CS313.P23 - Data Mining and Application

Huynh Trong Nghia - 22520003, Dinh Thien An - 22520010,
Nguyen An - 22520019, Nguyen Hoang Gia An - 22520021,
Pham Nguyen Anh - 22520069, Nguyen Gia Bao - 22520109

1 Overview

Energy efficiency in construction is a critical factor in reducing environmental impact and mitigating climate change. Buildings account for a significant portion of global energy consumption and carbon emissions, primarily due to heating, cooling, and operational energy demands. Optimizing building design in the early stages can help reduce these impacts, yet traditional energy modeling tools can be too complex and time-consuming.

This study leverages Machine Learning (ML) techniques, particularly the XGBoost model combined with Active Learning, to predict energy performance and environmental impact using basic design parameters. We use the Energy Efficiency Dataset (768 residential samples) created by Xifara & Tsanas to analyze the sensitivity of operational carbon emissions ($CO2eq$) to geometric design factors.

The goal is to predict heating load ($Y1$) and cooling load ($Y2$) to estimate $CO2eq$ emissions, while identifying key features, such as Relative Compactness and Overall Height, that most influence energy consumption. The results show that ML models, utilizing simple geometric data, offer an effective approach for preliminary environmental impact assessments, supporting sustainable design decisions at the early design phase.

2 Dataset

2.1 Overview

In this study, the team uses the `Energy Efficiency` dataset (add reference) created by Angeliki Xifara (add reference) and processed by Athanasios Tsanas. This dataset contains 728 samples and 8 features describing various building design metrics, such as the ratio of wall and window areas in different orientations, which play a crucial role in determining a building's energy consumption.

The features in the dataset are as follows:

- **X1**: The ratio between the building's area and its surface area
- **X2**: The surface area of the building
- **X3**: The wall area
- **X4**: The roof area
- **X5**: The height of the building
- **X6**: The building's orientation
- **X7**: The window area
- **X8**: The distribution of window area

Target variable for the problem:

- **X1**: The ratio between the building's area and its surface area
- **X2**: The surface area of the building

This dataset provides a solid foundation for studying the relationship between building design and energy efficiency. By using this data, the team aims to build a model that can predict the energy efficiency of residential buildings and suggest potential improvements for energy conservation.

2.2 Processing Pipeline

	Relative Compactness	Surface Area	Wall Area	Roof Area	Overall Height	Orientation	Glazing Area	Glazing Area Distribution	Heating Load	Cooling Load
count	48.000000	48.000000	48.000000	48.000000	48.000000	48.000000	48.0	48.0	48.000000	48.000000
mean	0.764167	671.708333	318.500000	176.604167	5.250000	3.500000	0.0	0.0	14.286458	19.706250
std	0.106827	88.960297	44.059438	45.614184	1.768519	1.129865	0.0	0.0	7.625241	8.134356
min	0.620000	514.500000	245.000000	110.250000	3.500000	2.000000	0.0	0.0	6.010000	10.900000
25%	0.682500	606.375000	294.000000	140.875000	3.500000	2.750000	0.0	0.0	7.037500	12.047500
50%	0.750000	673.750000	318.500000	183.750000	5.250000	3.500000	0.0	0.0	13.200000	18.980000
75%	0.830000	741.125000	343.000000	220.500000	7.000000	4.250000	0.0	0.0	19.747500	25.860000
max	0.980000	808.500000	416.500000	220.500000	7.000000	5.000000	0.0	0.0	29.900000	39.440000

Fig. 1: Descriptive statistics

- **Data visualization**

  `Outliers:` Outliers were checked using boxplots and that no significant outliers were detected

  `Variable Distribution:` The Heating Load and Cooling Load are postively skewed. 'Overall Height', 'Orientation' have values with equal distribution of values

