Machine Learning – Final Project

Appliances Energy Prediction: Nonlinear Regression with Ensemble Methods

---

Overview

This is an **optional individual honours project** reserved to students who obtained at least **23/25** in the written examination. The project contributes up to **8 additional points** to your final grade, up to a maximum of **31** (30 cum laude).

The project asks you to develop a complete supervised machine learning pipeline on the **Appliances Energy Prediction dataset**, focusing on **nonlinear regression** and **ensemble methods**.

Dataset

Title: Appliances Energy Prediction

Source: UCI Machine Learning Repository

(archive.ics.uci.edu/dataset/374/appliances+energy+prediction)
Reference: Candanedo, L. M., Feldmann, A., & Degemmis, D. (2017). Data driven prediction models of energy use of appliances in a low-energy house. *Energy and Buildings*, 145, 13–25. https://doi.org/10.1016/j.enbuild.2017.03.040

The dataset comprises 10-minute interval measurements of electricity consumption in a low-energy house in Belgium, collected over approximately 4.5 months. Predictors include temperature and humidity from multiple rooms, weather data (outdoor temperature, wind speed, atmospheric pressure), and time-derived features (hour, day of week).

Target variable: `Appliances` (energy consumed by appliances, in Wh, over 10 minutes)

Predictors: Indoor and outdoor temperature, humidity, weather conditions, date/time features, and lagged energy values (if you choose to include them).

Data characteristics:

- $\approx 19{,}700$ observations
- 25 features (including derived and lagged variables)
- Continuous target variable
- **Strongly nonlinear** relationships due to occupancy patterns, temperature dependence, and time-of-day effects

---

Learning Objectives

By completing this project, you will:

1. Apply the full machine learning pipeline (EDA → preprocessing → modelling → evaluation → interpretation).
2. Understand and implement **nonlinear regression models** (polynomial, tree-based) and their advantages over linear regression.
3. Build and compare **ensemble regression methods** (Random Forest, Gradient Boosting) and understand their role in reducing variance and improving generalization.
4. Perform **feature scaling and outlier detection** with clear justification for your choices.
5. Critically evaluate model performance, interpret results, and reflect on limitations.
6. Write clear technical documentation and defend your work in an individual discussion.

Project Requirements

1. Exploratory Data Analysis (EDA)

- Load and describe the dataset (shape, data types, missing values, summary statistics).
- Visualize the **target variable** (histogram, boxplot) and identify its distribution properties.
- Explore relationships between key predictors (e.g., temperature, humidity, occupancy patterns if available) and the target using scatter plots and correlations.
- Discuss which features are likely to exhibit **nonlinear associations** with energy consumption (e.g., comfort-temperature effects, occupancy thresholds).
- Include at least 3–4 meaningful plots.

2. Data Preprocessing

Handling missing values:

- Identify and report missing data (if any). Use appropriate imputation or removal, with clear justification.

Feature scaling:

- Apply **standardization** (z-score normalization) or **min-max scaling** as needed.
- Clearly state which models require scaling and why (e.g., linear regression, ensemble tree-based models typically do/don't require it).

Outlier detection and treatment:

- Use at least one method: z-score, IQR (boxplot rule.
- Visualize outliers (boxplot or scatter plot).

- Decide whether to remove, cap, or keep them; justify your choice with reference to model robustness.

Feature engineering (optional but encouraged):

- Create any meaningful derived features (e.g., interaction terms, polynomial features for relevant variables).
- Explain why they may improve nonlinear regression performance.

3. Train-Test Split & Cross-Validation

- Use a **train/test split** (e.g., 70/30 or 80/20) or **k-fold cross-validation** (k = 5 or 10).
- Briefly explain your choice and why consistent validation is important for regression.

4. Model Development

Implement and compare the following **minimum** set of regressors:

1. **Baseline linear regressor**
    - Establishes a lower bound; helps you assess whether nonlinearity matters.
2. **At least two nonlinear regressors**, chosen from:
    - Polynomial regression (with appropriate degree and regularization)
    - Decision Tree regression
    - Support Vector Regression (SVR) with nonlinear kernel
3. **At least one ensemble regressor**, chosen from:
    - Random Forest regressor
    - Gradient Boosting (e.g., GradientBoostingRegressor, XGBoost, or LightGBM)

Hyperparameter tuning:

- For each model, perform **basic hyperparameter search** (grid search or random search with 3–5 configurations).
- Report the best hyperparameters found and justify your ranges (e.g., "tree depth 5–15 for Random Forest").

5. Model Evaluation & Comparison

- Compute standard regression metrics:
    - **Mean Absolute Error (MAE)**
    - **Mean Squared Error (MSE)** and **Root Mean Squared Error (RMSE)**
    - **Coefficient of Determination (R^2)**
    - Optionally, other metrics (MAPE, Huber loss) if applicable.

- **Create a comparison table** summarizing all models' performance on train and test sets.
- Discuss **overfitting vs. underfitting** (e.g., via learning curves or train-test error comparison).
- Analyse **which model performs best and why** (e.g., ensemble methods' variance reduction, nonlinear models' flexibility).
- Include at least one **residual plot** (predicted vs. actual, or residuals vs. predicted) to visually assess model fit and heteroscedasticity.

6. Error Analysis & Interpretation

- Inspect **failure cases**: Where does your best model make large errors?
- Visualize or describe **residuals** and comment on patterns (e.g., bias at high/low consumption, presence of outliers in residuals).
- For tree-based and ensemble models, compute and visualize **feature importance** or **permutation importance**.
- For linear models, report **coefficients** and their signs/magnitudes.
- Discuss which features are most predictive of energy consumption and whether this aligns with domain knowledge.

