

Air Conditioner placement optimization with CFD and Neural Network

Thomas IDIER^{a,b,*}

^aDepartment of Architecture and Building Science, Graduate School of Engineering, Tohoku University, 6-6-11 Aoba, Aramaki, Aoba-ku Sendai 980-8579, Japan

^bCentraleSupelec, Gif-sur-Yvette, France

ARTICLE INFO

Keywords:

Neural Network
PMV prediction
Computational Fluid Dynamics
AC placement optimization

ABSTRACT

Energy management and indoor environment play crucial roles in daily human life, especially given the current challenges posed by climate change and increasing energy costs. Computational fluid dynamics (CFD) is utilized with various boundary conditions to assess indoor environments using metrics such as the Predicted Mean Vote (PMV), which can be directly computed using software like FlowDesigner. However, CFD simulations are time-intensive. To address this challenge, Neural Networks (NN) are employed to predict the PMV distribution of indoor environments rapidly. This study focuses on optimizing the placement of an Air Conditioner (AC) within a bedroom at University House Aobayama in Sendai. Two Neural Network models for summer and winter are developed where inputs include parameters such as Sol Air Temperature (SAT), AC power, and AC position, while the outputs predict the PMV distribution within the bedroom. The two NNs have less than 7% of mean prediction error. The optimization approach aims to identify the optimal position for the AC that ensures a specified PMV range within the room while minimizing energy consumption. The findings indicate that during summer, the optimal position for the AC unit is away from the window, whereas during winter, the optimal position is near the window. However, in comparison to winter, summer is less significant, making the best position near the window. Furthermore, NN enables this methodology to operate 4,653 times faster than doing only CFD simulations.

1. Introduction and Hypothesis

Building energy consumption is a major contributor to overall energy use, with air conditioning being particularly significant in Japan. Since 2012, 90% of households with more than two members have installed air conditioners (1). Moreover, climate change is expected to increase the annual cooling load for residential buildings in Japan (2).

So, in response to the growing focus on energy conservation, engineers are working to determine the optimal positioning of air conditioners. This aims to maximize residents' comfort while minimizing energy consumption (3) (4).

To attain this goal, CFD simulations are conducted with different boundary conditions. CFD simulation predicts detailed airflow, thermal, and pollutant concentration distributions within the target space (computational domain), and thus, has been widely used in indoor environment research. However, CFD simulations are time-consuming, and to find the best AC position inside a room with different boundary conditions, you need to do hundreds or thousands of CFD simulations.

To address the time-consuming challenge, some methods have been developed such as fast fluid dynamics (FFD) that can speed up the CFD processing time by 20-50 times (5). Despite these advancements, it still takes too much time for thousands of simulations.

In recent years, Neural Network and Artificial Intelligence (AI) are often used to predict output data with high speed and accuracy with nonlinear relations by training on a database. Especially, NN has been used to predict indoor airflow in 2D (6), which is 1.9 millions faster than CFD simulation and in 3D (7). Therefore, coupling NN and building energy simulations (BES) is way faster than coupling CFD and BES, it offers a significant reduction in computational time - approximately 94% of reduction time (8).

This study focuses on optimizing the placement of an Air Conditioner within a bedroom at University House Aobayama (Figure 1) in Sendai by using Neural Network to access the PMV field inside. To simplify the problem, the following hypotheses are proposed:

- Man studying during the evening
- Man sitting on the bed or on the chair
- Steady state
- Heat transmission across glass window and wall connected to the outside only
- No transmitted solar heat (evening, no sun)
- No heat transmission through wall, floor and ceiling (not connected to outside)

*Corresponding author

✉ thomas.idier@student-cs.fr (T. IDIER)

https://github.com/thoid/CFD-with-NN (T. IDIER)



Figure 1: University House Aobayama bedroom

2. CFD modeling

2.1. Room modeling

2.1.1. Dimensions

To predict the PMV field inside using NN, CFD simulations were conducted to build a database. Therefore, the first step was modeling the bedroom at University House Aobayama using the CFD software Flowdesigner. The dimension of the room is $2m \times 5m \times 2.5m$. Furthermore, the modeling of all the furniture in the room, including the bed, desk, and wardrobes were done in Flowdesigner.

