

ENHANCING ENERGY EFFICIENCY WITH EXPLAINABLE AI: A STUDY IN SMART BUILDING HEATING LOAD PREDICTION

Sandra, Sačarić¹, Ivan, Simić²

^{1 2} Faculty of Organization and Informatics, University of Zagreb, Pavlinska 2, Varaždin Croatia
ssacaric20@foi.hr, isimic21@foi.hr

Abstract

Artificial intelligence (AI) is a powerful tool for achieving sustainable development goals. However, the complexity of many AI models poses significant challenges in domains such as energy management. This paper explores the application of explainable artificial intelligence (xAI) techniques to enhance the transparency and reliability of machine learning models used for predicting energy consumption in smart buildings. Using a dataset containing structural and physical features of buildings, a predictive model was developed based on the Random Forest regression algorithm. To interpret and explain the model's predictions, two xAI methods - LIME and SHAP - were applied. The results demonstrate that both xAI techniques effectively identify the key features influencing heating load predictions, thereby providing actionable insights into energy-saving strategies. Furthermore, the combination of machine learning and xAI methods fosters trust and accountability in AI-driven energy management systems.

Keywords: Python, prediction model, Random Forest, energy management, xAI, LIME, SHAP

Izboljšanje energetske učinkovitosti z razložljivo umetno inteligenco: študija napovedi ogrevane obremenitve v pametnih stavbah

Izvleček

Umetna inteligenca (UI) je zmogljivo orodje za doseganje ciljev trajnostnega razvoja. Vendar pa kompleksnost številnih modelov UI predstavlja pomemben izziv na področjih, kot je upravljanje z energijo. Ta članek preučuje uporabo tehnik razložljive umetne inteligence (xAI) za izboljšanje preglednosti in zanesljivosti modelov strojnega učenja, uporabljenih za napoved porabe energije v pametnih stavbah. Z uporabo nabora podatkov, ki vsebuje strukturne in fizikalne značilnosti stavb, je bil razvit napovedni model na osnovi regresijskega algoritma Random Forest. Za interpretacijo in pojasnjevanje napovedi modela sta bili uporabljeni dve metodi xAI – LIME in SHAP. Rezultati kažejo, da obe xAI-tehniki učinkovito prepoznata ključne značilnosti, ki vplivajo na napovedi ogrevalne obremenitve, kar omogoča oblikovanje praktičnih strategij za varčevanje z energijo. Poleg tega kombinacija strojnega učenja in xAI-metod spodbuja zaupanje in odgovornost v sistemih upravljanja energije, podprtih z UI.

Keywords: Python, napovedni model, Random Forest, upravljanje z energijo, xAI, LIME, SHAP

1 INTRODUCTION

The focus of this work is the problem of excessive energy consumption in smart buildings, despite their modern design. Residents face high annual heating costs, indicating potential inefficiencies in energy management. This motivates the use of data mining methods to understand consumption patterns, optimize them, and build models that support more informed decision-making. Given the increasing need for energy efficiency and sustainable resource management, especially in urban areas, the work also aims to bridge the gap between engineering solutions and the general public.

The primary objective is to develop and evaluate predictive models of energy consumption using available data on building dimensions and characteristics. Through feature analysis, identification of key variables, and scenario testing, the paper provides recommendations for reducing energy use. A key contribution is the implementation of LIME and SHAP to explain machine learning models, enabling greater transparency, improved interpretation, and better decision-support. The work also contributes pedagogically by demonstrating the full modeling workflow; from preprocessing and feature selection to visualization, evaluation, and interpretation, in a widely accessible environment (Google Colab), offering a useful template for education in xAI and data science.

The paper is structured into several sections. The introduction presents the motivation, context, and objectives. The theoretical framework reviews relevant literature, data processing methods, key concepts, predictive modeling, and interpretability approaches, including the challenge of model nontransparency. A descriptive dataset analysis follows, covering missing values, outliers, and visualizations. The next section addresses model development, performance assessment, and the application of LIME and SHAP. The conclusion provides a performance analysis, comparison of xAI methods, and a reflection on the extent to which the project goals were met.

2 METHODOLOGY

This section will briefly go over the methods and techniques that were used, their importance, and what they represent. It will cover the general concept of machine learning, how it is categorized, what problems may arise after building a model, which algorithm will be used to construct the model, how it will be evaluated, and finally, it will go into more detail about the methods that will be used to explain the results obtained by the model.

2.1 Machine learning and regression models

Starting Machine learning (ML) combines statistics and computer science and is widely used because it automates learning from data. Unlike classical programming, ML does not require defining every possible case, making it especially effective for Big Data, where models can independently detect patterns and trends far faster than humans.

ML is generally divided into supervised and unsupervised learning. In supervised learning, models train on labeled data where the target variable is known [1], enabling regression or classification. In unsupervised learning, models analyze unlabeled data to discover structures such as clusters or anomalies, as shown by Muharemi et al. [2].

In this work, a regression model is used to predict heating consumption in buildings. The model learns from labeled consumption data to generate predictions for new inputs. As noted by Mosavi et al., ML approaches outperform classic statistical methods in energy efficiency applications and improve decision-making and long-term cost management [3].

2.2 The problem of model nontransparency

Rudin et al. identify model nontransparency as a major challenge in ML [4]. This “black box” problem raises concerns about safety, legality, ethics, and trust. Increasing demand for interpretability has led to the development of tools like LIME and SHAP, which enhance transparency, as confirmed by Ribeiro et al. [5].

Nontransparency arises when the decision process inside a model is unclear, especially in complex models with many interdependent parameters and nonlinearities. Although such models may be accurate, accuracy alone cannot ensure reliability if predictions rely on flawed reasoning, potentially causing real-world failures [4]. Transparent and fair systems are essential in sensitive domains such as healthcare, law, and education, where harmful outcomes raise questions of responsibility and make error correction difficult.

Ultimately, inexplicable decisions can reduce trust, violate regulations, or cause tangible harm. Panda et al., focusing on medical contexts, emphasize why black box models are problematic and highlight the value of xAI techniques in improving interpretability [6].

2.3 Model selection

While this study primarily focuses on the Random Forest model, an additional comparison with another black-box machine learning model is included to provide a more comprehensive evaluation of its predictive performance.

Since the available dataset is relatively small, deep learning approaches such as artificial neural networks were not considered appropriate due to their higher data requirements and increased risk of overfitting. Instead, Support Vector Regression (SVR) was selected as a suitable benchmark model, as it has been widely applied in energy consumption prediction tasks and demonstrates strong performance on smaller datasets. For example, Hussien et al. employed SVR alongside other machine learning techniques for long-term building energy consumption forecasting, highlighting its effectiveness in this domain [7]. The focus on black-box models is intentional, as the objective of this work is to evaluate and interpret complex predictive models using Explainable Artificial Intelligence (XAI) methods. The purpose of including SVR is therefore to provide a meaningful reference point for assessing the performance and interpretability of the Random Forest model.

Random Forest, as the main focus of the explainability analysis within this work, is an ensemble-based ML algorithm. According to the Scikit-learn documentation, it uses bootstrapping (bagging) to build multiple decision trees, increasing randomness and reducing overfitting compared to classical decision trees [8]. The regression variant, RandomForestRegressor, is used in this work.

The method trains many simple trees on resampled data and introduces additional randomness by considering only subsets of features at each node. Breiman states that this reduces variance and mitigates overfitting [9]. Research by Zhou and Mentch shows that Random Forest handles noisy data well and performs effectively on large datasets [10]. For this reason, it is suitable for data with multicollinearity or unstructured features. Mosavi et al. also demonstrate the algorithm’s applicability in energy systems, even when used for classification tasks [3].

Despite these advantages, Random Forest remains a black box model and is thus well suited for xAI approaches, as shown by Panda et al. and Ribeiro et al. [6][5].

2.4 Evaluation metrics

Regression model performance is measured using R^2 , MAE, and RMSE. R^2 represents the proportion of variance in the target explained by the input features and reflects predictive ability on new data [8]. Seeing as both Random Forest and SVR models solve the problem of regression, these metrics will be used to evaluate each of their performances.

MAE and RMSE both quantify prediction error. MAE measures the average absolute difference between actual and predicted values, while RMSE squares errors, making it more sensitive to outliers [11]. Lower MAE and RMSE values are desirable, while higher R^2 values indicate better performance.

