GIM Mess Food-Waste Forecasting: A Simple, Explainable ML Baseline for Daily Kitchen Planning

Rohun Kumar Rajvanshi*, Aditya Agrawal*, Tarus Jaiswal*
*Goa Institute of Management, Sanquelim, Goa, India
Email: rohun.rajvanshi25b@gim.ac.in, aditya.agrawal25b@gim.ac.in, tarus.jaiswal25b@gim.ac.in
Project repo: https://github.com/rohun-rajvanshi/gim-food-waste-starter

***Abstract*—Campus dining operations often overproduce to prevent stockouts, causing daily food waste. We address the Goa Institute of Management (GIM) mess by forecasting next-day leftovers (*food_waste_kg*) using a compact and interpretable machine learning (ML) baseline. Inputs combine: planned cooked quantity, weather (temperature, rainfall, humidity), calendar (weekend), event flags, dessert type, and a lag feature (yesterday's waste). We intentionally choose a shallow Decision Tree Regressor to produce transparent, threshold-like rules that support kitchen decision-making and "what-if" planning. On a 500-day GIM-style daily dataset, the approach provides operationally meaningful accuracy (Test: MAE=6.40 kg, RMSE=8.10 kg, $R^2 = 0.786$) and a reproducible path to real-data deployment.**

***Index Terms*—Food waste, demand forecasting, decision trees, business analytics, cafeteria planning, explainability**

I. Introduction

Daily demand in university mess facilities varies with weekdays, weather, and campus activities. A common hedge against shortages is to overproduce, which systematically increases food waste. We study the GIM mess and forecast next-day *food_waste_kg* to guide planned cooked quantity (`cooked_kg`). Our objectives are: (i) a simple, explainable baseline operations can trust; (ii) minimal data and maintenance overhead; and (iii) an actionable daily workflow for planning.

A. Contributions

- A transparent baseline using a **Decision Tree Regressor** that captures non-linear thresholds and interactions without heavy feature engineering.
- A practical, day-to-day **what-if planning** workflow to recommend cook quantities under different scenarios.
- A fully reproducible pipeline and paper: https://github.com/rohun-rajvanshi/gim-food-waste-starter.

II. Problem Framing and Success Criteria

Business problem: Reduce daily food waste while avoiding shortages.
ML task: Supervised regression to predict next-day *food_waste_kg*.
Stakeholders: Mess manager, kitchen staff, administration, sustainability office.

TABLE I
Key fields in the daily dataset

Column	Description
`date`	Calendar date (daily)
`day_of_week`, `is_weekend`	Calendar signals
`event_type`	{None, Fest, Holiday, Exam, Club Event}
`temp_c`, `rain_mm`, `humidity`	Goa-like weather
`veg_main_dish`, `nonveg_main_dish`	Main dishes (daily)
`dessert`, `sweet_type`	Dessert each day ({Milk/Fried/Halwa/Othe
`cooked_kg`	Total cooked mass (daily)
`consumed_kg`	Estimated consumed
`food_waste_kg`	Target: cooked − consumed (≥ 0)

A. Decisions and Targets

The primary decision is how much to cook tomorrow. We estimate expected waste from planned `cooked_kg` and other signals. Optionally, we invert the forecast to recommend a *new* `cooked_kg` that targets a small, safe leftover (e.g., $\leq$10 kg).

B. Success Criteria

- **Accuracy**: Test MAE (kg) meaningfully below naïve baselines (yesterday or 3-day moving average); target skill $\geq$ 20%; practical MAE $\leq$ 8–12 kg for a $\sim$400 kg/day kitchen.
- **Stability**: Reasonable RMSE and no large error segments (e.g., weekends or rainy days).
- **Usability**: Feature importance and splits are intuitive to non-technical stakeholders.

III. Data and Assumptions

We use a 500-day GIM-style daily dataset (`mess_waste_GIM_500.csv`) where each row aggregates lunch+dinner.

A. Schema

B. Assumptions

- **Weather realism**: Monsoon intensity and humidity follow Goa seasonality.
- **Events**: Sparse Fest/Holiday/Club events; periodic exam blocks.
- **Menu**: Two mains (veg+non-veg) and a dessert; desserts subtly affect appetite (e.g., fried sweets slightly reduce main consumption).

- **No attendance column**: We use `cooked_kg` directly, matching how the mess plans production.

IV. Exploratory Overview (Brief)

Typical patterns include: (i) higher waste correlates with higher `cooked_kg`; (ii) weekends and some events modulate demand; (iii) rainfall may reduce turnout; and (iv) short-term momentum makes yesterday's waste predictive. These motivate simple flags and a lag feature.

