A data-driven framework for fast building energy demand estimation across future climate conditions

Yukai Zou\textsuperscript{1,2,†}, Zhuoxi Chen\textsuperscript{1}, Jialiang Guo\textsuperscript{1}, Yingsheng Zheng\textsuperscript{1} and Xiaolin Yang\textsuperscript{1,*}
\textsuperscript{1}School of Architecture and Urban Planning, Guangzhou University, 230 Guangzhou Higher Education Mega Center West Outer Ring Road, Panyu District, Guangzhou, Guangdong 510006, China; \textsuperscript{2}State Key Laboratory of Subtropical Building Science, South China University of Technology, 381 Wushan Road, Tianhe District, Guangzhou, Guangdong 510640, China

\textbf{Abstract}

The rapid and precise forecasting of building energy requirements plays a crucial role in decision-making processes for architects during the early design phase. This study introduces a data-driven framework capable of projecting energy demands in the context of evolving climate conditions. We evaluated four widely-used machine learning algorithms. Our results indicated that a hybrid approach, integrating Catboost and Bayesian optimization, excelled in both accuracy and efficiency for predicting building energy demand under climate change conditions. The framework proposed herein has potential applications in fostering sustainability in early-stage architectural design.

\textit{Keywords:} climate change; future energy demand; data-driven model; parametric building simulation; Bayesian optimization

\textsuperscript{*Corresponding author.}
saupyangxialin@gzhu.edu.cn

Received 28 May 2023; revised 22 September 2023; accepted 18 December 2023

\section{1 INTRODUCTION}

As global economies develop, the demand for energy is projected to rise in the near future [1]. Nonetheless, energy consumption and global greenhouse gas (GHG) emissions are at the core of the current environmental crisis. The construction industry significantly contributes to the energy market and is a considerable source of greenhouse gas emissions [2]. Climatic conditions substantially influence building energy performance [3]. According to the Intergovernmental Panel on Climate Change (IPCC), global surface temperatures are expected to follow an upward trend at least until mid-century [4]. Numerous studies have reported shifts in building energy demand across different regions due to global warming [5–8]. In certain regions, cooling demands are likely to rise significantly while heating demands could considerably decrease [9]. Accounting for uncertainties in climate change is vital for robust building design.

Currently, methods for predicting building energy demand are broadly classified into two types: physical models and data-driven methods [10]. The physical models endeavor to forecast building performance by meticulously representing the intricate physical processes and interactions occurring within a building. Their primary advantage is their comprehensive nature and capability to depict real-world physical processes with high fidelity. However, they come with certain challenges. Physical models necessitate substantial computational resources, often resulting in protracted calculation durations [11]. Especially when considering multiple climate scenarios, which is crucial given the uncertain trajectory of future climates and increasing global temperatures, the computational costs escalate. This exacerbation of inefficiencies, due to expansive scenario analyses, has been observed and discussed in recent studies [12].
In contrast, data-driven methodologies harness historical datasets and deploy statistical or machine learning frameworks to infer building performance. Such methods, either through instantaneous forecasts or in tandem with optimization techniques, offer solutions tailored to distinct challenges [13]. As a result, data-driven approaches have gained popularity for predicting building performance in recent years [14]. These include forecasts for cooling and heating demand [15, 16], indoor environment [17–19] and electricity consumption [20, 21]. A variety of machine learning algorithms, such as artificial neural networks and support vector machine, have been employed in this field [22]. By rigorously evaluating these algorithms through a combination of cross-validation techniques, performance metrics such as Root Mean Square Error and Mean Absolute Error, as well as analyzing their ability to generalize on unseen data, one can pinpoint the optimal model tailored for a designated task. Additionally, considering factors like computational efficiency, model interpretability, and ease of implementation further aids in discerning the most appropriate model for specific building performance prediction scenarios. However, a notable caveat of data-driven approaches is their potential inadequacy in encapsulating intricate physical processes as robustly as their physical model counterparts [14].