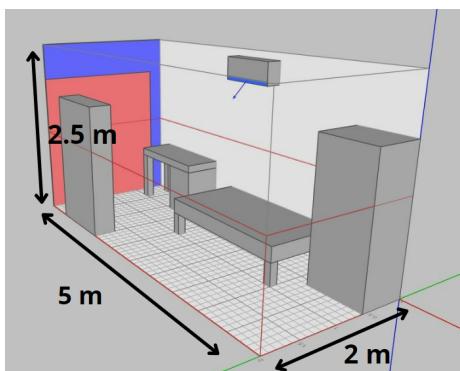


Figure 2: Room modeling

2.1.2. Wall

The outdoor temperature has an influence on the temperature inside, and to watch the impact of the outdoor temperature on the PMV field inside, the calculation of the heat transfer at the wall connected to the outside (area in blue in Figure 2) was done. Here are the different values used depending on the season (9).

	Heat Transfer Coefficient ($\text{W m}^{-2} \text{C}^{-1}$)
h_{inside}	8.29
$h_{\text{outside}_{\text{summer}}}$	22.7
$h_{\text{outside}_{\text{winter}}}$	34

Table 1

Combined convection and radiation heat transfer coefficient at wall and window

The wall area connected to the outside is obtained with the following equation:

$$A_{\text{wall}} = A_{\text{side}} - A_{\text{window}} = 2.5 \times 2 - 1.8 \times 2.0 = 1.4 \text{ m}^2 \quad (1)$$

To calculate the heat flow for the wall with thickness $L = 0.2m$ and heat transfer coefficient $\lambda_{\text{concrete}} = 1.4 \text{ W m}^{-1} \text{C}^{-1}$, the following equations are used:

$$\begin{aligned} U_{\text{wall}_{\text{summer}}} &= \frac{1}{\frac{1}{h_{\text{inside}}} + \frac{1}{h_{\text{outside}_{\text{summer}}}} + \frac{L}{\lambda_{\text{concrete}}}} \\ &= 3.25 \text{ W m}^{-1} \text{C}^{-1} \end{aligned} \quad (2)$$

$$\begin{aligned} U_{\text{wall}_{\text{winter}}} &= \frac{1}{\frac{1}{h_{\text{inside}}} + \frac{1}{h_{\text{outside}_{\text{winter}}}} + \frac{L}{\lambda_{\text{concrete}}}} \\ &= 3.41 \text{ W m}^{-1} \text{C}^{-1} \end{aligned} \quad (3)$$

2.1.3. Window

For the window (area in red in Figure 2), the wall heat transfer coefficients are used as described in Table 1. But to calculate the heat flow, first we must consider the different areas from a window such as center of glass, edge of glass or frame (Figure 3). Depending on the area, the heat flow will change.

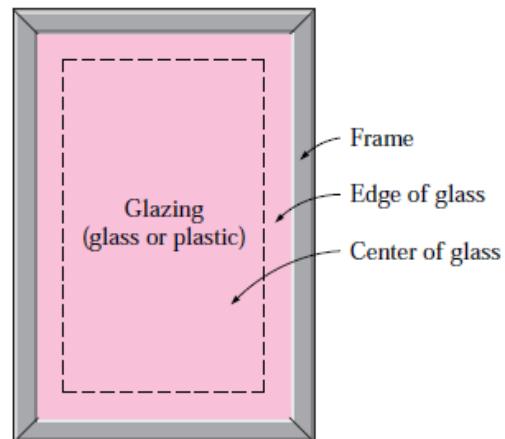


Figure 3: Window Regions

Here are the different window region areas:

$$A_{\text{window}} = 1.8 \times 2.0 = 3.6 \text{ m}^2 \quad (4)$$

$$A_{\text{glazing}} = 2 \times 1.72 \times 0.94 = 3.23 \text{ m}^2 \quad (5)$$

$$A_{\text{frame}} = A_{\text{window}} - A_{\text{glazing}} = 0.37 \text{ m}^2 \quad (6)$$