Although some studies argue that RMSE is not always the optimal metric [11], this work uses all three. If major discrepancies appear between RMSE and MAE, further analysis will be conducted.

2.5 Explainable artificial intelligence: LIME and SHAP

As machine learning models become more complex, the need for transparency increases to better understand decision-making and predictions. As defined by Panda et al., explainable artificial intelligence (xAI) emerges as a response to the question of how reliable these model outputs truly are. xAI provides insight into black box models, increasing transparency, reliability, and ethical use of AI.

Although there are many xAI methods, this work will focus on LIME and SHAP, as their importance is frequently emphasized in modern research that confirms their reliability. Panda et al. demonstrate the value of these methods in increasing user trust and reducing the risk of incorrect decisions [6].

LIME (Local Interpretable Model-Agnostic Explanations) aims to make the predictions of highly complex models more interpretable. It operates by locally interpreting the model, as noted by Lundberg and Lee [12]. Instead of explaining the entire model, it explains specific decisions for specific inputs by training a simpler model on a synthetic dataset generated around the point of interest [5]. Ribeiro et al. show that LIME consistently achieves strong performance, outperforming other xAI methods such as *Parzen* or heuristic techniques. It excels at explaining predictions even though it works locally, and users in simulations show that it helps distinguish reliable from unreliable predictions [5]. This is particularly important for Random Forest models, which are inherently black box systems.

SHAP (SHapley Additive exPlanations) is the second method used in this work. It is a modern approach based on cooperative game theory [12][6]. The fundamental idea is to assign each input feature a numerical value representing its importance in producing a prediction [12]. SHAP unifies several existing interpretation techniques into a "unified framework" making it flexible and intuitive. It also offers desirable properties such as accuracy and consistency [12]. SHAP uses approximations such as sampling. Its values are additive; each feature's contribution is calculated individually and then summed [13]. Because of that, it is not necessary to train the model for every possible combination of inputs, making SHAP efficient even on large datasets [12].

3 MODELING

After defining all relevant concepts and technical aspects that will be addressed in this work, the next step is to describe the processes related to building the predictive model and applying the two xAI methods.

To begin, the dataset used will be described, followed by the initial preparation and processing of the data that will be used to train the model. After that, the implementation of the Random Forest algorithm will be explained, along with the setting of hyperparameters for better model optimization. Finally, once the model has been successfully constructed and produces acceptable prediction results, methods that provide greater insight into how these results were generated will be applied.

3.1 Data description

The dataset used in this work was downloaded from the website Kaggle [14]. Similar datasets are commonly used when working with models for predicting the energy efficiency of buildings [3]. Specifically, the dataset contains information related to the physical characteristics of smart energy-efficient buildings. More precisely, it consists of 9 features, including surface areas of spaces, walls, and roofs, the orientation of balconies, and the ratio and distribution of glazing. The total number of instances exceeds 700 rows.

Table 1: Description of variables in the dataset.

N	Variable Name	Data type
1	Relative Compactness	Continuous
2	Surface Area	Continuous
3	Wall Area	Continuous
4	Roof Area	Continuous
5	Overall Height	Continuous
6	Orientation	Categorical nominal
7	Glazing Area	Continuous
8	Glazing Area Distribution	Categorical ordinal
9	Heating Load	Continuous

Table 1 lists all the exact feature names and their data types. It is evident that almost all features are continuous numerical variables, with only two exceptions. The feature related to orientation required special adjustment, which will be discussed later, while the feature *Glazing Area Distribution* is ordinal. However, since it appears in numerical form, no further preprocessing was necessary.

The following section goes through various data quality checks, a necessary step before model development to ensure the dataset's credibility and representativeness. As highlighted by Muharemi et al., data preprocessing plays a crucial role in establishing a strong foundation for building a predictive model [2]. The analysis included checking data types, followed by searching for missing, duplicated, and outlier values. For the first part of the analysis it was confirmed that this dataset contains no missing or duplicated values. The second part of analyzing the dataset included searching for outliers, which were examined primarily through visual inspection. Boxplots and histograms were used, and neither indicated the presence of outliers. Overall, the dataset was already quite clean and of high quality from the start, making it suitable for later modeling.

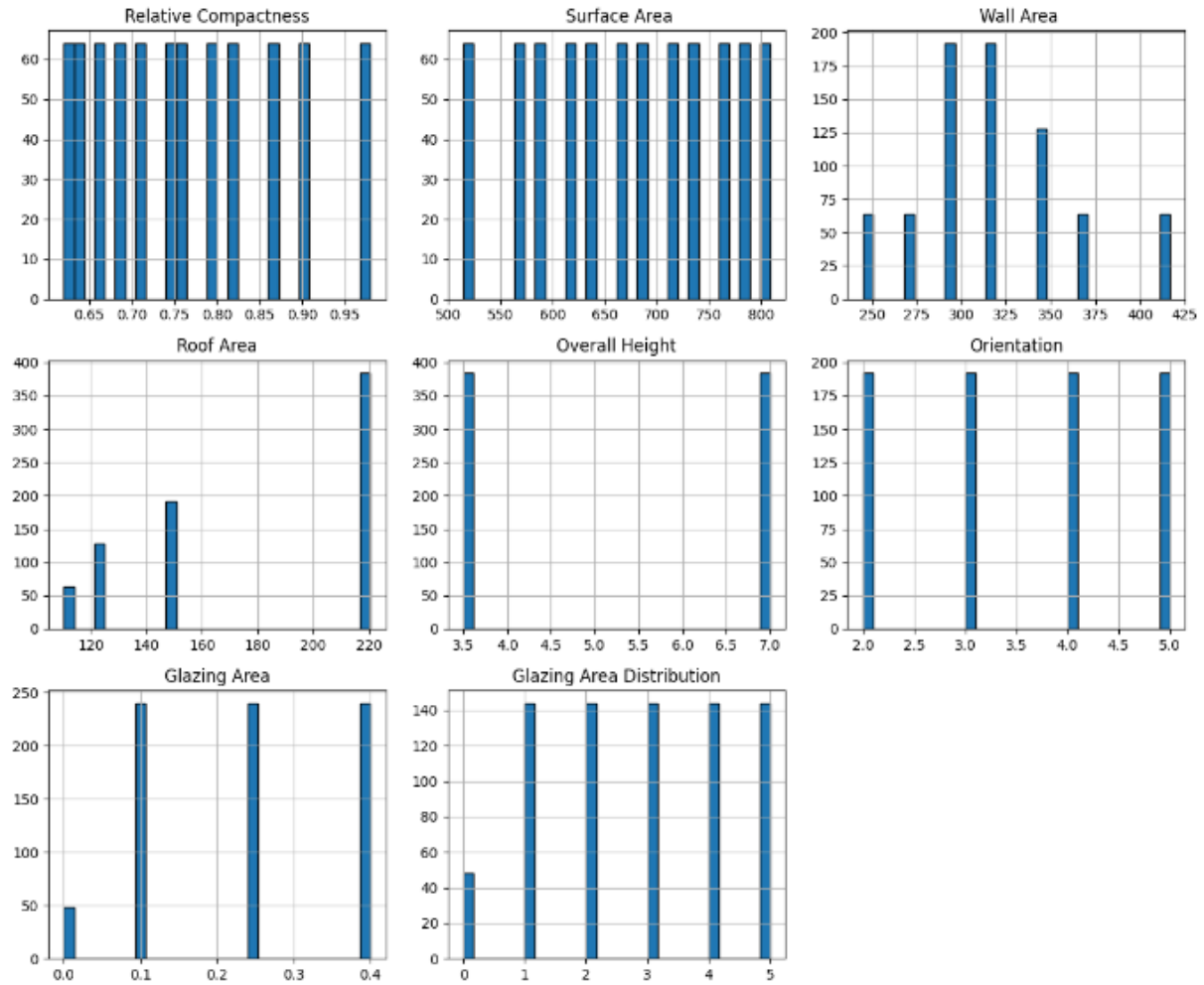


Figure 1: Histograms of variables.

Figure 1 displays histograms, bar style visualizations of the features, showing that the distributions and frequencies of values are mostly uniform. Some variation is visible only in the Wall Area and Roof Area features.

