviour, where users keep devices on despite a switch-off request. The prediction task uses the cleaned dataset from the preprocessing pipeline and the label feature *cust_seg_safe* generated from PCA with K-means.

The dataset was split chronologically: the first 70% of events for training, the next 15% for validation, and the final 15% for testing. A chronological split was chosen because override behaviour is time-dependent, with engagement evolving over the trial due to fatigue, adaptation, or changing incentives. This prevents information leakage from future to past events and mirrors real-world deployment, where models must forecast future behaviour from historical data.

In addition to general features, fatigue-related features were included to capture long-term participation trends. For example, *week_in_trial* measures progression through the trial, *cumulative_event_count* tracks exposure, and short-term indicators such as *events_in_last_7_days* and *override_rate_in_last_7_days* reflect recent behavioural history. These features allow the models to capture both gradual fatigue and short-term adaptation.

Model performance was assessed using Area Under the Precision–Recall Curve (AUPRC), which is more informative than accuracy or ROC-AUC given the imbalance between override and non-override events. AUPRC evaluates how well rare override events are identified, balancing precision (avoiding false alarms) and recall (capturing true overrides).

Baseline and advanced models were tested, with results summarised in Table 4. Among them, XGBoost performed best, achieving a test AUPRC of 0.803 and an F1 score of 0.731, outperforming both logistic regression and the LSTM sequence model.

3.3.1 Baseline Models (XGBoost, Regression)

The first step applied traditional machine learning methods to establish baseline predictive performance. XGBoost, a gradient-boosting tree ensemble, consistently outperformed other baselines. In its optimised setting, the model achieved VALID AUPRC = 0.866 and TEST AUPRC = 0.803, with an extremely high ROC-AUC (0.999). At the F1-optimal threshold, the override class reached precision = 0.781, recall = 0.688, and F1 = 0.731. This represents strong performance given the rarity of override events.

By contrast, logistic regression achieved a TEST AUPRC of only 0.232, showing that linear models cannot capture the non-linear patterns driving override behaviour. The gap in performance highlights the suitability of tree-based methods for this task.

Explainability analyses reinforce the conclusion that both SHAP and permutation importance identified reward rate as the dominant driver of override probability, followed by event frequency (*number_events_date*), baseline energy(*energy_Wh*, *daily_energy_Wh*), and incentives (*daily_reward*, *weekly_reward*). Fatigue-related features, particularly *override_rate_in_last_7_days*, also ranked highly. These results demonstrate that the model captures both structural incentives and behavioural adaptation, providing interpretable evidence of the mechanisms shaping override behaviour.

3.3.2 LSTM Model

To capture temporal dependencies in user activity, a Long Short-Term Memory (LSTM) neural network was implemented. Unlike static baseline models, LSTMs process sequences of events, enabling the model to learn patterns such as recent override streaks, fatigue effects, and engagement decay.

However, performance was weak compared to XGBoost. The LSTM achieved only TEST AUPRC = 0.013, with very low precision and recall for the override class. While the ROC-AUC was moderate (≈ 0.86), this metric overstates performance under class imbalance. The results indicate that the LSTM was unable to balance precision and recall effectively and struggled to detect rare override events.

3.3.3 Comparison and Discussion

The results show that although deep sequence models are theoretically well suited to capturing behavioural dynamics, in practice they require larger datasets, more balanced class distributions, or advanced re-weighting to perform effectively. In the present setting, XGBoost proved far more robust, delivering both higher predictive accuracy and greater interpretability.

As summarised in Table 4, XGBoost achieved a test AUPRC of 0.803 and an F1 score of 0.731, outperforming both the logistic regression baseline and the LSTM sequence model. These findings highlight the strength of tree-based ensembles in handling structured behavioural data, while also confirming the limitations of applying deep learning under severe class imbalance.

3.4 Machine Learning model for energy saving

This section evaluates models for predicting event-level energy savings and analysing feature contributions. Energy savings were defined as the difference between expected baseline consumption and observed consumption during a switch-off event. The baseline was estimated as the average device load before the event (*pre_switch_off_reading*) scaled by event duration (*period_duration*):

$$E_{baseline} = pre\_switch\_off\_reading \times period\_duration$$

The realised savings were then calculated as:

$$Energy\ Savings = E_{baseline} - E_{observed}$$

which $E_{observed}$ is the measured energy use during the event. If the device remains off, savings are maximised; if overridden, savings are reduced. The target, expressed in kWh, serves as the dependent variable for regression and sequence models.

As in the PCA strategy, strict care was taken to avoid information leakage. Pre-notice features such as *pre_switch_off_reading*, *average_power_before_switchoff*, and *mean_W_before_switchoff* directly determine baseline consumption and were excluded from the non-leaky feature set used for fair model evaluation.

The task was framed as a feature influence analysis problem, comparing baseline models (LightGBM, Linear Regression, Ridge Regression) with a sequence model (Temporal Convolutional Network, TCN). Since the target is continuous, performance was assessed using Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE). RMSE captures sensitivity to extreme cases, while MAE reflects overall predictive accuracy.

3.4.1 Baseline Models (LightGBM, Regression, Linear & Ridge Regression)

Baseline models were first applied to provide reference performance and interpretable insights. LightGBM achieved the strongest predictive accuracy under strict pre-notice settings, with near-zero error (Test RMSE = 0.0000, MAE = 0.0000). This performance justified its use as the primary benchmark for tabular prediction. Feature importance analysis consistently highlighted reward-related variables, baseline load (*power*, *pre_switch_off_reading*), and recent event number as dominant drivers of savings.

