Waste Management and Recycling in Indian Cities – Model Report

---

1. Introduction

India's rapidly urbanizing population poses a critical challenge in managing municipal solid waste. The mini-hackathon organized by PW Skills aimed to explore data-driven strategies for improving recycling efficiency. The primary goal of this project was to develop a machine learning model to predict the **Recycling Rate (%)** of a city, using select waste management attributes. By accurately forecasting recycling rates, municipalities can optimize resources, identify improvement areas, and move closer to sustainable waste practices.

---

2. Methodology

2.1 Data Understanding

The dataset spanned the years 2019–2023 and included multiple features such as:

- Waste Generated (Tons/Day)
- Population Density
- Municipal Efficiency Score
- Cost of Waste Management (₹/Ton)
- Awareness Campaigns Count
- Landfill Capacity (Tons)
- Year

The target variable was **Recycling Rate (%)**, a continuous value, making this a regression problem.

2.2 Data Preprocessing

- **Missing Data**: The dataset was clean with no missing values.
- **Outliers**: No significant outliers were found.
- **Categorical Encoding**: Categorical variables were converted into numeric format using pd.get_dummies().

Initially, all features were considered, resulting in 82 columns due to one-hot encoding of categorical data. However, due to UI constraints and to build a more user-friendly application, the model was retrained using only the following **7 features**:

- Waste Generated (Tons/Day)
- Population Density
- Municipal Efficiency Score

- Cost of Waste Management (₹/Ton)
- Awareness Campaigns Count
- Landfill Capacity (Tons)
- Year

2.3 Feature Engineering

- Focused only on numerical and directly impactful features.
- Removed location-based or categorical features not available in the app input.

2.4 Model Training

The following regression models were trained using only the 7 selected features:

1. **Ridge Regressor**
2. **Decision Tree Regressor**
3. **Random Forest Regressor**
4. **XGBoost Regressor**

Each model underwent hyperparameter tuning using GridSearchCV to enhance performance.

2.5 Model Evaluation

- **Metric**: Root Mean Squared Error (RMSE) was chosen as the evaluation metric to measure how well the model predictions aligned with the actual recycling rates.

Model	**RMSE (Lower is better)**
Ridge Regressor	16.513
Decision Tree Regressor	17.06
Random Forest Regressor	16.54
XGBoost Regressor	**16.511**

XGBoost Regressor gave the best performance with the lowest RMSE and was selected for final deployment.

---

3. Results & Visualizations

- XGBoost showed strong generalization on the validation set.
- The model's residuals were evenly distributed, indicating low bias.

- Visualizations included:
    - Feature importance plot (XGBoost)
    - Actual vs Predicted recycling rates

---

4. Challenges & Limitations

Challenges

- Initially trained the model using all 82 features got 82 features after converting categorical columns into numerical column, but the UI only accepts 7 features, leading to a complete retraining.
- Needed to re-engineer the pipeline to match input features allowed by the final deployment plan.

Limitations

- The final model is trained only on **7 numerical features**, possibly excluding rich categorical or geospatial insights.

---

5. Real-World Impact

This project offers a scalable solution for smart city administrators. With real-time input of only 7 measurable metrics, municipalities can:

- Predict recycling efficiency
- Improve operational planning
- Launch awareness campaigns in low-performing areas
- Optimize landfill management and resource allocation

Such predictions can reduce environmental impact and contribute to India's **Swachh Bharat Mission** and **Sustainable Development Goals (SDGs)**.

---

6. Deployment

Deployment is done on **Render**, a free and developer-friendly cloud platform for hosting web applications.

Why Render?

- Simple GitHub integration
- Fast Python app deployment
- Free tier supports small ML apps
- Great for Flask-based apps

Steps for Deployment:

1. Containerize the app or use a Flask API
2. Upload model (.pkl) and preprocessing logic
3. Link GitHub repo to Render
4. Deploy and test using sample inputs

Once deployed, the app will allow users to input the 7 required features and receive real-time predictions on Recycling Rate (%).

---

7. Conclusion and Future Work

This project demonstrated how a streamlined machine learning model, trained on real-world waste management features, can support urban planning and sustainability initiatives in India. XGBoost Regressor proved to be the most effective model based on RMSE.

Future Enhancements

- Incorporate additional features such as:
    - Waste type
    - Disposal method
    - Landfill location coordinates
- Build classification models to predict optimal **disposal methods**
- Deploy full-stack solution with analytics and insights on **Render or Hugging Face Spaces**