The dataset had also been checked for the presence of outliers – none were found. The boxplot diagrams showed no individual points plotted beyond the whiskers.

Other than that, it is important to point out the feature Orientation. Although represented by numerical values, it is actually a categorical variable because it describes the orientation of the building toward one of the four cardinal directions. Keeping these values in numeric form would be a major mistake, as the model could incorrectly interpret them as ordinal. Therefore, this feature was the only one that needed to be converted, specifically into binary dummy variables. Using this technique improves model interpretability, aligning with Rudin et al.'s recommendations regarding maintaining transparency throughout all stages of data processing [4]. For example, the value “south” is represented as 1 if present, 0 if absent, preventing unintended comparisons of value importance.

3.2 Data processing and preparation

Data preparation involved several steps to enable proper training and testing of the machine learning model. Although Sheth et al. focused on a classification problem, they clearly demonstrate that poor feature processing can lead to weak model performance even when advanced algorithms are used [15], highlighting the importance of correctly completing this

step. First, data normalization was necessary. Since different features have different value ranges, there is a risk that features with larger ranges could dominate the model. To address this, the data was transformed to make all features have a mean of 0 and a standard deviation of 1, effectively placing them on the same scale.

3.3 Model development

This phase focuses on building a predictive model using the Random Forest algorithm, which later demonstrated strong performance in solving the regression problem. Random Forest is commonly used in energy systems due to its robustness and interpretability, as shown by Mosavi et al. [3]. Zhou and Mentch also emphasize the importance of limiting tree depth to prevent overfitting [10].

Model development included analyzing and selecting the most important features, splitting the dataset into training and testing subsets, tuning hyperparameters for model optimization, and finally evaluating model performance. The following sections describe each of these steps in detail.

3.3.1 Splitting the data

Before making predictions, the model must be trained on the dataset. At the same time, its performance must be evaluated on a smaller subset. Therefore, the dataset was split into two parts: 80% for training and 20% for testing. This split provides a more objective evaluation of model performance and helps avoid overfitting.

3.3.2 The multicollinearity problem and feature selection

During the initial data analysis, the correlation matrix showed a considerable level of multicollinearity. As seen on Figure 3.

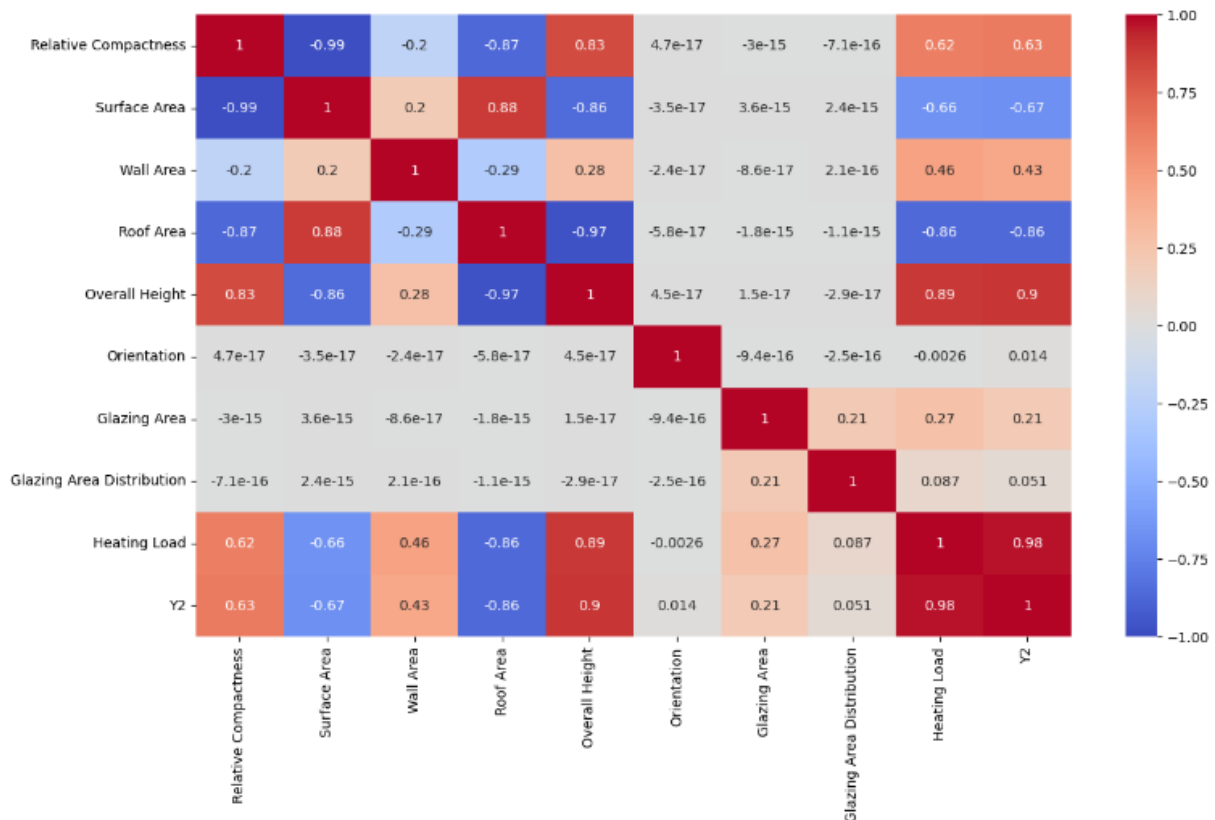


Figure 2: Heatmap correlation matrix.

The correlation matrix revealed several correlations above 0.90, considered very high. Therefore, including all features would be redundant, making feature importance analysis necessary. This revealed which features were essential and which should be removed. Through iterative testing and evaluation, two model versions were created: one trained on all features, and one trained on a reduced set after removing correlated variables. Figures 3 and 4 show these feature groups as well as evaluation metrics, which are discussed later.

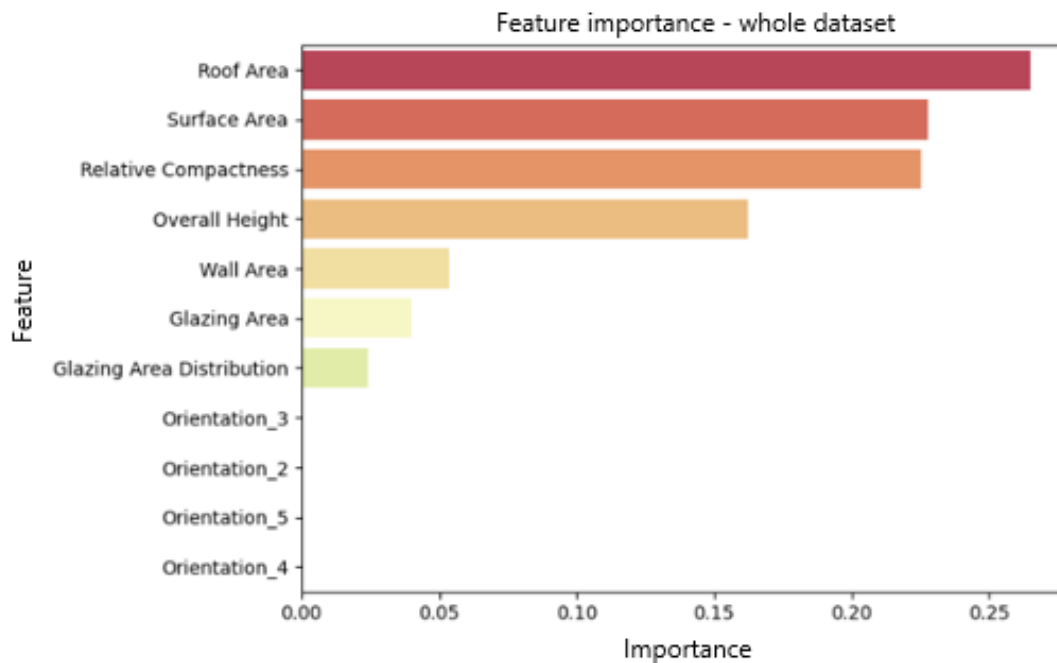


Figure 3: Whole dataset importance of features.

Figure 3 displays the importance of all original features. Figure 4 shows only the most important ones. Orientation was shown to have negligible impact and was therefore removed. Recursive Feature Elimination (RFE) further identified additional features to discard.