Uncertainties in future climate change are believed to significantly impact building energy demand [23]. However, as far as we are aware, these factors have not been adequately addressed in current building performance optimization studies [24]. Furthermore, the performance of different machine learning algorithms for assessing building life-cycle operational energy under climate change remains a subject for exploration. The novelty of this study lies in the comparison of the performance of different machine learning algorithms concerning building life-cycle operational energy prediction under climate change effects. We utilized Bayesian optimization techniques for hyperparameter tuning to enhance the predictive ability of the model. The insights derived from this study are anticipated to aid in formulating energy-efficient design strategies and promote sustainable building practices.

Beijing’s unique climatic characteristics, especially its trajectory of anticipated challenges, warrant meticulous scrutiny. Geographically ensconced in the northern ambit of China (116°28′E, 39°48′N), Beijing has historically been characterized by its cold climate. Yet, contemporary forecasts presage a marked thermal escalation in Beijing within this century’s purview. These impending temperature fluxes portend a surge in cooling demand, offset by a potential decrement in heating requisites [9, 12, 25]. This study’s lens narrows to emphasize office buildings, given their important footprint in urban architectural landscapes, thereby helping to explain the intricate energy dynamics inherent to the metropolis.

2 METHODOLOGY

Data-driven models are constructed using reliable data sources. In this study, we utilize EnergyPlus, a widely-accepted building simulation engine, verified through various experiments [26, 27], to generate future building performance data for the development of machine learning models. As Figure 1 presents, the workflow comprises seven steps. Step 1 to Step 3 generate raw building performance data under various future weather conditions. Step 1 involves the calculation of future hourly weather variables required for building simulation, achieved by downscaling data for future typical years within the General Circulation Model (GCM). Step 2 involves defining the target building information, including geometric dimensions and envelope construction of the building. In Step 3, we carry out batch building simulations using a parametric tool. Steps 4 and 5 involve the preparation of data and algorithms for training the machine learning model, respectively. In Step 6, we use a Bayesian optimization technique for hyperparameter tuning, which enhances the predictive performance of the data-driven models. Finally, in Step 7, optimizations under varying climate conditions are performed, and the optimal results and solutions are compared. Except for the building simulations performed using Grasshopper, all other processes are executed in Python.

2.1 Hourly weather data for future typical years

Weather conditions fluctuate over time. A typical meteorological year (TMY) is a synthetic representation comprising 12 typical meteorological months [28]. This is generated using past weather data spanning 10 to 30 years, with the aim of providing a climatic foundation for current local building performance calculations. In light of concerns about climate change and future building energy demand, future TMY data is being developed to predict potential changes in building performance [29, 30]. The future TMYs represent the average climate condition for a 10-year period in this study. Since climate change is mainly manifested by rising temperature [30], the Typical Meteorological Month (TMM) is chosen based on the monthly average temperature. The month with the average temperature closest to the decadal average is taken as the TMM [12].

Essential variables for building performance calculations can be derived from General Circulation Models (GCMs), which are mathematical models depicting the general circulation of a planet’s atmosphere or ocean. In this study, we use future weather data, including dry bulb temperature, relative humidity, solar radiation, wind speed and atmospheric pressure, from GISS-E2-1-G as boundary conditions for calculating energy demand. Four climate scenarios are considered, namely RCP2.6, RCP4.5, RCP7.0 and RCP8.5. The intensity of climate change under these four scenarios is from weak to strong. Consequently, 32 future TMYs, representing the period from the 2020s to the 2090s across the four scenarios, are generated to depict future climate conditions. Notably, hourly weather data is necessary for building simulations. However, time resolution of GCMs fail to meet this requirement. The weather data in GCMs must be downscaled in some way before it can be used for building simulations. Among the existing methods, ‘Morphing’ is popular for its simplicity of use and effectiveness [31] that modifies the historical TMY data according to
the GCMs. Details of the 'Morphing' method are found in Ref. [32].