The edge of glass correspond to a 6.5 cm band around the perimeter of the glazing, so the following equations correspond to the area of the center of the glass and the edge of the glass :

$$A_{\text{center}} = 2 \times (1.72 - 0.13) \times (0.94 - 0.13) = 2.58 \text{ m}^2 \quad (7)$$

$$A_{\text{edge}} = A_{\text{glazing}} - A_{\text{center}} = 0.65 \text{ m}^2 \quad (8)$$

In this study, the window has a double glazing, so to do the calculation, the heat transfer coefficient for air spaced trapped between the two vertical parallel glass layers is used. The value $h_{\text{space}_{\text{summer}}} = 8.8 \text{ W m}^{-2} \text{ }^{\circ}\text{C}^{-1}$ and $h_{\text{space}_{\text{winter}}} = 7.2 \text{ W m}^{-2} \text{ }^{\circ}\text{C}^{-1}$ given for an air space thickness of 6 mm (10) are used in the following equations.

To calculate the heat flow for the center of window with glazing thickness : $L_{\text{window}} = 0.003 \text{ m}$ and heat transfer coefficient $\lambda_{\text{window}} = 0.92 \text{ W m}^{-1} \text{ }^{\circ}\text{C}^{-1}$, the following equations are used:

$$\begin{aligned} U_{\text{center}_{\text{summer}}} &= \frac{1}{\frac{1}{h_{\text{inside}}} + \frac{1}{h_{o_{\text{summer}}}} + \frac{1}{h_{\text{space}_{\text{summer}}}} + \frac{2L_{\text{window}}}{\lambda_{\text{window}}}} \\ &= 3.51 \text{ W m}^{-2} \text{ }^{\circ}\text{C}^{-1} \end{aligned} \quad (9)$$

$$\begin{aligned} U_{\text{center}_{\text{winter}}} &= \frac{1}{\frac{1}{h_{\text{inside}}} + \frac{1}{h_{o_{\text{winter}}}} + \frac{1}{h_{\text{space}_{\text{winter}}}} + \frac{2L_{\text{window}}}{\lambda_{\text{window}}}} \\ &= 3.38 \text{ W m}^{-2} \text{ }^{\circ}\text{C}^{-1} \end{aligned} \quad (10)$$

Then, with the heat flow of the center of window calculated, we can have access to the heat flow of the edge depending on the spacer.

Here with a metallic spacer (9):

$$U_{\text{edge}_{\text{summer}}} = 4 \text{ W m}^{-2} \text{ }^{\circ}\text{C}^{-1} \quad (11)$$

$$U_{\text{edge}_{\text{winter}}} = 3.8 \text{ W m}^{-2} \text{ }^{\circ}\text{C}^{-1} \quad (12)$$

Then for the heat flow of the frame, it depends on the frame material. Here with an aluminium frame (9):

$$U_{\text{frame}} = 10.1 \text{ W m}^{-2} \text{ }^{\circ}\text{C}^{-1} \quad (13)$$

So the total heat flow for winter and summer for the window are :

$$\begin{aligned} U_{\text{window}_{\text{summer}}} &= \frac{1}{A_{\text{window}}} (U_{\text{center}_{\text{summer}}} A_{\text{center}} + \\ &U_{\text{edge}_{\text{summer}}} A_{\text{edge}} + U_{\text{frame}} A_{\text{frame}}) = 4.28 \text{ W m}^{-2} \text{ }^{\circ}\text{C}^{-1} \end{aligned} \quad (14)$$

$$\begin{aligned} U_{\text{window}_{\text{winter}}} &= \frac{1}{A_{\text{window}}} (U_{\text{center}_{\text{winter}}} A_{\text{center}} + \\ &U_{\text{edge}_{\text{winter}}} A_{\text{edge}} + U_{\text{frame}} A_{\text{frame}}) = 4.15 \text{ W m}^{-2} \text{ }^{\circ}\text{C}^{-1} \end{aligned} \quad (15)$$