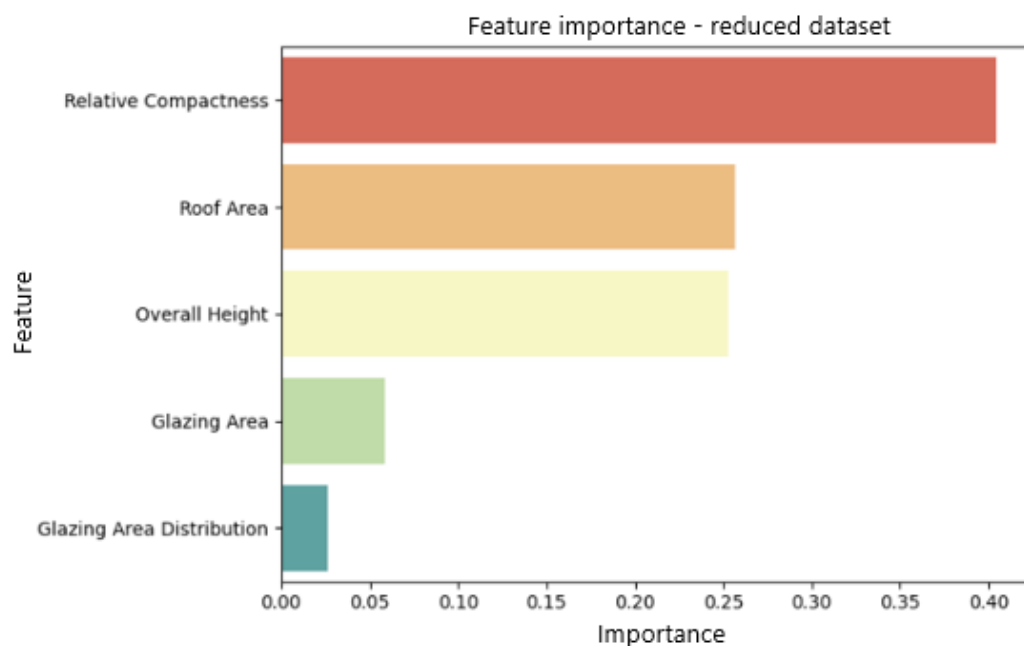


Figure 4: Reduced dataset importance of features.

3.3.3 Random Forest model parameters

The model was built using the Random Forest algorithm, which is based on training multiple decision trees. This work used the scikit-learn implementation with the following key hyperparameters:

- `n_estimators = 100`: the number of trees, sufficiently large to ensure model stability
- `max_depth = 5`: limits tree depth and reduces overfitting risk
- `max_features = "sqrt"`: the square root of the total number of features at each split
- `random_state = 42`: ensures the reproducibility of results

Hyperparameter settings significantly influence model accuracy and interpretability, as demonstrated by Sheth et al. and Zhou & Mentch [15][10].

Additional optimization techniques were applied after the initial model evaluation, discussed later.

3.3.4 Model training and evaluation

After setting the hyperparameters, the model was trained on the training subset. Predictions on the test subset were compared with the actual values using evaluation metrics R^2 , MAE, and RMSE.

These same metrics were used by Muharemi et al. for evaluating anomaly detection accuracy [2], confirming their relevance here. For both the reduced and full feature sets, R^2 was very high (over 97%), indicating that the model explained a large portion of variance in the target variable. Although generally desirable, such a high score may signal overfitting. Feature elimination improved model performance. MAE and RMSE values were low in both cases, indicating low prediction error.

3.3.5 Optimization

As previously noted, the reduced feature set produced better results. However, due to the extremely high R^2 values, further optimization was needed. Earlier in the project, several potential optimization methods were considered. Although genetic algorithms are also effective for tasks of this nature, as shown by Abbas et al. [16], GridSearch was selected as a simpler and more direct technique.

GridSearch was combined with cross-validation and K-Fold techniques, which divide the dataset multiple times during training and testing to identify optimal hyperparameter combinations, reduce bias, and provide more reliable results. After these optimizations, the R^2 value slightly decreased, which in this context indicated a more realistic and a better generalized model.

Table 2: Evaluation of the best model (Random Forest with the reduced dataset).

Method	Value
R^2 (best)	0.9762
MAE (best)	1.1817
RMSE (best)	1.5753

The table above shows the evaluation metric results. To optimize the model using GridSearch, the following key parameters were used in the process: the number of trees, maximum depth, minimum samples per split, minimum samples per leaf, and splitting criteria.

Several combinations of these parameters were tested, and the best performing ones were selected:

- `n_estimators = 200`
- `max_depth = 3, 5`
- `min_samples_split = 10, 20`
- `min_samples_leaf = 10, 20`
- `max_features = 0.3, 0.5`.

3.4 SVR Model Development

As a baseline for comparison with the Random Forest model, a Support Vector Regression (SVR) model was also developed. SVR is a well-established regression technique that attempts to find a hyperplane in a high-dimensional feature space that best fits the data within a defined margin of tolerance, known as the epsilon-insensitive tube. Unlike tree-based methods, SVR is sensitive to the scale of input features, which made preprocessing an essential step in the pipeline.

3.4.1 Pipeline and Preprocessing

To ensure fair and reproducible training, the SVR model was wrapped in a scikit-learn Pipeline that first applies `StandardScaler` normalization before fitting the regressor. `StandardScaler` transforms each feature to have zero mean and unit variance, which is critical for SVR since its optimization objective is distance-based and would otherwise be dominated by features with larger numerical ranges. The same reduced feature set identified during the Random Forest feature selection phase was used here, ensuring a consistent basis for comparison between the two models.

3.4.2 Hyperparameter Tuning

Given the sensitivity of SVR to its hyperparameters, `GridSearchCV` was applied in combination with 5-fold cross-validation using `KFold` with shuffling to identify the optimal configuration. The RBF (Radial Basis Function) kernel was selected, as it is generally well-suited for nonlinear regression problems and is the most widely used kernel in practice. The following hyperparameter grid was explored:

- `kernel = 'rbf'`: captures nonlinear relationships between features and the target variable.
- `C ∈ {1, 10, 100}`: the regularization parameter, controlling the trade-off between model complexity and training error tolerance.
- `epsilon ∈ {0.01, 0.1, 0.2}`: defines the width of the epsilon-insensitive tube; predictions within this margin incur no penalty.
- `gamma ∈ {'scale', 0.01, 0.1}`: controls the influence radius of individual training samples in the RBF kernel.

The grid search was scored using negative mean squared error and executed in parallel across all available CPU cores. This ensured that the best performing combination of hyperparameters was selected based on cross-validated performance.

3.4.3 Model Evaluation

The best estimator produced by `GridSearchCV` was used to generate predictions on the held-out test set.

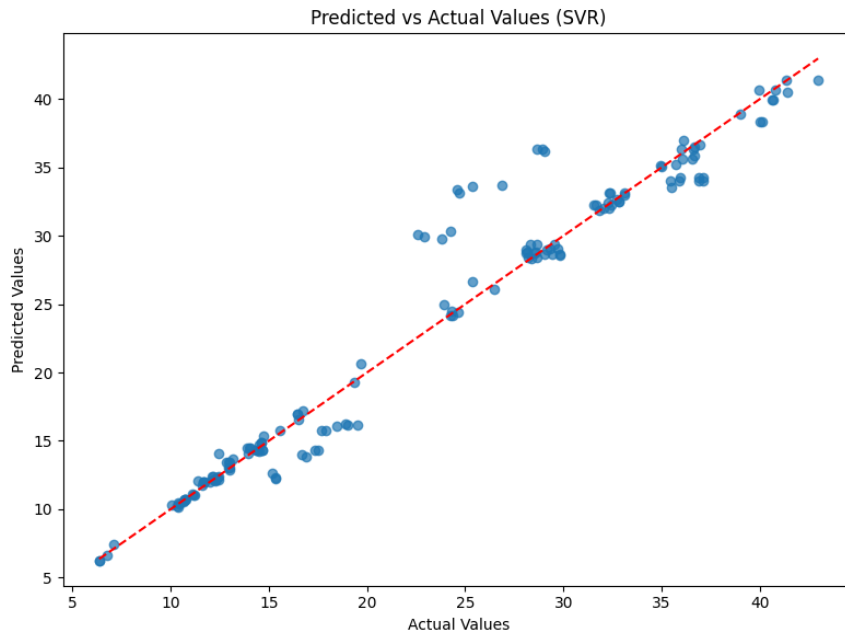


Figure 5: SVR predicted vs. actual values.