2.2 Characteristic of the target building
Office buildings, while serving as linchpins of urban commerce and activity, are also critical nodes in the urban energy and carbon nexus. Addressing their energy consumption and resultant carbon emissions is not only pivotal for sustainable urban development but also imperative in the global endeavor to mitigate climate change. In the present study, we focus on a prototypical rectangular, five-storey office edifice, as depicted in Figure 2. This structure exemplifies a standard architectural layout, wherein the building’s core is strategically positioned in the center, flanked by designated workspaces. The core’s spatial dimensions—both in terms of length and breadth—account for precisely one-third of the building’s overall metrics. Detailed design parameters with their corresponding ranges are delineated in Table 1, which are set according to the common scenes in building construction. The office structure is characterized by a set of 19 design determinants, bifurcated into two salient categories. The first, pertaining to the building’s geometric configuration, encapsulates parameters such as orientation, dimensional extents, window-to-wall ratio (WWR), and projections of overhangs. In contrast, the construction-oriented category incorporates aspects like the solar absorptance coefficient of the façade and roofing, as well as the specific type of glazing employed. Comprehensive thermal attributes associated with the roofing, exterior walling, and glazing elements are enumerated in Table 2. In terms of operational dynamics, the building aligns with the stipulations set forth by the Chinese regulatory framework (GB50189–2015). Adhering to a typical workday, the edifice remains functional from 8 a.m. to 7 p.m., with the cooling and heating thermostatic setpoints meticulously calibrated at 26°C and 20°C, respectively, ensuring both typicality and rationality in the building’s design and function. The ideal air conditioning system was used in the simulations.

![Figure 1. Workflow of this study.](image)

<table>
<thead>
<tr>
<th>Figure 1. Workflow of this study.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Table 1. Design parameters of the target building.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Design parameter</td>
</tr>
<tr>
<td>Building geometry</td>
</tr>
<tr>
<td>Orientation</td>
</tr>
<tr>
<td>Building side length x</td>
</tr>
<tr>
<td>Building side length y</td>
</tr>
<tr>
<td>WWR× (Table 2)</td>
</tr>
<tr>
<td>Shading length× (Table 2)</td>
</tr>
<tr>
<td>Wall construction type</td>
</tr>
<tr>
<td>Wall solar absorptance</td>
</tr>
<tr>
<td>Roof construction type (Table 2)</td>
</tr>
<tr>
<td>Roof solar absorptance</td>
</tr>
<tr>
<td>Window type× (Table 2)</td>
</tr>
</tbody>
</table>

*Values of these parameters are different for the four sides of the building.*
2.3 Future building performance data generation for training

In this study, we utilize the well-regarded building simulation engine, EnergyPlus, to generate future building performance data, which is crucial for training our machine learning model. Batch executions of building simulations with EnergyPlus are facilitated through Honeybee, a plugin of the visual programming platform, Grasshopper. We generate simulation tasks employing Latin Hypercube Sampling (LHS), a stratified sampling technique renowned for its ability to generate representative samples from multivariate parameter distributions. Within this sampling framework, we treat the future period (ranging from the 2020s to the 2090s, denoted as 0 to 7) as one of the significant features. Consequently, 20,000 annual energy demand simulations are performed under each future scenario, with the results meticulously documented in a .csv file.

Before the machine learning models can be trained, the simulation results require preprocessing. Initially, we rigorously check and cleanse the data to remove any invalid records, which are often a byproduct of conflicts within the simulation program. Failing to eliminate such records could detrimentally impact the accuracy of our machine learning model. Subsequently, we normalize the data with a view to expedite the training process and enhance model accuracy. To this end, we chose the Z-score normalization method for processing the data. The equation is as follows:

\[ Z = \frac{X - \mu}{\sigma} \]

where \( Z \) is the normalized score; \( X \) is the raw value to be standardized; \( \mu \) and \( \sigma \) are the mean and standard deviation of the sample.