2.2. Global Emissivity Approximation

Emissivity is a measure of an object's ability to emit infrared radiation compared to a perfect blackbody. It ranges from 0 to 1, where 1 indicates perfect emission. High emissivity means the object is efficient at radiating energy as heat.

To calculate the sol air temperature, it is needed to have access to the emissivity of the wall and the window (subsection 3.2). However, if we consider two different SATs, one for the window and one for the wall, it will add one input variable for the NN. It implies that for the database establishment (section 3) it will multiply by at least 5 the number of CFD simulations needed for the database, which takes too much time.

So the idea is to approximate only one global emissivity for the window and the wall to have only one SAT. First the emissivity of the wall and glass are :

$$\epsilon_{\text{glass}} = 0.84 \quad (16)$$

$$\epsilon_{\text{wall}} = 0.9 \quad (17)$$

However, here the window is a double glazing, so we need to use the formula of effective emissivity of two parallel plate of same emissivity :

$$\epsilon_{\text{window}} = \frac{1}{\frac{2}{\epsilon_{\text{glass}}} - 1} = 0.72 \quad (18)$$

So, the approximation is the following one :

$$\begin{aligned} \epsilon_{\text{global}} &= \frac{1}{A_{\text{wall}} + A_{\text{window}}} (\epsilon_{\text{wall}} A_{\text{wall}} + \epsilon_{\text{window}} A_{\text{window}}) \\ &= 0.77 \end{aligned} \quad (19)$$

2.3. Air Conditioner

2.3.1. AC modeling

An air conditioner cools indoor spaces using a refrigeration cycle in summer. Warm air from the room is drawn in and passed over evaporator coils containing refrigerant, which absorbs heat and changes to a gas, cooling the air. The cooled air is then blown back into the room. Moreover, an AC can also heat indoor spaces.

Here to do the modeling of the air conditioner, 1 inlet and 1 outlet (Figure 4) are used and the airflow is fixed to $8 \text{ m}^3/\text{min}$. The AC is 0.8 meters long and is always on the same wall. The airflow from my air conditioner is angled downward at 45° relative to the horizontal..

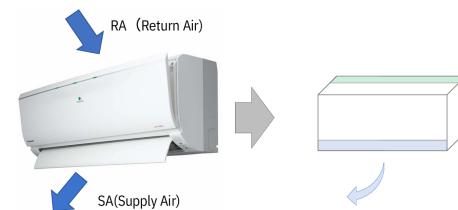


Figure 4: AC modeling

2.3.2. When to use AC

The initial point to understand is the timing of the AC usage indoors. Someone will not use AC when the outdoor temperature is a comfortable temperature.

The study focuses on the evening ([section 1](#)). So we must know when people use AC in the evening in Japan.

The paper ([11](#)) explains that in Japan during the evening, 50% of people use AC when outdoor temperature is superior to 27°C in Summer, and when outdoor temperature is inferior to 8°C in Winter. So these values are the conditions set in the study to have the AC ON.

2.4. PMV Index

The PMV (Predicted Mean Vote) index measures the thermal comfort of indoor environments. It predicts the average vote of a large group of people on a thermal sensation scale ranging from -3 (cold) to +3 (hot) ([Figure 5](#)). PMV considers factors like air temperature, humidity, air velocity, clothing insulation, and metabolic rate.

Usually, we consider it as comfortable when the PMV index ranges from -0.5 to 0.5.