On the figure above, we can see the relationship between the actual and predicted values for the best SVR model, which, similarly to the Random Forest result, exhibits a predominantly linear alignment, though with somewhat greater spread around the diagonal. The evaluation metrics for the best SVR model are as follows:

Table 3: Evaluation of the best model (CVR with the reduced dataset).

Method	Value
R ² (Best SVR)	0.9509
MAE (Best SVR)	1.1811
RMSE (Best SVR)	2.2612

As seen in the earlier table, the R² value of 0.9509 indicates that the SVR model explains approximately 95% of the variance in the target variable, which is a strong result in absolute terms, though notably lower than the Random Forest model's optimized R² of 0.9762. The MAE values of the two models are nearly identical (1.1811 vs. 1.1817), suggesting that the average magnitude of individual prediction errors is comparable. However, the RMSE of the SVR model (2.2612) is higher than that of the Random Forest (1.5753), which indicates that the SVR model produces larger outlier errors, since RMSE penalizes larger deviations more heavily than MAE does.

3.4.4 Cross-Validation Stability

Beyond the single test set evaluation, the cross-validation R² scores provide a more reliable picture of how the model is expected to generalize.

The distribution of R² values across the five folds, which ranged from approximately 0.919 to 0.943, yielding a mean cross-validated R² of 0.934. This is meaningfully lower than the test set R² of 0.9509, which may suggest that the test split happened to be somewhat favorable for the SVR model. The moderate variance across folds also indicates that the SVR model's performance is somewhat dependent and points to slightly less stable generalization compared to the Random Forest.

3.5 Visualization of results

This section will present a visual summary of the Random Forest model's predictive performance results. The graphs below illustrate the relationship between actual and predicted values on the test set, as well as the effect of varying the number of estimators on model stability.

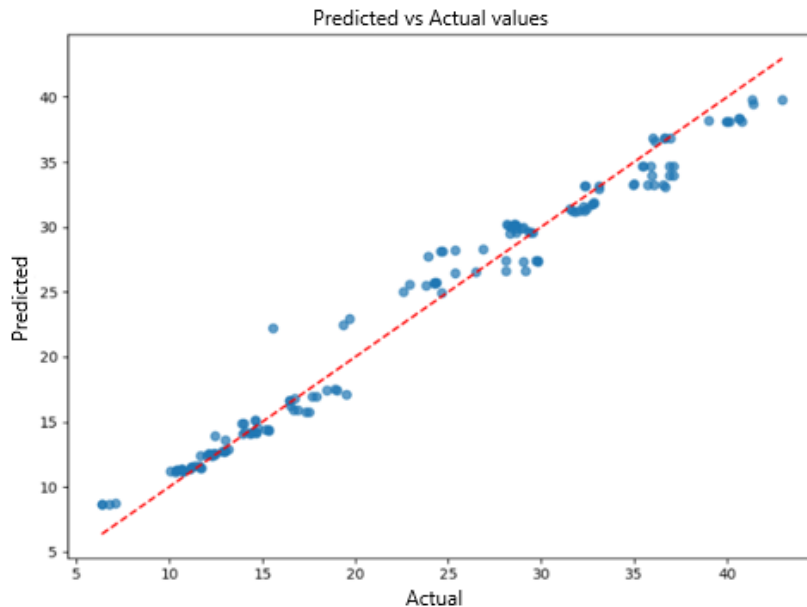


Figure 6: Evaluation metrics after hyperparameter optimization.

Model performance was visualized using two graphs. Figure 6 shows a narrow linear relationship between the actual and predicted values of the target variable. This strong linear alignment indicates that the model generalizes well to unseen data.

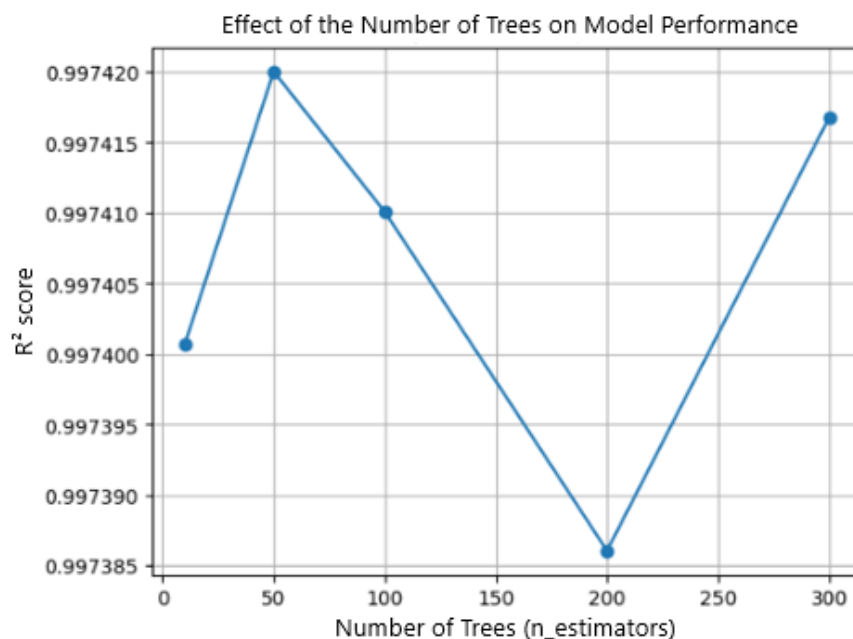


Figure 7: Number of trees optimization.

Figure 7 shows how the number of trees affects model performance. It illustrates how the R^2 value remains consistently high regardless of the number of trees, suggesting that the tree

count may not significantly impact the coefficient of determination compared to other hyperparameters.

4 MODEL COMPARISON

Overall, the SVR model serves as a solid and interpretable baseline, confirming that the target variable can be predicted with high accuracy even without ensemble methods. However, across all three primary metrics, the optimized Random Forest model outperforms SVR: it achieves a higher R^2 , a comparable MAE, and a substantially lower RMSE. Additionally, the Random Forest model's cross-validated performance was closer to its test set result, suggesting better and more stable generalization.

These findings motivate the choice of Random Forest as the primary model for the subsequent explainability analysis using LIME and SHAP, where feature attribution methods benefit from a model that more consistently captures the underlying data structure.

5 POST-HOC EXPLANATION OF THE MODEL

5.1 Applying the LIME method

LIME is a method used for interpreting local predictions. In other words, a single instance (one row in the dataset) is taken, and the method attempts to explain how the model arrived at that specific prediction.

First, a model is initialized that will be used to explain the prediction, with parameters related to the input data used for sampling the local neighborhood, the feature names, the model type (regression, in this case), the verbose parameter which prints results to the console to facilitate easier testing and a random state to ensure reproducibility. To explain one instance from the dataset, the input values for that instance are provided, after which the prediction algorithm is executed. Additionally, the desired number of features to include in the explanation is specified. With this, similar instances are generated, their predictions obtained through the model, and a simple local model is trained to explain the original instance.

Table 4: LIME local explanation for the instance.

Feature Condition	Contribution to Prediction
$0.25 < \text{Glazing Area} \leq 0.40$	2.9373
$1.25 < \text{Glazing Area Distribution} \leq 3.00$	0.3574
$\text{Relative Compactness} \leq 0.66$	-2.3003
$\text{Roof Area} \leq 3.50$	-6.6555
$147.00 < \text{Overall Height} \leq 220.50$	-6.8099

The table provides the explanation produced by the method. This makes it easier to see how the model arrived at a specific prediction for a selected instance. Based on conditions such as “Glazing Area between 0.25 and 0.40” the table shows how much that condition influenced the model to predict a value of 16.92 instead of the global average of 29.39. This global average represents the value the model would predict if no specific input values were provided.