Linear and ridge regression produced slightly weaker accuracy (Test RMSE ≈ 0.0001, MAE ≈ 0.0000) but offered coefficient-level interpretability. Their results supported LightGBM's findings, confirming that appliance type (e.g., car charging sockets) and

daily energy use also contributed to variation in energy savings. These models were retained as secondary baselines for interpretability rather than competitive prediction.

3.4.2 Temporal Convolutional Network (TCN)

To capture sequential dependencies in energy savings, a Temporal Convolutional Network (TCN) was trained on rolling event sequences. After hyperparameter optimization, the TCN achieved a Test RMSE of 0.0021 and MAE of 0.00049. Although this error was higher than LightGBM, performance remained within an acceptable range for exploratory sequence modelling.

Unlike baseline models, the TCN was able to incorporate temporal and fatigue-related dynamics, learning patterns across consecutive events. Feature attribution showed that appliance-specific behaviour, exposure over time(*week_in_trial*, *number_events_week*), and seasonal encodings (*month* and *day-of-year* cycles) played important roles. Thus, the TCN adds interpretive value by capturing adaptive and sequential effects that static models cannot.

3.4.3 Comparison and Discussion

The comparison demonstrates that LightGBM offers superior predictive accuracy, making it the most reliable model for short-term energy savings prediction. Linear and ridge regression add value through interpretability, validating the influence of incentives, load, and appliance type. By contrast, although the TCN underperformed slightly in accuracy, it contributes methodological value by revealing sequential dynamics such as fatigue, seasonal variation, and appliance-specific patterns.

This trade-off between accuracy and behavioural insight supports a dual-model strategy: LightGBM for high-fidelity predictions, and TCN for uncovering long-term behavioural drivers.

The performance of all energy savings prediction models under strict pre-notice settings is summarised in Table 5.

4. Results

To interpret feature contributions, I apply SHapley Additive exPlanations (SHAP). SHAP values quantify how much each feature shifts the model's predicted probability, both in direction and magnitude. To improve interpretability, I extend the analysis with four indices (all probability effects are expressed in percentage points, pp):

- *mean_abs_SHAP_logit*: overall importance of a feature, regardless of sign.
- *mean_delta_prob_pp*: the average change in predicted probability.
- *median_delta_prob_pp*: the typical effect, less sensitive to extremes.
- *p5..p95_delta_prob_pp*: the range of effects across most predictions, showing whether the feature's impact is consistent or heterogeneous (see Table 6, Subtable 1).

4.1 Override Behaviour (XGBoost)

The XGBoost model identifies *reward_rate* as the dominant driver of override probability (see Table 6, Subtable 1; Table 6, Subplot 1). It has the largest average influence (mean_abs_SHAP_logit = 5.27), consistently lowering override probability (mean_delta_prob_pp = –3.56pp, median = –3.24pp). The effect is strong and stable, with most cases between –10.64pp and –0.81pp. This confirms that higher rewards make overrides less likely in nearly all situations.

Other demand-side features also contribute. *daily_energy_Wh* shows moderate importance (mean_abs_SHAP_logit = 0.43), linked to a reduced override likelihood (mean_delta_prob_pp = –0.30pp). Similarly, *number_events_date* (–0.06pp) and *energy_Wh* (–0.13pp) both have consistent negative effects, indicating that heavier energy usage lowers the probability of ignoring switch-offs.

Behavioral history provides additional signals. *override_rate_in_last_7_days* decreases override probability by –0.05pp on average, though with a very narrow range (–0.00..0.01pp), making its effect modest but reliable. Incentive accumulation variables

such as *daily_reward* (–0.07pp) and *weekly_reward* (–0.01pp) also appear, but with weaker effects.

Overall, override behaviour is primarily shaped by incentive strength (*reward_rate*) and energy demand conditions(*daily_energy_Wh*, *energy_Wh*, *number_events_date*), with historical engagement indicators adding modest but consistent predictive value.

4.2 Energy Savings (LightGBM & TCN)

For energy savings, the LightGBM model highlights load- and reward-related variables as dominant drivers (see Table 7, Subtable 1; Table 7, Subtable 3, left). *reward* shows the highest average importance (mean_abs_SHAP_logit = 2.84e–05), linked to a reduction in predicted savings (mean_delta_prob_pp = –1.25e–04, median = –4.34e–04). *power* and *reward_rate* also contribute strongly, with consistent negative effects. Daily-level aggregates such as *daily_reward* and *daily_energy_Wh* provide secondary but stable contributions. Together, these findings confirm that energy savings are closely tied to load intensity and reward assignment.

By contrast, the TCN model distributes importance more broadly (see Table 7, Subtable 2; Table 7, Subtable 3, right). Seasonal encodings such as *month_sin* and *day_of_year_sin* increase predicted savings (mean_delta_prob_pp ≈ +1.2e–05), capturing seasonal variation in user responsiveness. Contextual weather signals, including *precipitation* (+1.34e–05) and *surface_net_solar_radiation* (+5.68e–06), also play a significant role. Unlike LightGBM, the TCN captures variation linked to seasonality and environmental context.

In conclusion, LightGBM concentrates importance on reward and demand features, producing narrow and stable effects on savings. Meanwhile, the TCN uncovers seasonal, appliance, and weather influences, consistent with its strength in modelling temporal dynamics.

5. Discussion and Conclusions

5.1 Discussion and Conclusions

This study built a predictive framework to forecast override behaviour and energy savings in automated DR. It used high-frequency smart-switch data enriched with weather and contextual features, applied gradient boosting, TCN, and linear baselines, and explained results through SHAP values.