Figure 5: PMV

2.4.1. Clothing Insulation

Clothing insulation refers to the ability of clothing to resist heat transfer between the body and the environment. It determines how well clothing traps heat close to the body, influencing thermal comfort. High clothing insulation value means that you have a lot of clothes and low clothing insulation value means less clothes. Clothing insulation is a parameters that impact the PMV index. Depending on the season, the clothing insulation will change.

Here, the clothes used in winter and summer are the following one from a database of clothing insulation values ([12](#)).

The winter clothing insulation is 0.84 clo and the summer clothing insulation is 0.39 clo corresponding to the clothes in the following figures ([Figure 6](#) and [Figure 7](#)).

16				
Boxer shorts	88 g	0.45 mm	CO	
Socks	44 g	0.93 mm	CO	
T-shirt	158 g	0.64 mm	CO	
Jeans (straight fit)	608 g	0.91 mm	CO, ELA	
Thick hooded jacket ("Hoodie")	830 g	4.65 mm	CO, PES	
Shoes	812 g	—	LTH, RB	
T-shirt tucked into jeans. Hood draped on back.				

Figure 6: Winter Clothes

23				
Boxer shorts	88 g	0.45 mm	CO	
Socks	44 g	0.93 mm	CO	
T-shirt	158 g	0.64 mm	CO	
Chino pants	422 g	0.70 mm	CO, ELA	
Trainers/ sneakers	424 g	—	PES, RB	

T-shirt not tucked in (over) chino pants.

Figure 7: Summer Clothes

2.4.2. Metabolic Rate

Metabolic rate is how fast your body uses energy from food to keep functioning. It depends on the activity that you are doing at the moment and it influences the PMV index. In this case, from the hypothesis ([section 1](#)), it is a man who is studying so it correspond to 1.2 met in [Table 2](#) which comes from ISO 7730 ([13](#)).

Activity	Metabolic rates (met)
Reclining	0.8
Seated, relaxed	1.0
Sedentary activity (Office, dwelling, school, laboratory)	1.2
Standing, light activity (shopping, laboratory, light industry)	1.6
Standing, medium activity (shop assistant, domestic work, machine work)	2.0

Table 2
Metabolic Rates

2.4.3. PMV Calculation

The PMV can be calculated using ISO - 7730 ([13](#)), and the main equation is the following one.

$$PMV = (0,303e^{-2.1M} + 0.028) \times ((M - W) - H - E_c - C_{res} - E_{res}) \quad (20)$$

Where :

M : metabolic rate

W : effective mechanical power

H : sensitive heat losses

E_c : heat exchange by evaporation on the skin

C_{res} : heat exchange by convection in breathing

E_{res} : evaporative heat exchange in breathing

And at the end, the calculation of the PMV index can be done only knowing 7 variables : metabolic rate, clothing

insulation, rate of mechanical work, ambient air temperature, mean radiant temperature, relative air velocity and relative humidity.

In this study, CFD software with metabolic rate and clothing insulation as inputs can directly calculate the PMV field inside the room. [Figure 8](#) is an example of a CFD simulation done using Flowdesigner.

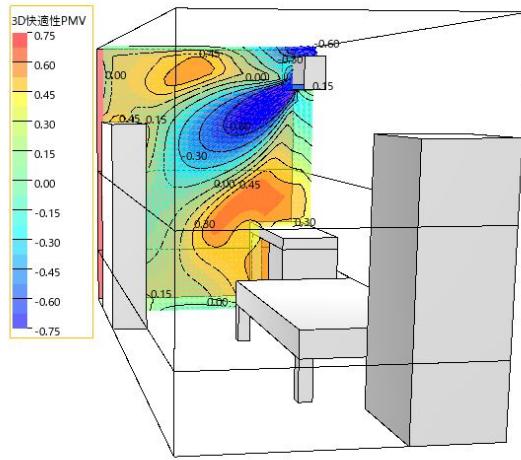


Figure 8: Example of CFD simulation

2.5. Simulation Setup

To conduct the CFD simulations, Flowdesigner software was used. The turbulence is modeled using the standard $k - \epsilon$ turbulent model calculated with the software. All simulations were done in steady state. [Table 3](#) shows the mesh type used. Export blocks correspond to the PMV values that are exported after the simulation is done. There are the PMV value for Z between 0m and 1.2m for the air (volume in red in [Figure 2](#)). This range is relevant as it pertains to a seated individual ([section 1](#)), focusing on PMV values within this height interval.