To describe the table in more detail, each row is explained individually. Starting with the feature Glazing Area in the range of 0.25 to 0.40, which positively contributes to the prediction with a value of +2.94. Because the feature value falls within this range, the model determines that it increases the final prediction by about 2.94 units. Similarly, Glazing Area Distribution within the range 1.25 to 3.00 adds another 0.36 to the total prediction. On the other hand, conditions referring to low Relative Compactness, Roof Area, and Overall Height contribute negatively to

the prediction. For example, “Roof Area ≤ 3.50 ” reduces the predicted value by 6.66 units, while “Overall Height ≤ 220.5 ” reduces it by an additional 6.81.

Based on the provided input values, the model predicted a value of 16.92, while the actual value for those same inputs was 16.68. Since the difference in this context is very small, it can be concluded that the model predicted the result quite accurately.

5.2 Applying the SHAP method

This section goes through three different SHAP visualizations, each of which is described and interpreted in detail.

To begin, the first visualization is the waterfall plot, as shown in Figure 8. Using available SHAP documentation, the goal is to explain how the waterfall plot works.

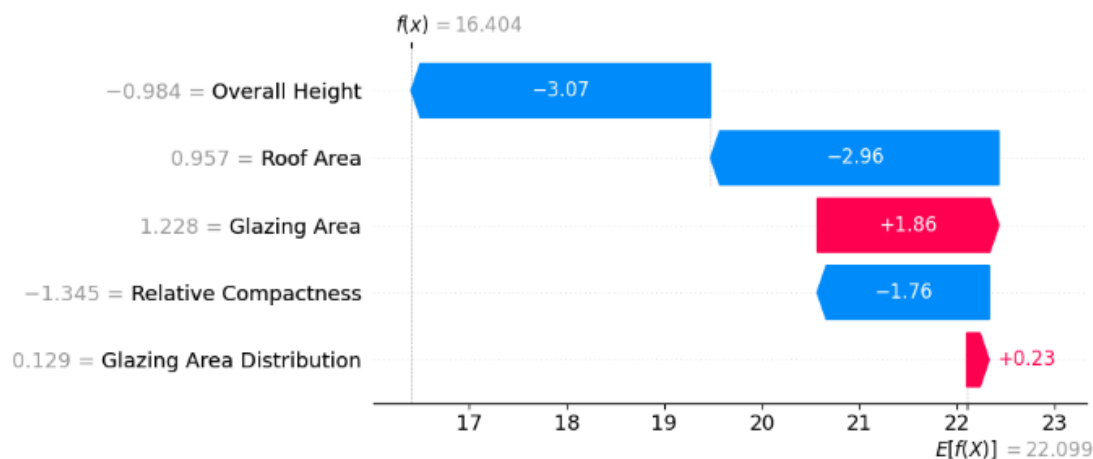


Figure 8: SHAP - waterfall.

This visualization provides an interpretation of the model’s *local* prediction for a single instance. The plot primarily shows the importance of each feature, or how much and in which direction it contributed to the prediction.

At the top of the plot, the actual prediction for the selected instance is visible (in this case, $f(x) = 16.404$), while at the bottom the average model prediction for all instances is shown ($E[f(x)] = 22.099$). In the center, all included features and their contributions are displayed. Besides numerical values, contributions can also be recognized by color: blue represents a decreasing effect on the prediction, while red represents an increasing effect [13].

For example, the feature Overall Height decreases the prediction by 3.07 units, while Glazing Area increases it by 1.86. Thus, this visualization provides an intuitive and clear overview of the contributions of all features for a specific instance. Similarly, if this were a classification problem instead of regression, the plot would show how the model arrived at a specific decision.

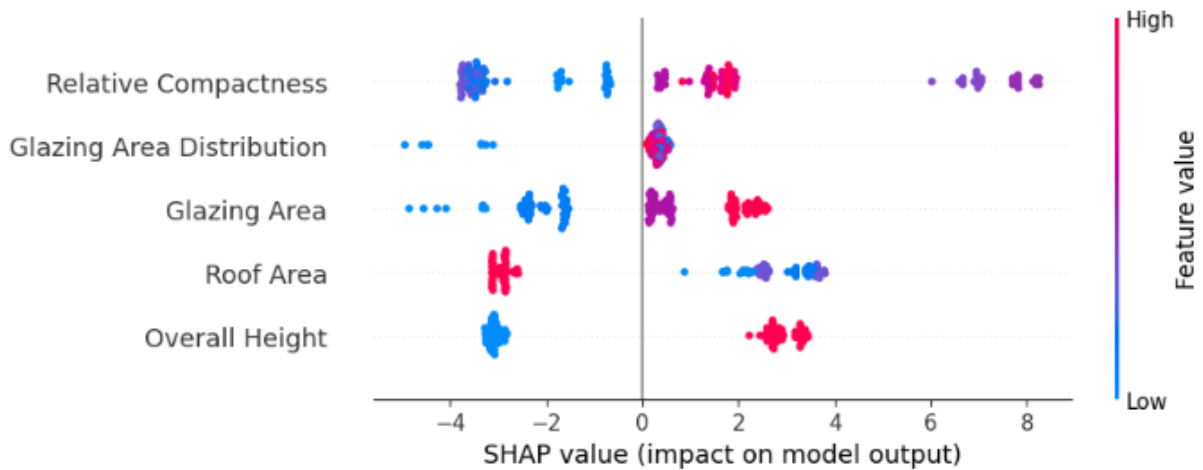


Figure 9: SHAP - beeswarm.

The next visualization is the beeswarm plot, as seen on Figure 9. Unlike the waterfall plot, the beeswarm plot shows the global influence of features on the model. Here, general feature contributions are analyzed, not only locally selected instances. Each instance is represented as a single dot in the graph, where its SHAP value is shown horizontally for each feature. Depending on whether the dot lies on the left or right side, it is possible to determine whether the contribution was positive or negative. Additionally, the color represents the actual value of the feature: red for high values, blue for low values [13].

For example, high values of Overall Height tend to increase prediction values (positive SHAP values). Conversely, for Relative Compactness, low values tend to decrease prediction values (negative SHAP values). This visualization therefore provides a global overview of the importance of attributes, useful for identifying general patterns in the data.

The third and final visualization is the bar plot. This is similar to the feature selection plot shown earlier, where the mean absolute SHAP value is shown, representing the average impact of each feature on predictions, regardless of whether it is positive or negative [13]. As seen in Figure 9, Relative Compactness, Overall Height, and Roof Area have the highest importance, consistent with the feature importance analysis conducted earlier during model development.

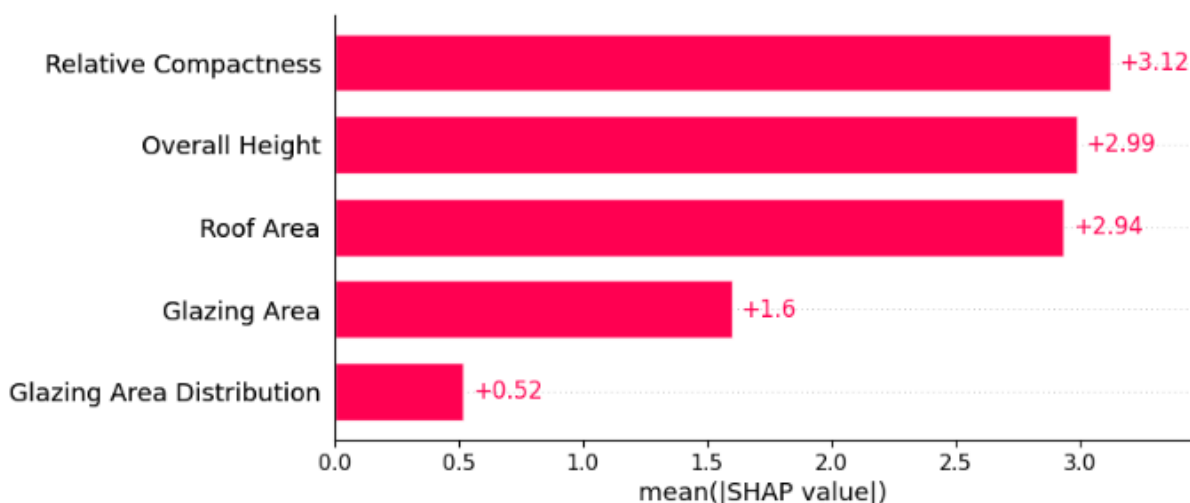


Figure 10: SHAP - bar.