Key Findings including:

- Rewards matter most
    - Higher rewards cut override risk by –3.6pp on average, consistently across users.
    - Rewards and load also drive energy savings, confirming incentives as the strongest tool for programme design.
- Demand levels shape outcomes
    - Higher daily energy use lowered override risk by –0.3pp.
    - Very high baseline power reduced predicted savings, showing diminishing returns.
    - Events should be targeted at times of solid but not extreme demand.
- Season and weather add context
    - Seasonal patterns and weather features (precipitation, solar radiation) influenced savings.
    - Higher precipitation (rainfall) was linked to increased savings.
    - Weather features (precipitation, solar radiation, temperature) were meaningful predictors of energy savings
- Fatigue reduces long-term performance
    - Savings declined with more weeks in trial, showing engagement fatigue.
    - Override history mattered: recent exposure changed compliance patterns.
    - Without adaptive incentives, long-term participation might decrease.

In conclusion, override probability and savings can be shaped by careful event design. Incentives remain the strongest lever, demand signals guide timing, weather and season refine targeting, and fatigue requires adaptive scheduling. Together, these findings offer utilities a predictive framework to anticipate risks and optimise demand response programmes for more reliable and cost-effective flexibility.

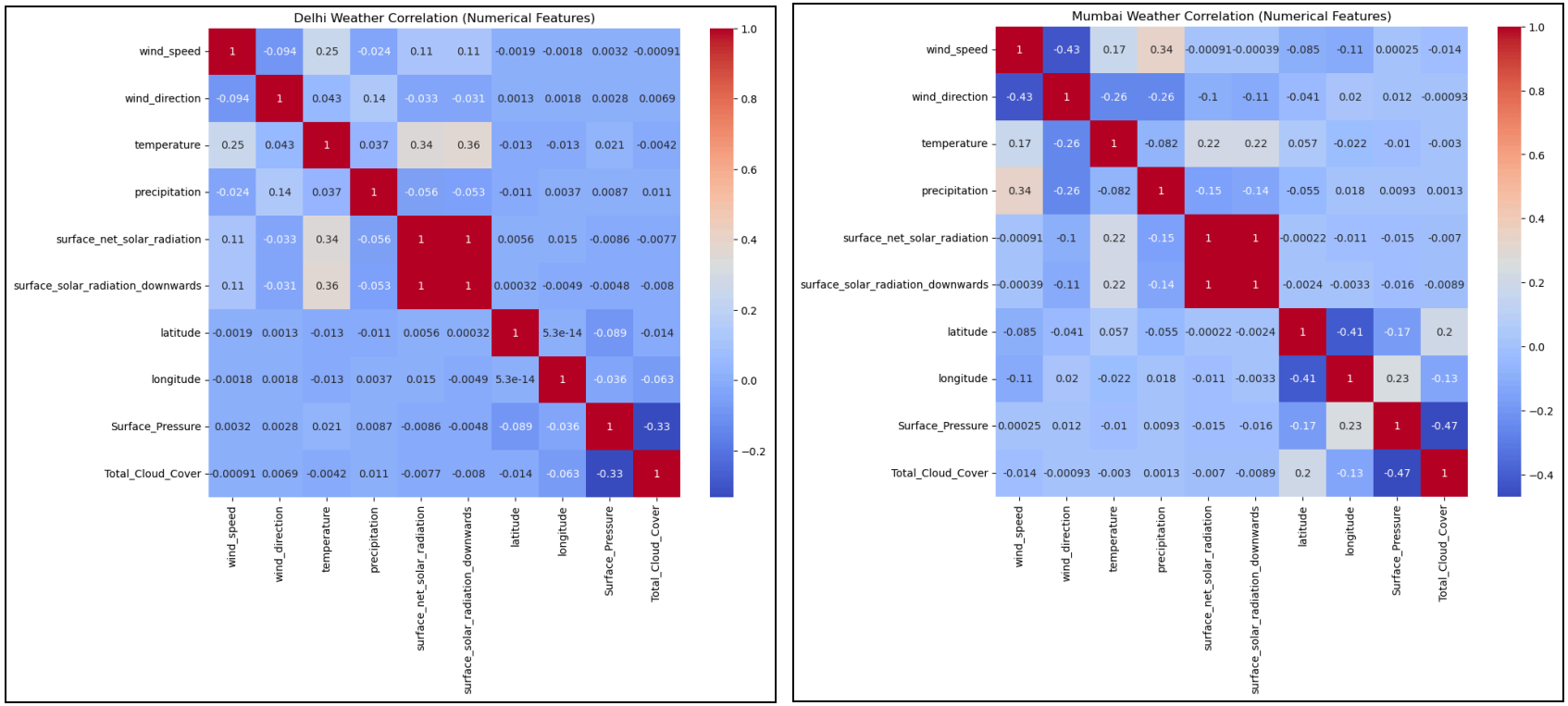
5.2 Limitations & Recommendations

Even though this report analyses feature effects on override and energy savings, several gaps remain. First, the interpretation of feature meaning is constrained by the predictive models, so some conclusions may be less reliable than others. Second, this study did not use household-level power consumption because too many records were missing. Third, while SHAP enhances interpretability, it does not establish causal mechanisms, and its insights are limited to the situations observed.

Future studies should improve data quality to enable household-level power analysis, even if this requires dropping some users or records. Incorporating household-level data would provide more meaningful and reliable predictions, especially for cost-based evaluations. Extending the modelling beyond correlations to causal or simulation-based approaches would help identify mechanisms behind user behaviour, strengthening the basis for optimisation. Finally, combining predictive accuracy with causal insight would support more robust and adaptive design of demand response programmes.