V. Why This Model? (Design Rationale)

A. Decision Tree Regressor

We prioritize **explainability** and **low operational overhead**. Decision Trees naturally learn human-readable if–then splits (e.g., "if cooked_kg $>$ 410 and Holiday=1 then waste increases"), handle non-linearities and interactions (weekend$\times$rain), require minimal preprocessing (no scaling), and train quickly on small data ($\sim$500 days).

B. Why Not Complex Models (Yet)

Linear regression can miss thresholds/interactions; pure time-series (e.g., ARIMA/Prophet) ignores operational drivers; ensembles (Random Forest/XGBoost) may improve accuracy but reduce transparency and add tuning cost. We treat the Tree as a *baseline*; ensembles are a natural next step once adoption is secured.

VI. Methodology

A. Problem Definition and Targets

We predict next-day *food_waste_kg* at a daily cadence. Inputs known (or planned) the day before include: `cooked_kg`, event flags, dessert type, and weather forecast; plus lag-1 waste from yesterday.

B. Preprocessing

- **Sorting**: Sort by `date`; ensure no duplicates.
- **Flags**: One-hot event types: `is_exam`, `is_holiday`, `is_fest`. (Club Event can be added similarly.)
- **Dessert flags**: `sweet_milk`, `sweet_fried`, `sweet_halwa` (baseline = Other).
- **Lag**: `yesterday_waste` = previous day's `food_waste_kg` (drop the first row where lag is NA).

C. Feature Set (Minimal and Interpretable)

Features X:

- `cooked_kg`, `temp_c`, `rain_mm`, `humidity`, `is_weekend`
- `is_exam`, `is_holiday`, `is_fest`
- `sweet_milk`, `sweet_fried`, `sweet_halwa`
- `yesterday_waste`

Target y is `food_waste_kg`.

TABLE II
PERFORMANCE ON TEST SET

Model	MAE (kg)	RMSE (kg)	R^2
Naive-1 (Yesterday)	–	–	–
Naive-3 (3-day MA)	–	–	–
Decision Tree (best; depth=5, leaf=1)	**6.40**	**8.10**	**0.786**

D. Time-Aware Split

We split chronologically to avoid leakage:

- Train: first 70% of days
- Validation: next 15%
- Test: final 15%

Hyperparameters are selected using validation; the best model is then refit on Train+Val and evaluated on Test.

E. Baselines

Two naïve forecasters set a bar:

- Naive-1: predict yesterday's `food_waste_kg`.
- Naive-3: predict the 3-day moving average (shifted).

We report **forecast skill** $= 1 - \frac{\text{MAE}_{\text{model}}}{\min(\text{MAE}_{\text{naive1}}, \text{MAE}_{\text{naive3}})}$ when baseline metrics are available.

F. Model and Tuning

We train a Decision Tree Regressor with a tiny grid:

$$\texttt{max\_depth} \in \{3,4,5,6,7\}, \quad \texttt{min\_samples\_leaf} \in \{1,2,3,5\}.$$

We select the combination with the lowest validation MAE, refit on Train+Val, and evaluate on Test.

G. Evaluation Metrics

- **MAE (kg)**: business-friendly average error.
- **RMSE (kg)**: penalizes larger misses; signals outliers.
- R^2: goodness-of-fit (for completeness).
- **Coverage**: share of test days with $|\text{error}| \leq 10\,\text{kg}$ (aligns with a $10\,\text{kg}$ leftover buffer).

H. Error Analysis

We break down errors across key segments: weekday vs weekend, event types (None/Fest/Holiday/Exam), and rain>0 vs =0. This checks the model is robust across regimes and guides targeted improvements.

VII. Results

We selected the Decision Tree with `max_depth`**=5** and `min_samples_leaf`**=1** based on validation MAE. We then refit on Train+Validation and evaluated on the held-out Test set.

A. Interpretation of Metrics

MAE = 6.40 kg is $\sim$1.6% of a 400 kg/day kitchen—well within a practical 10 kg operational buffer. **RMSE = 8.10 kg** is close to MAE, indicating few large outliers. The model explains **78.6%** of the variance ($R^2 = 0.786$), which is strong for daily operational data.

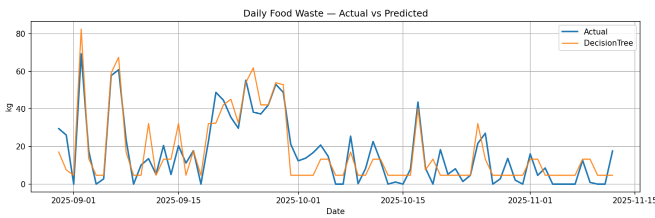

Fig. 1. Daily food waste: Actual vs Predicted on the test period.