Overall, it is clear that these three SHAP visualizations together provide a more comprehensive understanding and greater transparency of the model starting with local explanations for a chosen instance, followed by a global overview of how individual features generally affect predictions, and finally the average SHAP values per feature.

5.3 Comparison of model explanations (LIME vs. SHAP)

Having gone through the application of the two xAI methods and the review of their results, their comparison follows. Since both methods aim to explain how the model arrives at predictions, their effectiveness will be evaluated according to that goal. Additionally, the comparison will consider their methodological approaches and theoretical foundations. This provides insight into their differences, as well as the advantages and disadvantages of each.

First, local explanation of predictions is considered. LIME is designed almost exclusively for this purpose, focusing on explaining the prediction of a single instance. As shown earlier, similar instances are generated, and a local linear model is trained to determine the importance of features. Because it explains only local predictions and not global behavior, the method is faster and more efficient. SHAP, however, can also provide local explanations. The method is based on cooperative game theory and, specifically, Shapley values. Unlike LIME's approximation approach, SHAP considers the contribution of each feature by examining all possible combinations, implying that it captures a broader spectrum of the data domain. Although each combination does not require retraining the model, the process is still more computationally demanding.

Comparing the results, in LIME's case, the predicted value (16.92) did not differ significantly from the actual value (16.68), indicating high prediction accuracy. The method also showed feature contributions, such as Glazing Area having a positive influence and Roof Area and Overall Height having negative influences. In SHAP's waterfall plot, the prediction for one instance was 16.404, with an average prediction of 22.099. Regarding feature importance, SHAP also showed that Glazing Area and Overall Height positively influence predictions, while others reduce them. The strongest contributors in SHAP (Overall Height and Roof Area) are the same features that LIME identified as having the highest absolute impact.

Another issue with LIME is reduced robustness and reproducibility, since the method uses sampling for approximation for each instance. SHAP, on the other hand, produces more consistent results, although at a higher computational cost. The table below summarizes these observations.

Table 5: Comparison of LIME and SHAP model interpretation methods.

Criterion	LIME	SHAP
Basic approach	Local linear model	Cooperative game theory
Local interpretation	Yes	Yes
Global interpretation	No (or very limited)	Yes (beeswarm, bar chart)
Result stability	Dependent on sampling	High (deterministic)
Computational complexity	Low to medium	Medium to high
User intuitiveness	High - conditional statements easy to read	Medium - requires additional interpretation of visualizations
Explaining prediction relative to model average	No	Yes (e.g., waterfall chart)
Support for feature interactions	No	Yes
Visual tools	Text output	Waterfall, beeswarm, bar, etc.

From the comparison of local explanations, both methods point to similar results; the same features contribute positively or negatively, and the same features have the largest overall effect. However, SHAP goes further by showing how feature contributions relate to the model's average prediction. Additionally, while LIME offers only local explanations, SHAP provides insight into global patterns through various visualizations. It can show how features affect the model globally and what their average contributions are, matching the earlier feature importance analysis. LIME lacks this global interpretability.

In conclusion, both LIME and SHAP provide useful tools for local model interpretation, but only SHAP offers insight into global model behavior. SHAP stands out with greater stability and richer visualizations. However, LIME is simpler to understand and computationally less demanding, making it suitable for quick and concise analyses. Therefore, the choice of method depends on the specific interpretability goals and available resources.

5.4 Domain interpretation and critical analysis

While the preceding comparison establishes the methodological differences between LIME and SHAP, a deeper examination of the results reveals important insights from the perspective of building energy efficiency.

5.4.1 Feature relevance in the energy domain

The consistent identification of Glazing Area, Overall Height, and Roof Area as the most influential features aligns well with established knowledge in building thermodynamics. Glazing Area directly governs heat gain and heat loss through windows, making it one of the primary determinants of heating load. The positive contribution to predictions is therefore physically expected. Roof Area acts as a primary surface through which heat dissipates, and its negative influence on the predicted heating load suggests that buildings with larger roof surfaces relative to their volume tend to lose heat more efficiently, which is consistent with the role of the surface-to-volume ratio in thermal performance. Overall Height introduces vertical stratification effects and affects the compactness of a building, both of which influence energy demand.

The fact that both LIME and SHAP independently converge on the same set of dominant features strengthens confidence that these are genuine structural patterns in the data rather than artifacts of either method's approximation strategy. This parallel between the two is not trivial; where LIME constructs a local linear surrogate, SHAP attributes contributions across feature combinations. Their agreement lends additional credibility to the Random Forest model's learned representations.

5.4.2 SHAP interaction effects

One area where SHAP's analytical depth becomes particularly valuable is in the examination of feature interaction. In the context of building energy modeling, these interactions are meaningful: the effect of Glazing Area on heating load is unlikely to be independent of Overall Height, since taller buildings with more glazing face compounded heat loss through both vertical surfaces and increased exposure. Similarly, the interaction between Roof Area and Wall Area may capture the compactness coefficient implicitly, a known predictor of thermal efficiency.

On the other hand, if Glazing Area's positive contribution was amplified in buildings of greater height, this would represent a meaningful finding that neither a linear model nor LIME's local approximation could surface. With that said, even if the interaction effects are found to be modest in magnitude, this itself is an interpretable result, suggesting that the dominant features act relatively independently.

5.4.3 Critical limitations of the applied approach

Despite the strong quantitative performance of the model and the consistency of the xAI explanations, several limitations warrant acknowledgment.

First, the dataset used; the energy efficiency dataset is generated through building simulation software rather than collected from real buildings. This means the model has learned relationships that reflect the assumptions of the simulation tool, and the xAI explanations, however consistent, ultimately describe patterns within a synthetic space. Extrapolating these for real-world buildings should therefore be done cautiously.

Second, LIME's instability introduces a subtle problem that is easy to overlook in a single-instance analysis. Because LIME re-samples the neighborhood of each instance stochastically, repeated explanations for the same prediction may assign different feature importances. For example, this inconsistency in the scenario of an energy auditor using the model to justify a renovation decision could undermine trust.

Third, both LIME and SHAP are post-hoc explanation methods applied to a black-box model. While they provide insight into the model's behavior, they do not guarantee that the model's reasoning is causally correct. High feature importance for Glazing Area, for instance, reflects correlation within the training distribution, not necessarily a causal mechanism. This distinction matters in the energy domain requires causal confidence that a post-hoc explainer alone cannot provide.

5.4.4 Practical implications

Despite these limitations, the combined use of Random Forest with SHAP offers a viable framework for energy-efficient building analysis. The global SHAP summary (beeswarm plot) provides building designers and energy analysts with an interpretable ranking of which design parameters most strongly drive heating load predictions across the entire building stock represented in the dataset. The local waterfall explanation then allows case-by-case reasoning about individual building configurations. This two-level interpretability is precisely what transparent AI systems in engineering applications require, as it bridges the gap between statistical modeling and actionable domain knowledge.

6 DISCUSSION

With the comparison of results from the last two methods, the chapter related to the analysis and the core of this work has concluded. Now follows a critical review aimed at presenting all advantages and contributions of the work, as well as its shortcomings and limitations.

This work set out to demonstrate how machine learning, and specifically the Random Forest algorithm, can be applied to the problem of building energy efficiency prediction, and how the decisions of such a model can be made transparent through the use of explainable AI methods. Random Forest was selected for its strong predictive performance, robustness to overfitting, and suitability as a complex model whose internal reasoning is not directly human-readable. In fact, it's the prime model to be used as it benefits most from post-hoc explanation, due to the very nature of black box models. LIME and SHAP were then applied as the central analytical lens of the work, with the goal of understanding which building characteristics drive energy consumption predictions and whether those explanations are consistent, stable, and meaningful in a real-world engineering context.

The section on advantages will describe the practical application of xAI in real-world problems, the general transparency of this approach, and its usefulness in education and raising awareness. Regarding the disadvantages, the distinction between post-hoc explanations and interpretable models will be defined, along with general limitations of xAI methods and

constraints in their domain of application. The academic relevance and pedagogical contribution of the work will also be discussed.