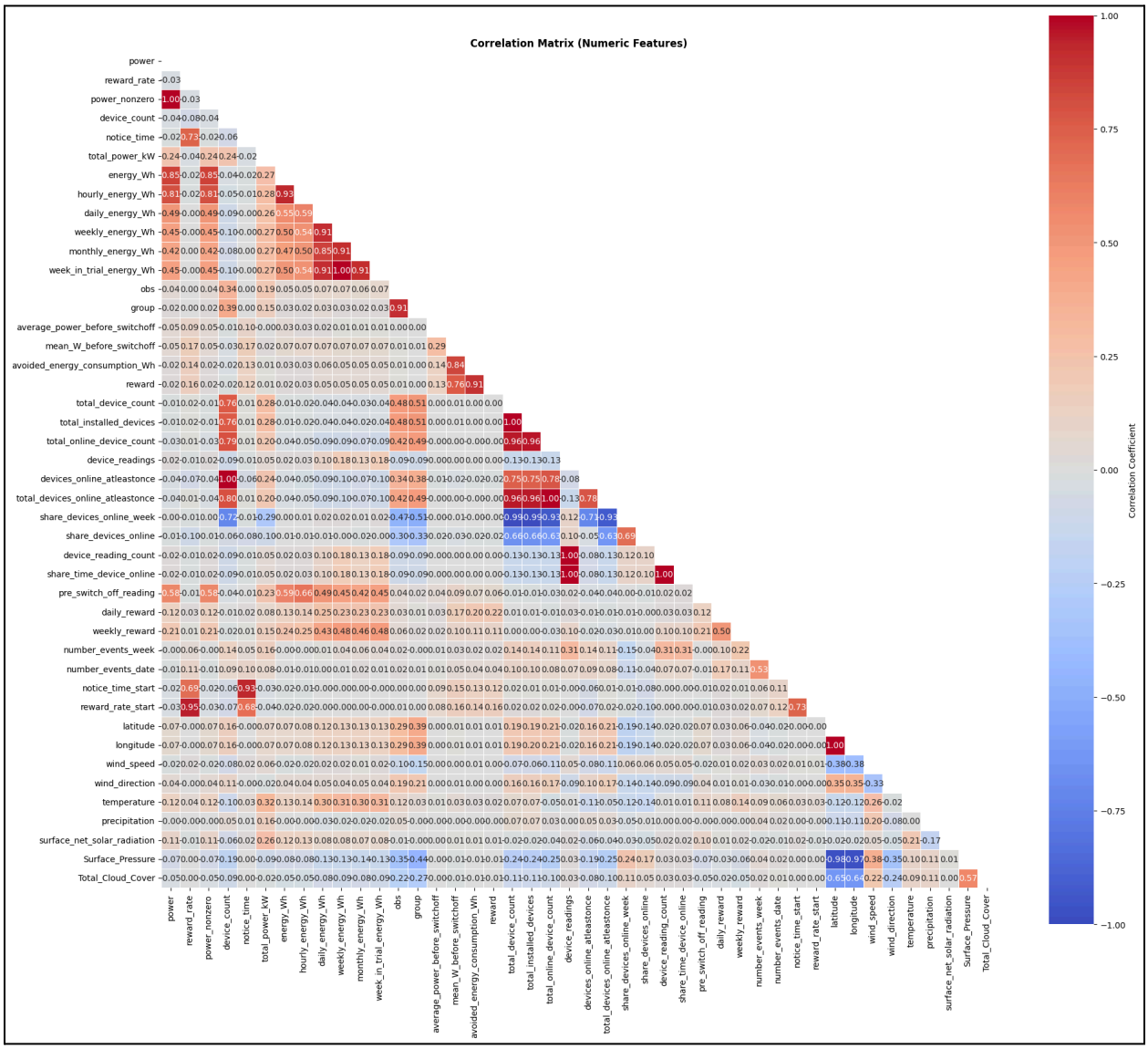
Appendix

Figure 1. Correlation matrix of weather features for Delhi and Mumbai



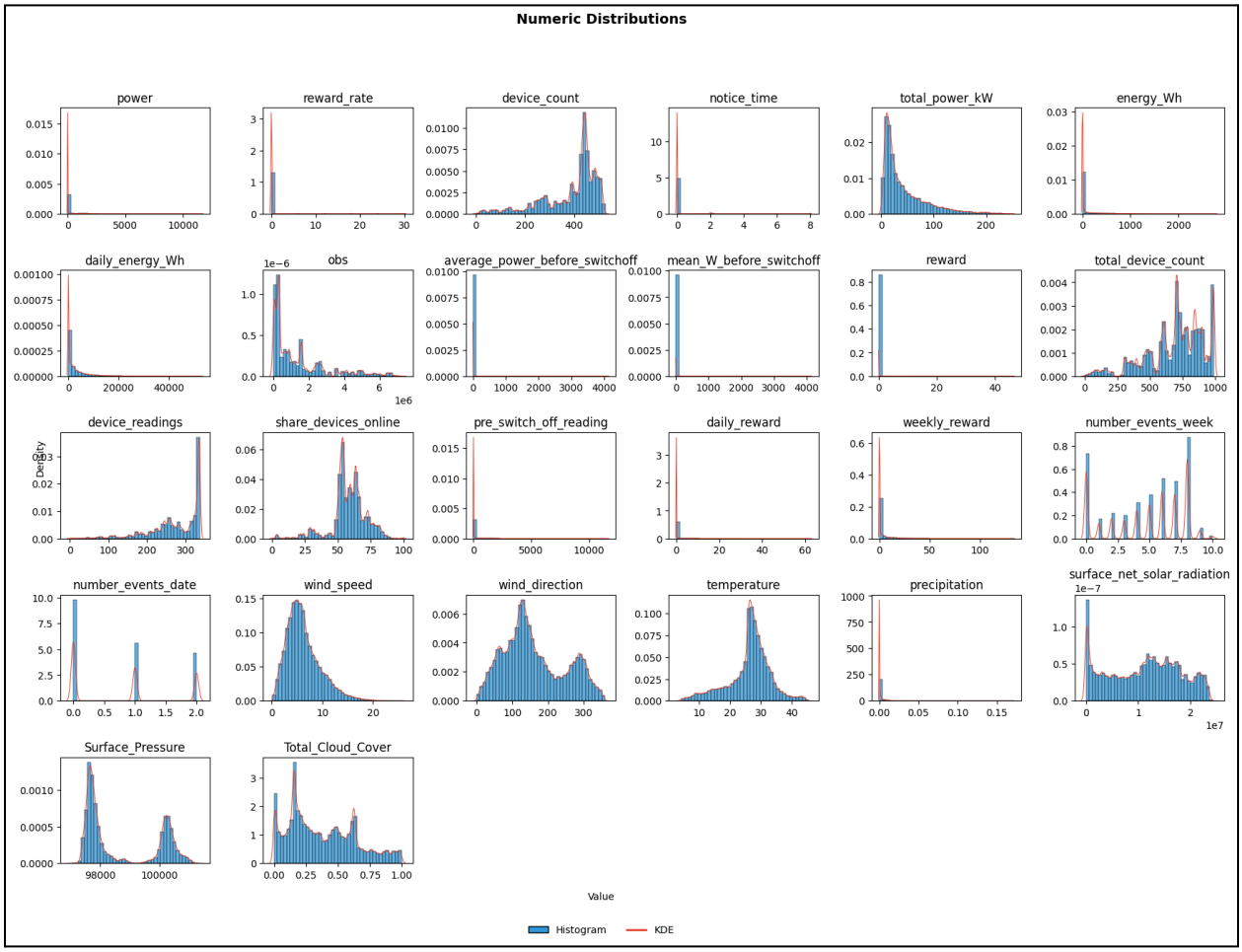
Note. This figure shows the pairwise correlations among weather variables used in the preprocessing stage. *Surface_solar_radiation_downwards* was removed due to its near-perfect correlation with *surface_net_solar_radiatio*