2.4 Machine learning models and hyperparameter tuning method

In this study, we employ four distinct machine learning models that have gained popularity in the field of building performance prediction. They are namely, Artificial Neural Network (ANN), Support Vector Regression (SVR), Random Forest (RF), and CatBoost. These models are constructed on various logical frameworks as explained below. The tuning of hyperparameters for these different algorithms is performed utilizing the Bayesian optimization technique. The involved hyperparameters along with their respective value ranges are provided in Table 3. We established these models using Python as the programming language.
Table 3. Hyperparameters being tuned in training and their value ranges.

<table>
<thead>
<tr>
<th>Hyperparameter</th>
<th>ANN</th>
<th>SVR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Learning rate</td>
<td>[10^{-5}, 0.1]</td>
<td>Hyperparameter</td>
</tr>
<tr>
<td>Dense nodes in layer 1</td>
<td>[4, 256]</td>
<td>Value range</td>
</tr>
<tr>
<td>Activation function</td>
<td>relu, sigm, tanh</td>
<td>Initial setting</td>
</tr>
<tr>
<td>RF</td>
<td>[50, 500]</td>
<td>Data type</td>
</tr>
<tr>
<td>Max depth</td>
<td>[10, 100]</td>
<td>Kernel function</td>
</tr>
<tr>
<td>Max features</td>
<td>[2, 10]</td>
<td>C</td>
</tr>
<tr>
<td>Max leaf nodes</td>
<td>[10, 1000]</td>
<td>Gamma</td>
</tr>
<tr>
<td>Catboost</td>
<td></td>
<td>[10^{-9}, 1000]</td>
</tr>
<tr>
<td>Hyperparameters tuning</td>
<td></td>
<td>10^{-2}</td>
</tr>
</tbody>
</table>

Figure 3. Average temperature of future TMY under different climate scenarios of Beijing.

2.4.1 Artificial neural network (ANN)

An Artificial Neural Network (ANN) is a mathematical model designed to replicate the structure and functionality of a biological neural network. At its core, an ANN is made up of interlinked processing units known as artificial neurons or nodes. These nodes are arranged into layers, typically comprising an input layer, one or more hidden layers, and an output layer. The nodes in each layer connect to those in the adjacent layers via weighted connections, which signify the strength of the links. During the learning process, these weights are adjusted to enhance the network's performance. Data is processed from the input layer through the hidden layer(s), with the final results presented by the output layer.

Assuming that a neural network receives M inputs, denoted by $x_1 \ldots x_M$. Let input $x$ be used to represent the weighted sum of the input signal $x$ obtained by the neuron, which expresses as the following equation:

$$z = \sum_{m=1}^{M} w_m x_m + b$$  \hspace{1cm} (1)

where $w$ is the weight and $b$ is the bias. After the net input $z$ passed through a nonlinear function $f(*)$, the activation value $a$ can be calculated:

$$a = f(z)$$  \hspace{1cm} (2)

where $f(*)$ is called the activation function. The ANN propagates information by iterating the following equation:

$$z^{(l)} = W^{(l)} a^{(l-1)} + b^{(l)}$$  \hspace{1cm} (3)

$$a^{(l)} = f_l(z^{(l)})$$  \hspace{1cm} (4)

where $z^{(l)}$ and $a^{(l)}$ are net input and output of layer $l$, respectively. $W^{(l)}$ and $b^{(l)}$ are the weight matrix and bias between layers $l - 1$ and $l$, respectively. $f_l(*)$ represents the activation function of neurons in layer $l$. Taking vector $x$ as the input $a^{(0)}$ of the first layer and the output of layer $L$ as the results of ANN, the entire information transfer process can be described as:

$$x = a^{(0)} \rightarrow z^{(1)} \rightarrow a^{(1)} \rightarrow z^{(2)} \rightarrow \ldots \rightarrow a^{(L-1)} \rightarrow z^{(L)} \rightarrow a^{(L)}$$  \hspace{1cm} (5)

2.4.2 Support vector regression (SVR)

Support Vector Machines (SVMs) are widely used in classification and regression problems. They are employed to analyze cases that are linearly separable. In situations where linearity cannot be achieved, SVMs map the linearly inseparable samples in the low-dimensional input space into a high-dimensional feature space using a kernel function, making them linearly separable. Support Vector Regression (SVR) is a form of SVM used for regression.
Data-driven framework for fast building energy demand estimation