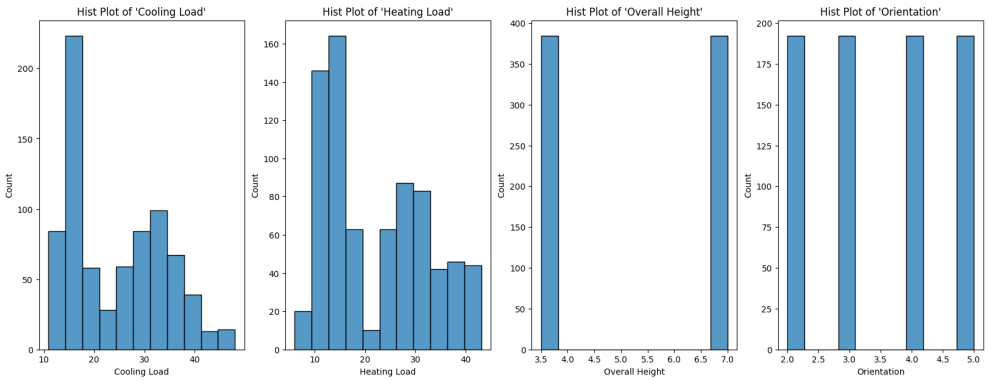


Fig. 2: Variable Distribution

`Relationships between Variables:`

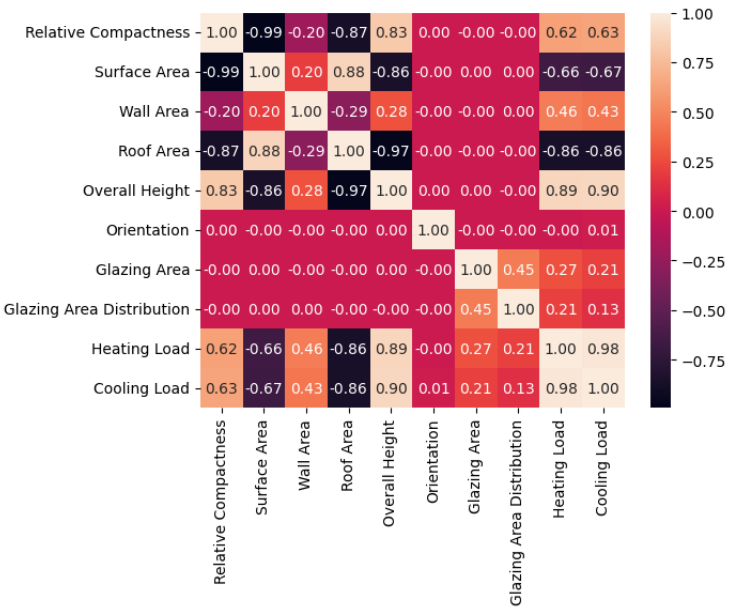


Fig. 3: Relative Compactness and Surface Area have a strong negative correlation of -0.99; Heating Load and Cooling Load have a strong positive correlation of 0.98

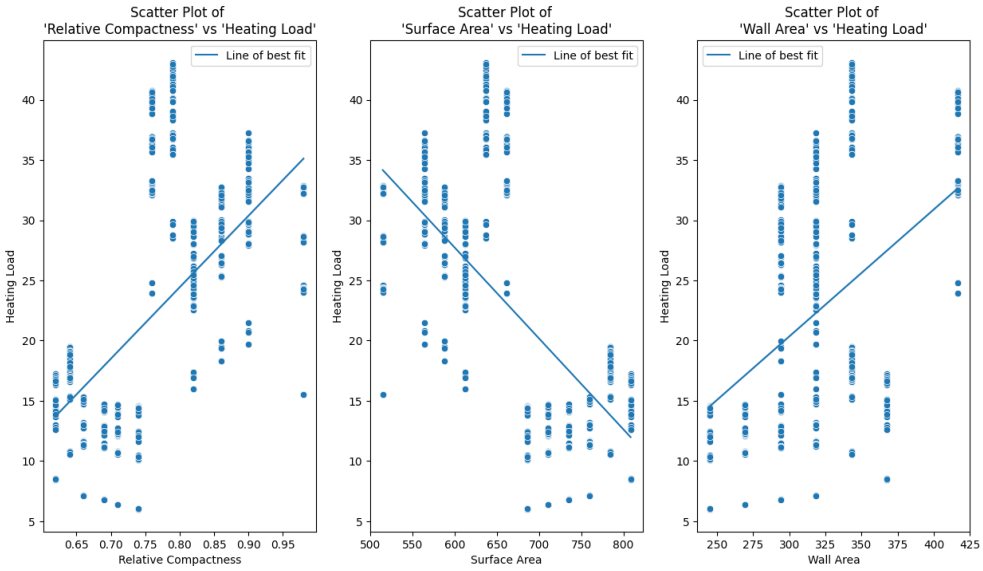


Fig. 4: Relative Compactness and Wall Area show a positive correlation with Heating Load, while Surface Area has a negative correlation. These findings emphasise how building design impacts energy consumption for heating.