Figure 2. The Correlation Matrix of Numerical Features from Integrated Dataset (powbal_clean with weather features).



Note. This figure displays correlations across all numerical variables in the combined dataset. Several features were removed due to redundancy or high correlation (see Table 2)

Figure 3. Distribution of Numeric Features in the Integrated Dataset (powbal_clean with weather features).



Note. The histograms (blue) with overlaid kernel density estimates (red) display the distributions of all numeric variables used in model training.

Table 1. Features Dropped During Preprocessing, Categorized by Type of Redundancy or Lack of Informational Value

Features Dropped	Category	Reasons
device_online_atleastonce, _fillin	Constant Feature	No variance (only 1s or 0s)
device_id, registered_at	Low-Info Feature	Not meaningful or redundant with other variables
ca_number_str	Duplicate: ID	Same as ca_number
timestamp, datetime, date, month, week, week_of_year, month_of_year	Duplicate: Time	Covered by round_datetime
timestamp_id, minute_all, minute_str, half_hour_id	Duplicate: Time	Derivable or redundant
first_occurence, first_occurence_date, first_occurence_week	Duplicate: Time	Same as week_in_trial
total_power	Duplicate: Power Unit	Same as total_power_kW
energy	Duplicate: Energy Unit	Same as energy_Wh
hourly_energy_kWh, daily_energy_kWh, weekly_energy_kWh, monthly_energy_kWh, week_in_trial_energy_kWh	Duplicate: Energy Unit	Wh version retained

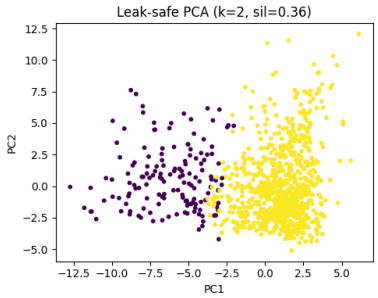
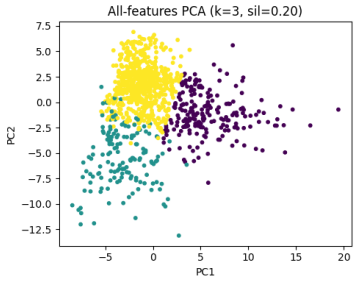
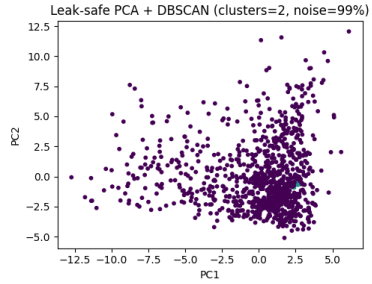
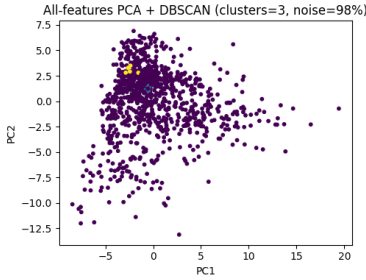
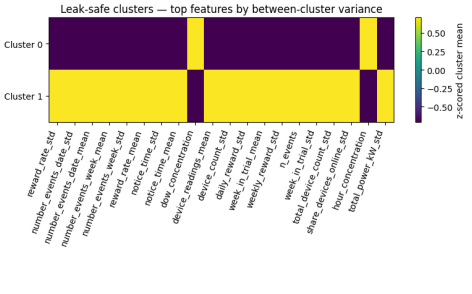
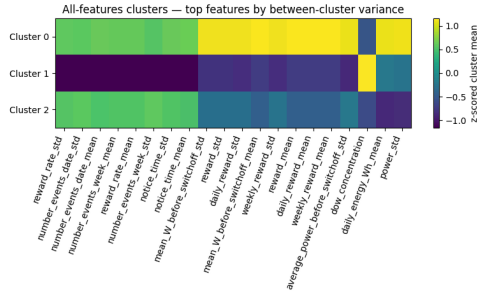
Note. Features were excluded when they showed no variance (constant features), carried little or no informational value, or were duplicates of other variables (e.g., different units or redundant time encodings). This step reduced dimensionality and ensured that only informative variables were retained for modelling.