Figure 4. Simulation results of heating and cooling demand of different future periods under the four climate scenarios.

problems, which can be presented as:

\[ f(x) = \sum_{i=1}^{m} (\hat{\alpha}_i - \alpha_i) \kappa(x_i^T x) + b \]  

\[
\text{s.t.} \quad \sum_{i=1}^{m} (\hat{\alpha}_i - \alpha_i) = 0 \quad 0 \leq \hat{\alpha}_i, \alpha_i \leq C
\]

Where \( \hat{\alpha}_i \) and \( \alpha_i \) are the Lagrange multipliers and \( C \) is the penalty parameter. \( \kappa(x_i^T x) \) represents the kernel function. Commonly used kernel functions include radial basis function, sigmoid, etc.

2.4.3 Random Forest (RF)
Random Forest (RF) is an ensemble learning algorithm that consolidates multiple decision trees (DTs) based on the principle of bagging. A DT, which is a tree-like model of decisions, forms the basic unit of the RF. DTs are quick and straightforward to construct, consisting of nodes and directed edges, with two types of nodes: internal nodes, representing a feature or attribute, and leaf nodes, symbolizing a category or a value. RF constructs a multitude of independent DTs, creating a 'forest'. The prediction result of RF is determined by the collective voting of all trees. By introducing randomness into the training process, RF is less susceptible to overfitting than a single DT. Training data for each subtree is generated via random sampling, and the features for each subtree are randomly selected to ensure diversity, thereby boosting the robustness of RF. For regression problems, RF aggregates the predictions from multiple decision trees to yield a final estimate, which can be illustrated as:

\[ \hat{y}_{RF}(x) = \frac{1}{K} \sum_{k=1}^{K} \hat{y}_k(x) \]

where \( \hat{y}_{RF}(x) \) is the prediction of the RF model for input \( x \); \( K \) denotes the number of trees; \( \hat{y}_k(x) \) represents the prediction of the \( k \)th tree for input \( x \).

2.4.4 Catboost
Boosting algorithms, another type of ensemble learning, aim to convert a weak learner into a robust one during the training process. Initially, a base learner is established with an initial training set. Subsequently, the weight of the samples misclassified by the base learner is increased to ensure they gain more attention.
in the next round of training. This allows the next base learner to be trained using the adjusted samples. This two-step process repeats until the number of learners meets a specified condition. The results of regression problems can be computed via weighted average predictions from the learners. Catboost is a boosting algorithm developed by Yandex [33, 34]. One of CatBoost’s key differentiators is its treatment of categorical features. Instead of traditional encoding methods, CatBoost uses a technique known as ‘ordered boosting’ and an innovative way of processing categorical variables using statistics (like mean) on the target variable. Mathematically, for a categorical feature \( C \) with categories \( c_1, c_2, \ldots, c_K \), and an instance with category \( c_k \), the algorithm computes a numerical value based on the target statistics of all prior instances in the training dataset with category \( c_k \). This approach tends to be more expressive than traditional encoding methods.

2.4.5 Bayesian optimization
Bayesian optimization is a method often utilized to solve complex ‘black-box’ optimization problems characterized by two main traits: the objective function and its derivatives are unknown, and the computational cost of evaluating the objective function is high. Hyperparameter optimization in machine learning serves as a quintessential example of such problems. It’s widely acknowledged that hyperparameters are crucial to the accuracy of machine learning models. The primary concept behind Bayesian optimization is the continuous updating of the posterior distribution of the objective function by progressively adding samples. In simpler terms, it incorporates information from prior training to adjust the hyperparameters.

2.4.6 Model evaluation metrics
Three metrics are employed to evaluate the predictive performance of different models in this study, which are coefficient of determination (\( R^2 \)), root mean square error (RMSE) and mean absolute error (MAE). The formulae of the metric are given below:

\[
R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2}
\]  

Figure 5. Comparison of ANN predictive results and simulated results.
where $y$ is the actual value, $\bar{y}$ is the arithmetic mean of $y$, and $\hat{y}$ is the predictive value.