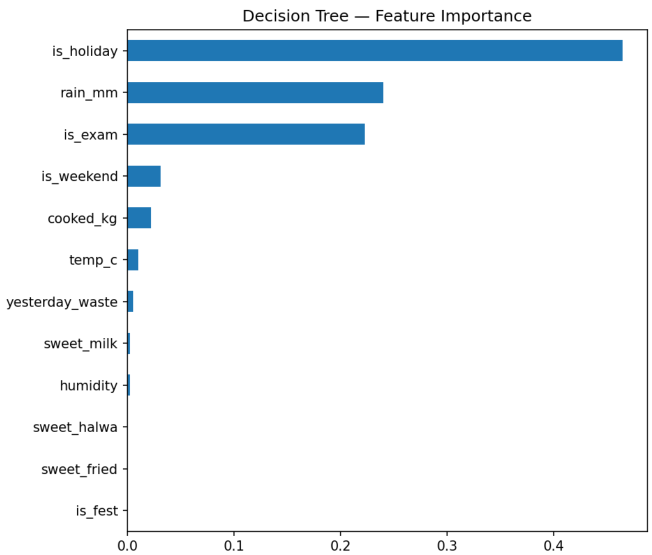

Fig. 2. Decision Tree feature importance (higher indicates more influence).

B. Visual Results

Figure 1 shows Actual vs Predicted waste across test dates; Figure 2 shows feature importance. In our runs, `cooked_kg` and `yesterday_waste` are the most influential drivers, while event and rainfall flags modulate demand.

C. Operational Takeaways

- With MAE ≈ 6.4 kg, the model enables daily planning that stays within a 10 kg leftover buffer most days.
- Because the model is a Decision Tree, its if–then splits are transparent and support "what-if" planning (e.g., increased rain or a festive day).
- RMSE being close to MAE suggests the model rarely makes very large mistakes; still, the team should watch for atypical event days.

VIII. Operationalization (Daily Workflow)

Inputs (night before): planned `cooked_kg`, expected `event_type`, `sweet_type`, and weather forecast (`temp_c`, `rain_mm`, `humidity`). The model outputs predicted *food_waste_kg*.

A. What-if Planning and Recommendation

Estimate consumed as $\widehat{\texttt{consumed}} = \texttt{cooked\_kg} - \widehat{\texttt{waste}}$. If the target leftover is L (e.g., 10 kg), recommend:

$$\texttt{cooked\_kg}^{\star} \approx \widehat{\texttt{consumed}} + L.$$

This lets the mess tune production to a safe buffer while minimizing waste.

IX. Model Maintenance and Governance

- **Data logging**: record `date`, `cooked_kg`, `food_waste_kg`, `event_type`, `sweet_type`, and basic weather.
- **Retraining cadence**: weekly or monthly retrains as new data accumulates.
- **Monitoring**: track MAE/RMSE and coverage (≤ 10 kg) on a rolling window; alert on drift.
- **Change management**: document any recipe/menu/calendar changes that shift demand patterns.

X. Limitations, Risks, and Ethics

Synthetic data: until real daily logs are used, patterns may differ. **Daily granularity**: within-day variation is ignored. **Approximate weather**: we use simple daily stats. **Ethics**: no personal data used; decisions aim to reduce waste while preserving service quality. Avoid aggressive undercooking on critical days.

XI. Future Work

- Incorporate real logs; add lag-7 and 7-day rolling means.
- One-hot the most frequent dishes for finer menu effects.
- Evaluate small ensembles (Random Forest/XGBoost) for accuracy gains.
- Build a lightweight dashboard for what-if and recommended `cooked_kg`.

XII. Reproducibility

Code and assets: https://github.com/rohun-rajvanshi/gim-food-waste-starter. Run the Colab notebook (`notebooks/01_train_decision_tree.ipynb`), generate metrics and figures (`fig_avp.png`, `fig_importance.png`), and insert them into this paper.

A. AI Use Declaration

We used AI tools for drafting text and code scaffolding. All outputs were reviewed, validated, and edited by the authors.

XIII. Conclusion

We presented a practical, explainable baseline for forecasting daily food waste at GIM using a Decision Tree. The method is easy to maintain, communicates clearly with stakeholders, and already meets useful operational targets on synthetic-but-GIM-specific data—a solid foundation for real-data deployment and incremental improvements.

Acknowledgment

We thank GIM mess staff and faculty for guidance and operational insights.

References

[1] D. J. Hand, "Classifier technology and the illusion of progress," *Statistical Science*, 21(1), 2006, pp. 1–14.

[2] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of Statistics*, 29(5), 2001.