Table 2. Feature Reduction Based on High Correlation, Grouped by Category

Features Dropped	Category	Reasons
hourly_energy_Wh, *weekly_energy_Wh,* *monthly_energy_Wh,* *week_in_trial_energy_Wh*	Energy Metrics	Highly correlated with *daily_energy_Wh*; cumulative metrics across overlapping time windows
total_installed_devices, *total_online_device_count,* *share_time_device_online,* *device_reading_count,* *share_devices_online_week,* *total_devices_online_atleastonce,* *notice_time_start,* *reward_rate_start,* *avoided_energy_consumption_Wh*	Device & Activity Counts	Near-perfect correlation; all describe similar aspects of device usage, connectivity, or event setup
power_nonzero	Binary Power Activity	Strong correlation with target variables *power*, *reward_rate*; risk of data leakage or overfitting

Note. Features were removed when they showed strong or near-perfect correlations with other variables, leading to redundancy or risk of information leakage.

Table 3. Comparison of Clustering Methods for Feature Engineering and Customer Profiling

	For Supervised Model Training	For Customer Clustering & Profiling
PCA with K-means	 **Subplot 1**	 **Subplot 4**
PCA with DBSCAN	 **Subplot 2**	 **Subplot 5**
K-means Cluster Profile Heatmap	 **Subplot 3**	 **Subplot 6**

Note. This table compares PCA-based clustering strategies for supervised model training (Subplots 1–3) and customer profiling (Subplots 4–6). K-means produced balanced clusters with meaningful separation, while DBSCAN generated highly imbalanced groups dominated by noise. Heatmaps (Subplots 3 and 6) show the top discriminative features for each cluster, highlighting differences in reward rate, energy usage, and scheduling patterns. These results confirm K-means as the more effective clustering method for both predictive feature creation and behavioural segmentation.

Table 4. Summary of Override Prediction Model Results

Model	VALID AUPRC	TEST AUPRC	ROC-AUC (Test)	F1 (Override, Test)	Precision (Override)	Recall (Override)	Key Notes
Logistic Regression	0.349	0.232	0.994	0.308	0.326	0.292	Linear baseline; limited ability to capture non-linear behaviour
XGBoost (Optimised)	0.866	0.803	0.999	0.731	0.781	0.688	Best overall performance; top features = reward rate, events per day, daily energy, fatigue signals
LSTM	0.006	0.013	0.859	0.024	0.013	0.244	Sequence model struggled with severe class imbalance; underperformed baselines

Note. The table reports model performance across validation and test sets. AUPRC, ROC-AUC, F1 score, precision, and recall for the override class. XGBoost achieved the strongest results, while logistic regression provided only limited predictive value and the LSTM underperformed due to severe class imbalance.

Table 5. Summary of Energy Saving Prediction Model Results Under Strict Pre-notice Settings

Model	Test RMSE	Test MAE	Top Features (via SHAP / Coefficients)	Key Notes
LightGBM	0.0000	0.0000	reward, power, reward_rate, period_duration, daily_energy_Wh	Best accuracy; incentive and load-related variables dominate
Linear Regression	0.0001	0.0000	power (+), car charging socket (+), daily_energy_Wh (+)	Interpretable baseline; weaker accuracy
Ridge Regression	0.0001	0.0000	Similar to Linear Regression	Adds stability; same key drivers
TCN (Optimised)	0.0021	0.00049	appliance_Light/Bulb, week_in_trial, reward_rate, month_sin, precipitation	Slightly higher error; captures sequential and fatigue dynamics

Note. RMSE and MAE on the test set. Feature importance is reported using SHAP values (LightGBM, TCN) or model coefficients (Linear, Ridge Regression). In linear and ridge regression, the coefficient sign indicates direction: “+” denotes a positive association with energy savings, and “−” denotes a negative association. LightGBM achieved the best accuracy, while the TCN contributed additional insight into sequential and fatigue dynamics.

Table 6. Top 15 Feature Drivers of Override Probability (XGBoost, SHAP Importance) With Beeswarm Plot

Feature	mean_SHAP_logit	mean_abs_SHAP_logit	mean_delta_prob_pp	median_delta_prob_pp	p5..p95_delta_prob_pp
reward_rate	-5.1661	5.2741	-3.5586	-3.2438	-10.64..-0.81
notice_time	-1.7802	1.8185	0.0733	-0.0817	-0.15..-0.04
switch_off_event	-0.4380	0.4462	0.0090	-0.0090	-0.01..-0.01
daily_energy_Wh	-0.4208	0.4332	-0.2978	-0.0001	-0.05..0.00
number_events_date	-0.2846	0.3282	-0.0648	-0.0050	-0.02..0.00
energy_Wh	-0.0862	0.3083	-0.1305	-0.0037	-0.01..0.01
override_rate_in_last_7_days	-0.0989	0.2280	-0.0547	-0.0031	-0.00..0.01
daily_reward	0.0488	0.2132	-0.0654	0.0019	-0.01..0.00
power	0.0074	0.1967	-0.0556	-0.0017	-0.00..0.01
city_delhi	-0.0708	0.0708	-0.0255	-0.0010	-0.00..-0.00
weekly_reward	-0.0448	0.0680	-0.0118	-0.0009	-0.00..0.00
obs	-0.0043	0.0592	0.0078	-0.0000	-0.00..0.00
events_in_last_7_days	0.0085	0.0534	-0.0061	-0.0003	-0.00..0.00
day_of_year_cos	-0.0424	0.0478	-0.0124	-0.0002	-0.00..0.00
appliance_AC	-0.0135	0.0414	-0.0098	0.0002	-0.00..0.00