7. Reflection & Conclusions

- Summarise key findings: Which modelling approach worked best and why?
- Discuss **trade-offs** (accuracy vs. interpretability, simplicity vs. performance).
- Reflect on the **limitations** of your analysis:
    - Dataset size, time period, building specificity.
    - Features not available (e.g., explicit occupancy counts, weather forecasts).
    - Temporal dynamics (time-series structure) not fully exploited.
- Suggest **future improvements** (e.g., recurrent neural networks for time series, real-time occupancy data).

Deliverables

1. **Jupyter Notebook** (`final_project.ipynb`)
    - Clean, well-commented code with Markdown explanations between cells.
    - All plots and tables embedded.
    - Reproducible from start to finish.
2. **Short Report** (3–4 pages, PDF)

   - **Introduction**: Problem statement and dataset overview; why nonlinear regression is needed.
   - **Methods**: EDA summary, preprocessing choices, models selected, hyperparameter tuning strategy.
   - **Results**: Comparison table, best model performance, key plots (EDA, residuals, feature importance).
   - **Discussion**: Error analysis, interpretation, limitations, and future work.
   - **References**: Cite the dataset paper[1] and any other sources used.
3. **Individual Oral Explanation** (5–7 minutes, optional Q&A, PowerPoint presentation)
   - You will be asked to briefly explain:
     - Why you chose your specific nonlinear/ensemble models.
     - How scaling and outlier treatment affected results.
     - Why nonlinear regression outperforms (or doesn't outperform) linear regression on this problem.
   - This discussion helps verify originality and your understanding.

Grading Rubric (0–8 points)

Component	Points	Criteria
Problem formulation & data understanding	1	Clear description of the dataset, target variable, and nonlinear characteristics. Justification for why a nonlinear approach is appropriate.
EDA & visualizations	1.5	At least 3–4 informative plots. Distribution analysis of target and key predictors. Correlation or scatter plots showing nonlinear patterns.
Preprocessing (scaling, outliers, missing data)	1.5	Coherent handling of missing values, justified scaling decisions, systematic outlier detection and treatment with visualization.
Models & hyperparameter tuning	2	Correct implementation of baseline linear + ≥2 nonlinear + ≥1 ensemble models. Basic hyperparameter tuning with documented ranges. Clear model comparison table.
Evaluation & residual analysis	1	Appropriate metrics (MAE, RMSE, R^2). Train-test comparison. At least one residual or diagnostic plot. Discussion of overfitting.
Interpretation & feature importance	0.5	Feature importance or coefficient analysis. Error analysis: where does the model struggle? Alignment with domain intuition.

Code quality & reproducibility	0.5	Clean, commented notebook. No hardcoded paths; reproducible from scratch. Clear variable names and section structure.

Grading scale:

- 8/8: Excellent, thorough work with deep insights, clear writing, and strong technical execution.
- 6–7/8: Good work, all requirements met, minor gaps in depth or clarity.
- 4–5/8: Acceptable, requirements mostly met, some sections incomplete or unclear.
- 2–3/8: Below expectations, significant omissions or conceptual errors.
- 0–1/8: Incomplete or severely deficient work.

Academic Integrity & Generative AI Policy

This is an **individual project**. **Use of generative AI tools (ChatGPT, Claude, GitHub Copilot, etc.) to write code, text, or produce analyses is not permitted**.

What you must do:

1. Write all code and analysis yourself.
2. Any external code snippets, tutorials, or resources must be **cited** (e.g., "Adapted from https://...").
3. At the end of your report, include an **"AI Usage Statement"** declaring:
   - Did you use any generative AI tools? (Yes/No)
   - If yes, for what purpose (e.g., "Conceptual question about Random Forest interpretation") and how they were used.
   - Undeclared or misused AI assistance constitutes academic misconduct.

What is acceptable:

- Consulting documentation, textbooks, or research papers.
- Using Stack Overflow or GitHub for debugging existing errors (if cited).
- Using scikit-learn, pandas, etc., documentation to understand API usage.

Verification: You will be asked to explain your work orally. Failure to explain or defend your notebook and report may result in a grade of zero or negative.

Resources

- **Dataset**: https://archive.ics.uci.edu/dataset/374/appliances+energy+prediction

- **Primary reference**: Candanedo, L. M., Feldmann, A., & Degemmis, D. (2017). Data driven prediction models of energy use of appliances in a low-energy house. *Energy and Buildings*, 145, 13–25. https://doi.org/10.1016/j.enbuild.2017.03.040
- **Scikit-learn documentation**: https://scikit-learn.org
- **Pandas & Matplotlib**: https://pandas.pydata.org, https://matplotlib.org
- **Other references as needed** (e.g., lecture notes on nonlinear regression, ensemble methods).

Questions & Support

Please reach out if you:

- Are unclear on any requirement.
- Need advice on model selection or hyperparameter ranges.
- Have questions about the dataset or statistical methods.

Note: I cannot review your code line-by-line before submission, but I can discuss conceptual questions and clarify requirements.

Final Note

This project is designed to consolidate your understanding of the machine learning pipeline in the context of a real-world regression problem. The focus is on **depth and reasoning** rather than exhaustive experimentation. Show your thinking clearly, justify your choices, and reflect critically on your results. Good luck!

References

[1] Candanedo, L. M., Feldmann, A., & Degemmis, D. (2017). Data driven prediction models of energy use of appliances in a low-energy house. *Energy and Buildings*, 145, 13–25. https://doi.org/10.1016/j.enbuild.2017.03.040