Mesh	
Type	Hexahedron
Size	$0.05m \times 0.05m \times 0.05m$
Count	200 000
Export blocks	83 184

Table 3
Mesh settings

3. Database Establishment

In the following [section 3](#) and [4](#), the study focuses on summer since it's the same methodology for winter.

Now that the CFD modeling is done and that we know how to run a simulation, the goal is to find the best position of AC during the year that guarantee a indoor comfort and minimize the energy consumption. So for each simulation the input variables are :

- Y_{AC} : Position of the center of AC on the wall ([subsection 3.1](#))
 - SAT : Sol Air Temperature ([subsection 3.2](#))
 - Q_{AC} : AC Power input ([subsection 3.3](#))
- And the export data from the CFD simulations is :
- PMV field for Z between 0m and 1.2m (volume in red in [Figure 2](#))

3.1. Position

The AC position is always on the same wall, and it only moves on the Y-axis. Since the AC is 0.8 meters long, Y_{AC} which corresponds to the position of the center of the AC is between 0.4m and 4.6m.

Here for the database establishment the values chosen for Y_{AC} are 1m, 2m, 3m and 4m.

3.2. Sol Air Temperature

The sol-air temperature (SAT) is the hypothetical outside air temperature that, in the absence of solar radiation, would produce the same temperature distribution and heat transfer rate through a wall (or roof) as observed under the combined influence of actual outdoor temperatures and incident solar radiation.

So the SAT depends on the outdoor temperature and the radiation, here from the hypothesis ([section 1](#)), we consider only evening, so there is no solar radiation.

The SAT is calculated with the following formula :

$$SAT = T_{out} - \frac{\epsilon_{global} \phi_a R}{h_{outside}} \quad (21)$$

Where:

T_{out} : outdoor temperature

ϵ_{global} : global emissivity ([subsection 2.2](#))

ϕ_a : form factor of the sky as seen from glass window (here 0.5)

R : effective radiation to the sky from horizontal surface

SAT depends on $h_{outside}$, so depending on summer or winter, SAT will be different ([Table 1](#)). Outdoor temperature and effective radiation of Sendai can be found in weather data and are hourly data.

Now that the calculation of SAT can be done, I take all the SAT when we are in the evening and when the outdoor temperature is superior to 27 °C. And [Figure 9](#) shows the SAT distribution that respect the two conditions.

For the database establishment, the value chosen are :

$SAT_1 = 26.41^\circ\text{C}$ (minimum), $SAT_2 = 26.81^\circ\text{C}$ (1st quartile), $SAT_3 = 27.75^\circ\text{C}$ (3rd quartile), $SAT_4 = 30.33^\circ\text{C}$ (maximum).

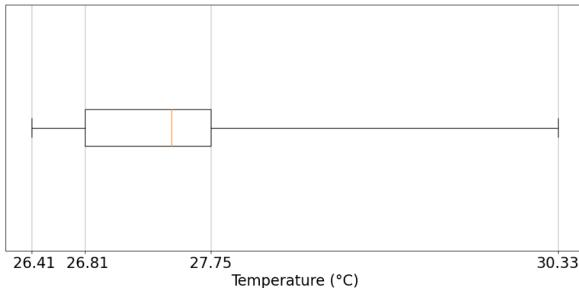


Figure 9: SAT distribution

3.3. AC Power

AC power is the power used by the AC to cool down the room during summer, and to heat up the room during winter. AC power depends on the SAT and the inside temperature (T_{in}) that the resident wants. It can be calculated in steady state with the following equation :

$$Q_{AC} = (A_{wall}U_{wall} + A_{window}U_{window}) \times (SAT - T_{in}) \quad (22)$$

So to have a good PMV field inside the room to establish the database, we must have an idea of the comfortable inside temperature, the paper (14) explains that in Summer the comfortable temperature is around 26 °C.