### 3 RESULTS

#### 3.1 Temperature variation of a future typical year

Beijing exhibits a continental monsoon climate characterized by its distinct seasonality. Affected by climate change, the future temperature of Beijing might rise substantially. Figure 3 depicts Beijing’s evolution of average temperature for the future TMYs over the course of the 21st century. In the most conservative scenario, RCP2.6, temperatures average around 16.0°C during the 2020s and 2030s, experiencing a decrease by 2.0°C in the 2040s. Subsequently, they fluctuate between 15°C and 16°C over the ensuing three decades. Under the RCP4.5 and RCP6.0 scenarios, there’s an alternating rise in temperature, with the trend being more pronounced. In the 2020s, both scenarios register temperatures marginally above 15.0°C. For RCP4.5, a decline is observed from the 2030s to the 2040s, followed by a surge exceeding 16.0°C in the 2050s—the apex across all scenarios. Between the 2060s and 2080s, temperatures remain below 15.5°C, eventually reaching 17.0°C by century’s end. In contrast, the RCP6.0 trajectory maintains temperatures around 15°C from the 2020s through 2040s, ascends to approximately 15.7°C in the 2050s, declines to 15°C, and oscillates between 16.0°C and 16.7°C during the final three decades of the century. Notably, despite RCP6.0 denoting greater radiative forcing than RCP4.5, the terminal century temperature under RCP4.5 surpasses that of RCP6.0. This counterintuitive outcome may stem from intricate interactions within Earth’s climate system, among other factors. For the RCP8.5 scenario, there’s a conspicuous upward temperature trajectory. By mid-century, the mean temperature gravitates around 16°C, marking a 1°C increment compared to the 2020s and 2030s, culminating in nearly 18°C by the 2090s. Such temperature divergences across these climate scenarios...
could engender profound shifts in future building thermal requirements.

3.2 Simulation records under different climate scenarios

The simulation results of future building heating and cooling demand are categorized by future periods and climate scenarios, as illustrated in Figure 4. It’s evident that rising temperatures can lead to decreased heating demand and increased cooling demand. Under the RCP2.6 trajectory, a marked upswing in heating demand is anticipated in the 2040s and 2050s, linked with diminished average temperatures. The zenith is reached in the 2050s, surpassing 90 kWh/m², which is roughly 34 kWh/m² higher than the median cooling demand. This is congruent with the 2050s registering the nadir in mean air temperatures across all timelines and scenarios. Post the 2070s, the median heating demand stabilizes around 40 kWh/m² for the century’s remainder. For both RCP4.5 and RCP6.0, heating and cooling demands demonstrate alternating dominance, albeit the disparities between them aren’t as stark as in other climatic scenarios. Within the RCP4.5 scenario, cooling demand overshadows heating demand during the 2050s and 2090s, correlating with mean decadal temperatures surpassing 16°C and 17°C, respectively, notably elevated compared to other periods. Conversely, under RCP6.0, cooling demand assumes prominence from the 2070s onward until the 2090s, whereas heating energy consumption takes the lead in the 2060s and earlier. Of particular note are the 2030s and 2060s, where the mean temperature hovers around the 15°C mark, the lowest across this scenario. Pertaining to the most extreme climate alteration framework, RCP8.5, there’s a tangible curtailment in heating demand accompanied by a marked augmentation in cooling demand. In the initial decade of the 2020s, median heating and cooling burdens are proximate, estimated at 50 kWh/m² and 45 kWh/m², respectively. By the 2090s, median heating demand dwindles to 30 kWh/m², a 40% reduction relative to the 2020s, while the median cooling demand plateaus around 50 kWh/m² from the 2070s through the 2090s. These elucidations underscore that future building energy requisites are profoundly contingent on both chronological spans and the respective climate scenario.
3.3 Predictive performance of different algorithms