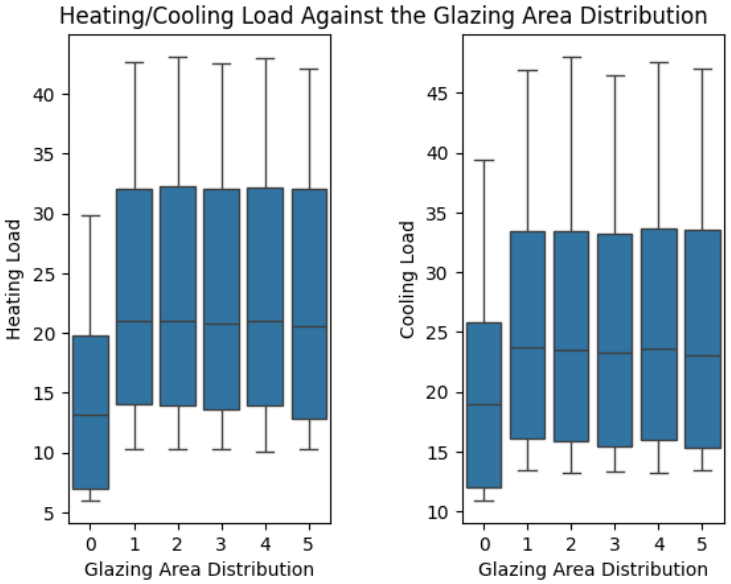
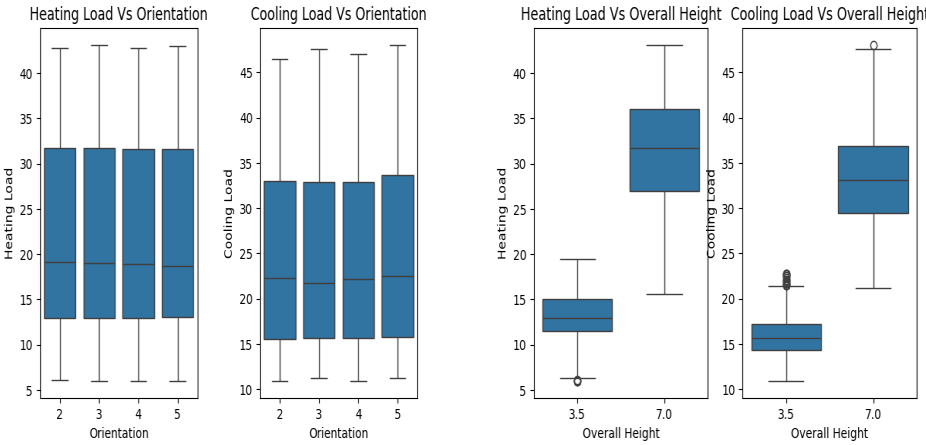


Fig. 5: The "Cooling Load" and "Heating Load" is affected either when the "Glazing Area Distribution" is 0 or greater than 0. The feature will be binned to have a value of 0 and 1 to reduce dimensionality



(a) (b)

Fig. 6: Orientation has no clear relationship with $Y1/Y2$, while Overall Height does

- **Feature Engineering**

  `Categorical Variable Handling:` "Glazing Area Distribution" feature was binarized. Values of 0 were kept as 0, and all other values (1-5) became 1, indicating presence (1) or absence (0) of glazing distribution.

  `Feature Selection/Removal:` "Surface Area" was dropped due to high correlation (-0.99) with "Relative Compactness"."Orientation" was dropped due to low correlation with target variable

  `Data Splitting:` The data was split into training (614 samples) and testing (154 samples) sets using an 80/20 ratio and a random_state of 42.

  `Data Scaling:` MinMaxScaler was applied to scale the remaining numerical features to a [0, 1] range, fitting only on the training data

Finally, the remaining features are: **X1**: The ratio between the building's area and its surface area, **X3**: The wall area, **X4**: The roof area, **X5**: The height of the building, **X7**: The window area, **X8**: The distribution of window area.

3 Modeling

3.1 Machine Learning Algorithms

This section outlines the regression algorithms considered for predicting heating/cooling loads.

3.1.1 Linear Regression

Models linear relationships between independent variables and a continuous target. Simple, interpretable, often used as a baseline.

3.1.2 Decision Trees

Partitions the feature space into regions to make predictions. Can capture non-linear relationships but prone to overfitting.

Key Parameters:

- `max_depth`: Maximum tree depth.
- `min_samples_split`: Minimum samples to split a node.
- `min_samples_leaf`: Minimum samples per leaf node.

3.1.3 Random Forest

An ensemble of Decision Trees trained on random data subsets and features to reduce variance and improve stability.

Key Parameters:

- `n_estimators`: Number of trees in the forest.
- `max_depth`: Maximum depth of each tree.
- `min_samples_split`: Minimum samples to split a node.
- `min_samples_leaf`: Minimum samples per leaf node.
- `bootstrap`: Whether bootstrap samples are used.

3.1.4 XGBoost

A high-performance gradient boosting algorithm that builds trees sequentially, focusing on correcting prior errors, with effective regularization.

Key Parameters:

- `n_estimators`: Number of boosting rounds (trees).
- `max_depth`: Maximum depth of each tree.
- `learning_rate`: Controls the contribution of new trees.
- `subsample`: Fraction of samples used per tree.
- `colsample_bytree`: Fraction of features used per tree.

3.1.5 Support Vector Machine (SVM)

Finds an optimal hyperplane for separation (classification) or fits data within an error margin (regression - SVR). Handles non-linearity via kernels. Parameter sensitive.