Table 6, Subtable 1

Note. *mean_SHAP_logit* indicates whether a feature typically increases (+) or decreases (–) override probability. *mean_abs_SHAP_logit* measures the overall strength of influence regardless of direction. *mean_delta_prob_pp* represents the average change in predicted probability (in percentage points). *median_delta_prob_pp* shows the typical change, less sensitive to outliers. *p5..p95_delta_prob_pp* gives the range from the 5th to 95th percentile, showing consistency or variability of effects

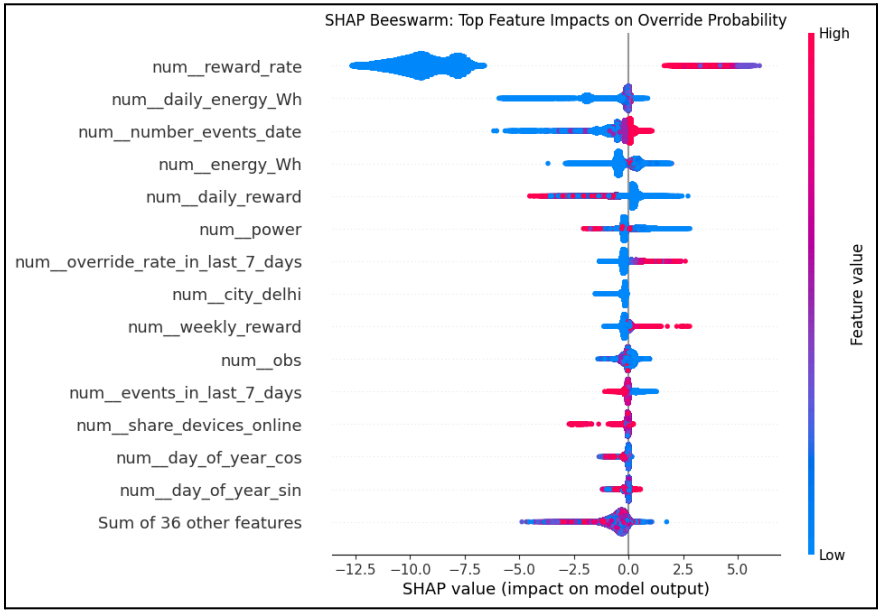


Table 6, Subplot 1

Note. Positive SHAP values indicate higher override probability, while negative values indicate lower probability.

Table 7. Top 15 Feature Drivers of Energy Savings With SHAP Importance (LightGBM, TCN) and Beeswarm Plots

Rank	Feature	mean_SHAP_logit	mean_abs_SHAP_logit	mean_delta_prob_pp	median_delta_prob_pp
1	reward	-5.01e-06	2.84e-05	-1.25e-04	-4.34e-04
2	power	-5.52e-06	1.80e-05	-1.38e-04	1.60e-04
3	reward_rate	-4.97e-07	9.12e-06	-1.24e-05	5.60e-05
4	daily_reward	1.37e-07	2.77e-06	3.44e-06	9.00e-06
5	daily_energy_Wh	4.52e-07	2.42e-06	1.13e-05	-2.20e-05
6	notice_time	-1.86e-07	2.23e-06	-4.66e-06	2.20e-05
7	energy_Wh	-6.42e-08	1.18e-06	-1.60e-06	-4.00e-06
8	weekly_reward	1.54e-07	6.20e-07	3.85e-06	9.00e-06
9	total_power_kW	-1.42e-08	3.44e-07	-3.55e-07	-3.00e-06
10	share_devices_online	1.29e-07	3.35e-07	3.22e-06	3.00e-06
11	appliance_Electric Geyser	-5.21e-09	3.09e-07	-1.30e-07	0.00e+00
12	Surface_Pressure	-8.26e-08	2.99e-07	-2.07e-06	-2.00e-06
13	surface_net_solar_radiation	-2.09e-08	2.70e-07	-5.23e-07	-3.00e-06
14	max_future_notice	-1.10e-07	2.61e-07	-2.75e-06	-3.00e-06
15	mean_future_reward	5.86e-08	2.38e-07	1.46e-06	0.00e+00

Table 7, Subtable 1. LightGBM for Top 15 Features

Note. Variables are ranked by SHAP importance. *mean_SHAP_logit* shows whether the feature typically increases (+) or decreases (–) predicted savings. *mean_abs_SHAP_logit* measures overall influence regardless of sign. *mean_delta_prob_pp* indicates the average effect on predicted probability (in percentage points). *median_delta_prob_pp*shows the typical effect, less sensitive to outliers. Definitions are consistent with those in Table 6, Subtable 1.