The predictive performance of the different algorithms on the test sets is depicted in Figures 5 to 8. The optimized hyperparameters of the machine learning models are shown in the Appendix. As per the evaluation metrics, both Catboost and ANN distinctly outperform the other two algorithms across all climate scenarios. All $R^2$ values for Catboost are no less than 0.997, signifying that Catboost can provide reliable predictions on future building energy performance. For ANN, all $R^2$ values exceed 0.99, which, although lower than those of Catboost, are still highly commendable. The MAE and RMSE values for ANN range from 4.76 MWh to 6.17 MWh and 6.54 MWh to 8.22 MWh, respectively, which is acceptable considering that the prediction targets are hundreds of MWh.

The more scattered data points in Figure 7 clearly illustrate that the predictive performance of RF does not match that of Catboost and ANN. The $R^2$ values for RF are noticeably lower, ranging between 0.957 to 0.979 in varying scenarios. The MAE and RMSE values for RF are around 15 MWh and 20 MWh, which are also higher than Catboost and ANN. Among the four algorithms, SVR demonstrates the poorest predictive performance. The MAE and RMSE can be as high as 40.83 MWh and 53.62 MWh, respectively. Its $R^2$ value falls below 0.9 for cooling demand predictions under RCP4.5, and heating demand predictions under RCP2.6, RCP6.0, and RCP8.5. SVR is more precise in estimating cooling load than heating load, with the highest $R^2$ value for heating load prediction being 0.921 and the lowest being 0.838. The prediction error increases as the target value of energy demand escalates. This is evident from Figure 6, where data points deviate more from the fitted line as energy demand increases. Given the changing climate conditions during a building’s lifecycle, energy demand in future stages presents a certain degree of variability. SVR’s poor performance in predicting building life-cycle operational energy could be attributed to a mismatch between the algorithm’s inherent logic and the data characteristics. Each data point in the test set represents the energy performance of a building at a specific future period.

3.4 Convergence and computing expense of the training process

Figure 9 illustrates the historical best $R^2$ values for different models on the test sets following ‘n’ epochs of training. The poten-
In R² values is less conspicuous than in the other algorithms. Additionally, SVR is highly case-sensitive. Post-training, the best R² values range between 0.82 and 0.97, suggesting that the predictive performance of the SVR model is heavily influenced by the data. Aside from SVR, the models’ prediction accuracy significantly increases after several training rounds. RF converges fastest among the four algorithms, as the maximum R² values rise sharply in most cases and stabilize before five training rounds. Relative to ANN and Catboost, RF is more case-sensitive, as the best R² values in different cases hover around 0.95. In contrast, the best R² values for ANN and Catboost remain nearly identical across various climate scenarios, approaching 1.00. All instances of Catboost generally converge in the 11th round. Typically, ANN models require more training rounds than Catboost for convergence. As the figure demonstrates, the cases of cooling load under RCP2.6, heating load under RCP8.5, and cooling load under RCP6.0 essentially converge at calls 33, 24, and 16, respectively.

In terms of computing costs, Catboost holds a distinct advantage with superior calculation efficiency, followed by RF, while ANN requires the most time for training. The procedure runs on an AMD Ryzen 7 5800U device. On average, Catboost completes 50 epochs of training in 50 seconds, while RF and SVR take approximately 4.5 and 14.5 minutes to finish the training process, respectively. Training ANNs can be computationally intensive, potentially lasting as long as 2.5 hours, which is 180 times longer than Catboost training.