Starting with practical application, this work clearly demonstrates how LIME and SHAP methods can be used to analyze models on an authentic dataset. The dataset used here was based on the physical characteristics of buildings and heating consumption, and the insight into the most important attributes that determine the level of energy efficiency of buildings highlights the contribution xAI can have in the real world. The work repeatedly touches on these themes, and research by Domingues shows how this insight gained from xAI can support practical decision-making by turning complex model outputs into interpretable data which then improves transparency, communication, and trust in intelligent systems [17].

In the context of building energy management, this means that architects, engineers, and energy planners can use model results explained by xAI to better understand how specific design decisions influence heating and cooling demand. For example, if LIME and SHAP analyses consistently identify glazing area as a major contributor to heating load, this may support decisions regarding window placement, insulation strategy, or facade optimisation during building design. Furthermore, explainable models can contribute to the future development and implementation of smart-building systems, where transparent and understandable predictions are important for monitoring, optimisation, and improving user trust in automated energy-management systems [17].

It is clear how important transparency is when designing and using any artificial intelligence framework that will at some point be used for decision-making or predictions that affect individuals' lives. To achieve this goal, issues such as multicollinearity, hyperparameter tuning, and optimization were addressed to ensure a solid foundation for applying methods that explain black-box models.

It can be concluded that this work is a good educational case study for students. Besides the application of the methods themselves, it also demonstrates data-cleaning phases, training, and evaluation, which are repeatedly identified as important steps in developing a reliable model. It also helps to define when it is better to use xAI methods and when to build interpretable models.

Next, we move on to the challenges associated with using xAI, starting with the difference between post-hoc explanations, used here to explain a black-box model, and implementing interpretable white-box models. This difference primarily becomes important in high-risk sectors. Rudin et al. argue in their work that it is better to build a model that is inherently interpretable rather than attempting to explain a black-box model, regardless of the explanation method used. The main reason is that for white-box models, there is no need to question the reliability or transparency of the model, since it is inherently present, although this typically comes at the cost of reduced precision compared to black-box models.

Moving on to the limitations of the methods themselves, SHAP is computationally expensive to run, while LIME is highly sensitive to changes in the data because it approximates local behavior when predicting an instance, and it also does not offer visualization of its results. Finally, another drawback worth mentioning is the potential limitation of the domain of application, since the dataset used here consisted of only about 700 instances, which is quite small, whereas the literature often discusses the value of these methods in the context of complex models.

7 CONCLUSION

Overall, this work provides insight into the practical application of tools for explaining machine-learning models and contributes to the academic community by defining and clearly presenting a comparison between two modern xAI methods, which serves as a basis for discussing the importance of model interpretability in real-world settings. The theoretical background that motivates and explains the importance of transparency in modern models was defined, followed by the development of the Random Forest model and the use of methods to explain its prediction results, which were then critically evaluated. This concept opens possibilities for further discussion about the justification of using black-box models and xAI methods versus already transparent white-box models.

Regarding academic significance, the work represents a high-quality educational resource because it illustrates all stages of developing a predictive model. From data cleaning and feature selection, through evaluation and hyperparameter tuning, to the final interpretation of predictions. This makes it a suitable teaching material for machine learning, xAI methods, and black-box modeling in general, especially in the context of data-driven decision-making in socially relevant domains.

8 ACKNOWLEDGEMENTS

This work is supported by Erasmus+ programme KA220-HED - Cooperation partnerships in higher education within project AI2SEP: Developing Talents in Artificial Intelligence to Solve - Disruptive Environmental Problems.

References

- [1] I. Goodfellow, Y. Bengio & A. Courville, *Deep Learning*. MIT Press, 2016., <http://www.deeplearningbook.org>.
- [2] F. Muharemi, D. Logofătu & F. Leon, „Machine learning approaches for anomaly detection of water quality on a real-world data set,” *Journal of Information and Telecommunication*, vol. 3, no. 3, pp. 294–307, 2019. DOI: 10.1080/24751839.2019.1565653.
- [3] A. Mosavi, M. Salimi, S. Faizollahzadeh Ardabili, T. Rabczuk, S. Shamshirband & A. R. Varkonyi-Koczy, „State of the Art of Machine Learning Models in Energy Systems, a Systematic Review,” *Energies*, vol. 12, no. 7, pp. 1301, 2019. DOI: 10.3390/en12071301.
- [4] C. Rudin, C. Chen, Z. Chen, H. Huang, L. Semenova & C. Zhong, „Interpretable Machine Learning: Fundamental Principles and 10 Grand Challenges,” *Statistics Surveys*, vol. 16, no. 1, pp. 1–85, 2022. DOI: 10.48550/arXiv.2103.11251.
- [5] M. T. Ribeiro, S. Singh & C. Guestrin, „Why Should I Trust You?”: Explaining the Predictions of Any Classifier,” *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, 2016., pp. 1135–1144. DOI: 10.1145/2939672.2939778.
- [6] M. Panda & S. R. Mahanta, „Explainable Artificial Intelligence for Healthcare Applications Using Random Forest Classifier with LIME and SHAP,” *Explainable, Interpretable, and Transparent AI Systems*, CRC Press, 2024., pp. 89–106, ISBN: 9781003442509.
- [7] A. Hussien, A. Maksoud, A. Al-Dahhan, A. Abdeen, and T. Baker, “Machine learning model for predicting long-term energy consumption in buildings,” *Discover Internet of Things*, vol. 5, no. 1, Mar. 2025, doi: 10.1007/s43926-025-00115-7.
- [8] F. Pedregosa, G. Varoquaux, A. Gramfort & et al., „Scikit-learn: Machine Learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [9] L. Breiman, „Random Forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001., ISSN: 0885-6125. DOI: 10.1023/A:1010933404324.
- [10] S. Zhou & L. Mentch, „Trees, Forests, Chickens, and Eggs: When and Why to Prune Trees in a Random Forest,” *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 16, no. 1, pp. 45–64, 2022. DOI: 10.1002/sam.11594.
- [11] T. Chai & R. R. Draxler, „Root mean square error (RMSE) or mean absolute error (MAE)? – Arguments against avoiding RMSE in the literature,” *Geosci. Model Dev.*, vol. 7, no. 3, pp. 1247–1250, 2014. DOI: <https://doi.org/10.5194/gmd-7-1247-2014>.
- [12] S. M. Lundberg & S.-I. Lee, „A Unified Approach to Interpreting Model Predictions,” *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS)*, pp. 4768–4777, 2017. DOI: 10.48550/arXiv.1705.07874.
- [13] S. Lundberg et al., SHAP Documentation, <https://shap.readthedocs.io/en/latest/>, 2023.
- [14] J. Priester. „Predicting a building’s energy efficiency,” Kaggle. (2022.), available: <https://www.kaggle.com/code/jarredpriester/predicting-a-building-s-energy-efficiency#6.Conclusion> (accessed 23. 6. 2025.).

- [15] V. Sheth, U. Tripathi & A. Sharma, „A Comparative Analysis of Machine Learning Algorithms for Classification Purpose,” *Procedia Computer Science*, vol. 215, pp. 422–431, 2022., ISSN: 1877-0509. DOI: 10.1016/j.procs.2022.12.044.
- [16] F. Abbas, S. Habib, D. Feng & Z. Yan, „Optimizing Generation Capacities Incorporating Renewable Energy with Storage Systems Using Genetic Algorithms,” *Electronics*, vol. 7, no. 7, pp. 100, 2018. DOI: 10.3390/electronics7070100.
- [17] N. S. Domingues, “nZEB beyond prediction in smart built environments: formalising engineering knowledge through modular explainable machine learning,” *Energy Informatics*, vol. 9, no. 1, Jan. 2026, doi: <https://doi.org/10.1186/s42162-025-00613-6>.

Sandra Sačarić has received a bachelor's degree in the field of Artificial Intelligence in Business from the Faculty of Organization and Informatics, University of Zagreb, and is currently pursuing a master's degree in Information and Software Engineering. Her research interests include machine learning, data analysis, and intelligent systems. While her recent research explored explainable AI for improving energy efficiency in smart buildings, she is also the author of the student paper Integrating Cognitive Agents into Visual Novels.

Similarly, *Ivan Simić* received a bachelor's degree in the same field - Artificial Intelligence in Business from the Faculty of Organization and Informatics - and is currently pursuing a master's degree in Databases and Knowledge Bases at the same institution. His research interests include databases, knowledge representation, with an interest in data-driven systems and intelligent information management.