Rank	Feature	mean_SHAP_logit	mean_abs_SHAP_logit	mean_delta_prob_pp	median_delta_prob_pp
1	precipitation	5.34e-07	6.46e-07	1.34e-05	-1.04e-10
2	appliance_Light / Bulb	-4.70e-07	5.71e-07	-1.17e-05	0.00e+00
3	month_sin	4.90e-07	5.53e-07	1.23e-05	1.05e-06
4	day_of_year_sin	4.67e-07	5.21e-07	1.17e-05	1.15e-06
5	surface_net_solar_radiation	2.27e-07	2.75e-07	5.68e-06	0.00e+00
6	temperature	-2.21e-07	2.29e-07	-5.53e-06	-1.01e-06
7	week_in_trial	2.28e-08	2.11e-07	5.69e-07	0.00e+00
8	notice_time	6.81e-08	1.95e-07	1.70e-06	0.00e+00
9	appliance_Electric Geyser	1.90e-07	1.90e-07	4.75e-06	0.00e+00
10	obs	-1.48e-08	1.74e-07	-3.69e-07	-4.11e-13
11	day_of_week_cos	4.45e-08	1.60e-07	1.11e-06	0.00e+00
12	month_cos	-1.12e-07	1.51e-07	-2.81e-06	-2.04e-07
13	appliance_AC	1.08e-07	1.46e-07	2.70e-06	0.00e+00

14	device_count	1.19e-07	1.42e-07	2.96e-06	4.26e-08
15	reward_rate	1.39e-08	1.38e-07	3.48e-07	0.00e+00

Table 7, Subtable 2. TCN for Top 15 Features

Note. The same interpretation applies as in Subtable 1. TCN highlights seasonal encodings and weather features as more influential compared to LightGBM.

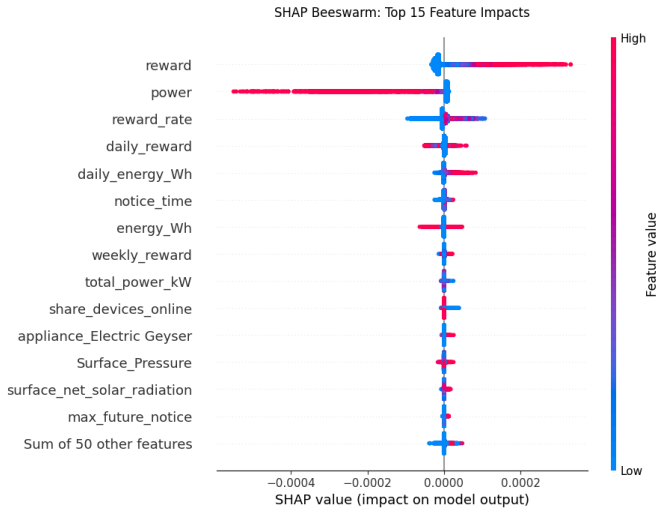
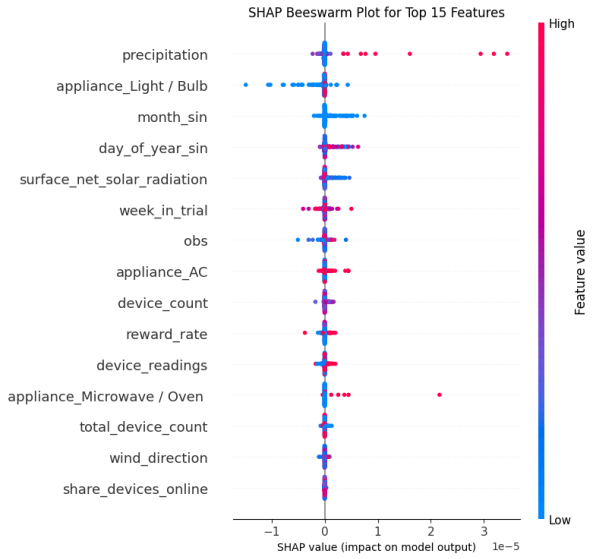
	LightGBM	**TCN**
Beeswarm Plot		

Table 7, Subtable 3. SHAP Beeswarm Plots for Top 15 Features: LightGBM vs TCN

Note. Beeswarm plots show the distribution of SHAP values across all samples. Positive values indicate higher predicted savings, while negative values indicate lower savings.

References

Ahmad, T., Chen, H., Huang, J., & Zhang, Y. (2020). A review on machine learning techniques for electricity load forecasting. *International Journal of Computer and Information Engineering, 14*(7), 235–241.

Birol, F., et al. (2020). *World Energy Outlook 2020*. International Energy Agency (IEA).

Copernicus Climate Change Service (C3S). (2017). *ERA5 hourly data on single levels from 1940 to present* [Dataset]. Copernicus Climate Data Store (CDS). European Centre for Medium-Range Weather Forecasts (ECMWF). https://cds.climate.copernicus.eu/datasets/reanalysis-era5-single-levels

Gellings, C. W. (2021). *The smart grid: Enabling energy efficiency and demand response*. CRC Press.

Géron, A. (2019). *Hands-On machine learning with Scikit-Learn, Keras, and TensorFlow* (2nd ed.). O'Reilly Media.

Jammalamadaka, S. R., & SenGupta, A. (2001). *Topics in circular statistics*. World Scientific.

Khanna, S., Martin, R., & Muûls, M. (2025, January 9). *Building virtual power plants: Incentives and automation for demand-side flexibility*. SSRN. https://doi.org/10.2139/ssrn.5089649

Lam, J. C., Wan, K. K. W., & Yang, L. (2008). Solar radiation influences on building cooling and lighting energy consumption. *Energy, 33*(10), 1570–1583. https://doi.org/10.1016/j.energy.2008.06.005

Sailor, D. J. (2001). Relating residential and commercial sector electricity loads to climate—Evaluating state level sensitivities and vulnerabilities. *Energy, 26*(7), 645–657. https://doi.org/10.1016/S0360-5442(01)00023-8

Santamouris, M., Cartalis, C., Synnefa, A., & Kolokotsa, D. (2015). On the impact of urban heat island and global warming on the power demand and electricity consumption of buildings—A review. *Energy and Buildings, 98*, 119–124. https://doi.org/10.1016/j.enbuild.2014.09.052

Siano, P. (2014). Demand response and smart grids—A survey. *Renewable and Sustainable Energy Reviews, 30*, 461–478. https://doi.org/10.1016/j.rser.2013.10.022