

# Report: Energy Consumption Prediction Using Machine Learning

## Objective

The aim of this project is to predict `equipment_energy_consumption` using regression techniques based on sensor and operational features from the dataset.

## Data Analysis

### Dataset Overview

- The dataset contains **16,857 rows** and **28 features**, along with a target variable `equipment_energy_consumption`.
- The data included both numerical and categorical variables, with some stored as strings (e.g., timestamps).
- Exploratory Data Analysis (EDA) included:
  - Distribution plots
  - Correlation heatmaps
  - Handling of string-formatted features

## Preprocessing

- **Dropped Columns:** Features like `timestamp`, `random_variable1`, and `random_variable2` were excluded from modelling as they were deemed non-informative or irrelevant.
- **Encoding & Scaling:**
  - Categorical data (if any) was encoded.
  - `StandardScaler` was used to normalize the features before model training.

## Model Training and Selection

### Train-Test Split

- The dataset was split into **80% training** and **20% testing** using `train_test_split`.

### Model Used

- A **RandomForestRegressor** was the primary model for prediction.
- **GridSearchCV** was applied for **hyperparameter tuning** with parameters like:
  - `n_estimators`: [50, 100, 200]
  - `max_depth`: [None, 10, 20, 30]
  - `min_samples_split`, `min_samples_leaf`, and `max_features`

## Evaluation Metrics

- **R<sup>2</sup> Score** was used to evaluate the model's performance.
- The best model after tuning yielded a strong R<sup>2</sup> score on the test set.
- **Mean Squared Error (MSE)** and **Mean Absolute Error (MAE)** were also computed during evaluation.

## Final Results

- **Model Selected:** RandomForestRegressor with hyperparameter tuning.
- **Best Parameters:** As found by GridSearchCV.
- **R<sup>2</sup> Score:** Achieved a high score indicating good predictive performance.

## Conclusion

- The Random Forest model proved effective in predicting energy consumption.
- Feature selection and hyperparameter tuning significantly improved model accuracy.
- Future improvements could include:
  - Testing other regressors (e.g., XGBoost, GradientBoosting)
  - Incorporating feature engineering
  - Time series modeling if timestamp info is useful.