4 CONCLUSION

In contemporary research, data-driven models have emerged as pivotal tools for predicting building performance. While numerous studies have ventured into the realm of building energy prediction, a discernible gap persists regarding the comparative analysis of predictive capacities across varied machine learning algorithms, particularly in the context of life-cycle building operational energy demand under the auspices of climate change. In bridging this gap, this research proffers a nuanced data-driven framework, meticulously tailored for the expeditious prediction of life-cycle operational energy requisites. A rigorous case study set against the frigid terrains of China accentuates the potential implications of imminent climate shifts, such as elevated temperatures, on the very tenor of building energy demand. Such shifts hold profound repercussions for energy provisioning and intrinsic architectural design strategies. Of the scrutinized algorithms, both CatBoost and ANN stand out, exhibiting robust energy demand predictions across all envisaged scenarios. Especially Catboost, which R² values for both cooling load and heating load estimation in all climate scenarios are not less than 0.997. Compared with ANN, Catboost has tremendous advantages in computational cost; 50 epochs of Catboost training can be finished in 50 seconds, which is 1/180 of the time consumed by ANN. Therefore, it is recommended to give priority to the Catboost algorithm when creating a rapid prediction model of building future energy demand that takes into account climate change.
In summation, the inextricable linkage between building energy demand and prevailing climatic conditions renders the latter an indispensable variable in future energy demand projections. By shedding light on this, the study endeavors to catalyze the evolution of sustainable architectural paradigms, centralizing energy efficiency and carbon footprint minimization across the entire building operation period.

AUTHOR CONTRIBUTIONS
Yukai Zou (Conceptualization [Equal], Methodology [Equal], Validation [Equal], Writing—original draft [Equal], Writing—review & editing [Equal]), Zhaoxi Chen (Investigation [Equal], Software [Equal], Validation [Equal], Visualization [Equal]), Yingsheng Zheng (Data curation [Equal], Formal analysis [Equal]), and Xiaolin Yang (Conceptualization [Equal], Funding acquisition [Equal], Methodology [Equal], Supervision [Equal]).

ACKNOWLEDGEMENTS
This research was funded by National Natural Science Foundation of China (grant No. 52308016), Guangdong Basic and Applied Basic Research Foundation (grant No. 2023A151011364), Science, National Undergraduates’ Innovation and Entrepreneurship Training Program (grant No. S202311078012), and Technology Program of Guangzhou University (grant No. PT25202006).

REFERENCES
[4] IPCC. Climate change 2021: the physical science basis; 2021.


## APPENDIX A

Table A1. *Optimized hyperparameters of the different algorithms.*

<table>
<thead>
<tr>
<th>Hyperparameter</th>
<th>Cooling load</th>
<th>Heating load</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RCP2.6</td>
<td>RCP4.5</td>
</tr>
<tr>
<td>Learning rate</td>
<td>2.60×10^{-4}</td>
<td>6.65×10^{-4}</td>
</tr>
<tr>
<td>Dense nodes in layer 1</td>
<td>256</td>
<td>256</td>
</tr>
<tr>
<td>Dense nodes in layer 2</td>
<td>256</td>
<td>173</td>
</tr>
<tr>
<td>Activation function</td>
<td>relu</td>
<td>relu</td>
</tr>
<tr>
<td>Kernel function</td>
<td>sigmiod</td>
<td>rbf</td>
</tr>
<tr>
<td>C</td>
<td>4.50×10^{-4}</td>
<td>10^{-4}</td>
</tr>
<tr>
<td>Gamma</td>
<td>10^{-4}</td>
<td>10^{-4}</td>
</tr>
<tr>
<td>Estimators</td>
<td>103</td>
<td>54</td>
</tr>
<tr>
<td>Max depth</td>
<td>83</td>
<td>94</td>
</tr>
<tr>
<td>Min samples leaf</td>
<td>10</td>
<td>4</td>
</tr>
<tr>
<td>Max features</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Max leaf nodes</td>
<td>314</td>
<td>649</td>
</tr>
<tr>
<td>Iterations</td>
<td>490</td>
<td>307</td>
</tr>
<tr>
<td>Learning rate</td>
<td>1.16×10^{-1}</td>
<td>6.48×10^{-1}</td>
</tr>
<tr>
<td>I2 leaf reg</td>
<td>4.50</td>
<td>8.79</td>
</tr>
<tr>
<td>Bagging Temperature</td>
<td>0.63</td>
<td>6.39×10^{-5}</td>
</tr>
<tr>
<td>Random strength</td>
<td>19.01</td>
<td>9.85</td>
</tr>
</tbody>
</table>