Key Parameters:

- `kernel`: Type of kernel (e.g., linear, rbf, poly).
- `C`: Regularization parameter.
- `gamma`: Kernel coefficient (for rbf, poly).
- `epsilon`: Margin of tolerance for SVR.

3.1.6 K-Nearest Neighbors (KNN)

An instance-based algorithm predicting based on the K nearest data points in the feature space. Requires no training but prediction can be computationally expensive.

Key Parameters:

- `n_neighbors`: Number of neighbors (K).
- `weights`: Weighting method for neighbors (uniform or distance).
- `p`: Power parameter for the Minkowski distance (1: Manhattan, 2: Euclidean).

4 Experiment

4.1 Experiment setup

1. **Hyperparameter Tuning with GridSearchCV**

Model	Parameter	Result
SVM	`kernel`	['linear', 'rbf', **'poly'**]
	`C`	[0.1, 1, **10**]
	`epsilon`	[0.01, 0.1, **0.5**]
	`gamma`	[**'scale'**, 'auto']
KNN	`n_neighbors`	[3, 5, 7, **9**]
	`weights`	['uniform', **'distance'**]
	`p`	[1, **2**]
	`algorithm`	[**'auto'**]
Decision Tree	`max_depth`	[**None**, 5, 10, 20]
	`min_samples_split`	[**2**, 5, 10]
	`min_samples_leaf`	[**1**, 2, 4]
	`max_features`	[**None**, 'sqrt', 'log2']
Random Forest	`n_estimators`	[100, **200**]
	`max_depth`	[**None**, 10, 20]
	`min_samples_split`	[**2**, 5]
	`min_samples_leaf`	[**1**, 2]
	`max_features`	[**None**, 'sqrt']
	`bootstrap`	[**True**]
XGBoost	`n_estimators`	[100, **200**]
	`max_depth`	[1, **2**]
	`learning_rate`	[**0.1**]
	`subsample`	[**0.8**, 1.0]
	`colsample_bytree`	[0.5, **0.8**]
	`min_child_weight`	[**3**, 5]

2. **Active Learning**

Active Learning is a strategy to boost model performance using limited labelled data by enabling the model to selectively query the most informative or uncertain samples for annotation by an oracle (typically human). This reduces labelling cost and time by focusing on the most valuable data points.

In this project, we simulated active learning to explore its benefits. Lacking a human annotator, we used ground-truth labels from the test set as a proxy for oracle responses after a sample was selected based on uncertainty. This allowed us to investigate performance gains as uncertain samples were iteratively labelled and added to the training set. We implemented two query strategies, using the test set as the query pool, to assess how effectively AL can achieve strong performance with fewer labelled examples, especially for ambiguous data.The strategy is :

Error-Based Sampling: Models identified samples with the highest prediction errors (squared residuals). These samples were added to the training set, and the models were retrained. This was applied to all baseline models

4.2 Performance

4.2.1 Evaluation Metrics

1. Mean Squared Error (MSE)

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

2. Mean Absoluete Error (MAE)

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - x_i|$$

3. Root Mean Squared Error

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y})^2}$$

4. R-squared

$$R^2 = 1 - \frac{\sum(y_i - \hat{y})^2}{\sum(y_i - \bar{y})^2}$$

4.2.2 Result

Model	MSE	MAE	RMSE	R^2
XGBoost	1.152116	2.952587	1.718309	0.968134
Decision Tree	1.165358	3.055920	1.748119	0.967019
Random Forest	1.169375	3.058888	1.748968	0.966987
K-Nearest Neighbor	1.214726	3.342413	1.828227	0.963927
SVM	1.637182	7.093737	2.663407	0.923441
Linear Regression	2.187545	9.650917	3.106592	0.895843

Table 1: Baseline Model Performance Comparison

Model	MSE	MAE	RMSE	R^2
XGBoost	1.148804	0.383093	1.709603	0.968456
Decision Tree	1.159368	3.014823	1.736325	0.967463
Random Forest	1.163210	3.018240	1.737308	0.967426
K-Nearest Neighbor	1.207157	3.386573	1.840264	0.963451
SVM	1.639293	7.146387	2.673273	0.922873
Linear Regression	2.192641	9.725087	3.118507	0.895042

Table 2: Active Learning Performance Comparison

5 Conclusion and future works

In this study, we compared the performance of multiple machine learning models for predicting energy consumption in residential buildings. XGBoost demonstrated the best performance, followed by decision trees and random forests. Active learning showed promise in reducing labeled data requirements and enhancing model accuracy. For future work, we aim to explore advanced feature engineering, experiment with other ensemble methods, and integrate more complex environmental data to improve prediction accuracy further. Additionally, incorporating real-time data for

model adaptation could offer valuable insights for energy optimization in real-world scenarios.