So here T_{in} chosen are:

$T_{in_1} = 25$ °C, $T_{in_2} = 26$ °C, $T_{in_3} = 26.5$ °C, and $T_{in_4} = 27.5$ °C.

Thus, Q_{AC} can be calculated depending on SAT and T_{in} and if the result is inferior to 0, AC power is set to 0.1. Table 4 shows the different AC powers used.

	T_{in_4}	T_{in_3}	T_{in_2}	T_{in_1}
SAT_1		0.1	8.17	28.13
SAT_2	0.1	6.14	16.12	36.08
SAT_3	4.90	24.86	34.83	54.79
SAT_4	56.58	76.53	86.51	106.47

Table 4
 Q_{AC} depending on SAT and T_{in}

Finally, 60 simulations are done to establish the summer database.

4. Neural Network

A neural network is a computational model inspired by the way biological neural networks in the human brain process information.

NN are versatile and used in various fields, including image and speech recognition, natural language processing (NLP), medical diagnosis, autonomous vehicles, and more.

It consists of interconnected nodes, or neurons, organized in layers. Figure 10 shows the architecture used in this study.

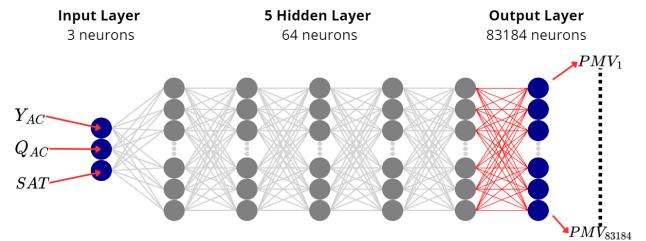


Figure 10: NN architecture

Each neuron corresponds to a function called activation function that receives the weighted outputs of other neurons from the previous layer and a bias. The following equation corresponds to the role of a neuron :

$$y = f(\sum_i x_i w_i + b) \quad (23)$$

Where :

x_i : output of the i^{th} neuron from the previous layer

w_i : weight that connect with the i^{th} neuron from the previous layer

y : output of this neuron

f : activation function

During the training of the NN, at each epoch of the training the NN will return a score/loss value that is calculated with a score/loss function. For regression problems, mean squared error (MSE) is generally used as the loss function.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (24)$$

\hat{y}_i : i^{th} predicted value by NN

y_i : corresponding true value by CFD

So, the goal of training is to find the minimum of the loss function by modifying weights and biases values. In most cases, Gradient Descent algorithm is used to find the minimum.

This section purpose is to find the best model possible. The NN calculation is performed on Google Colab using a VM. The programming language is Python with many libraries such as numpy (15), pandas (16) and tensorflow (17).

4.1. Database Splitting

To train a NN and to see the accuracy of the model, the first thing to do is to split the database between the training, testing and validation dataset. Figure 11 shows the splitting chosen.

To validate the chosen dataset and the input variables chosen to establish the dataset, Figure 12 shows the mean of the PMV field for each CFD simulation. This value is around 0.5 each time which is the value that is interesting.

$Y_{ac} = 1m$

SAT/Tin	27.5	26.5	26	25
26.41		0	1	2
26.81	3	4	5	6
27.75	7	8	9	10
30.33	11	12	13	14

 $Y_{ac} = 2m$

SAT/Tin	27.5	26.5	26	25
26.41		15	16	17
26.81	18	19	20	21
27.75	22	23	24	25
30.33	26	27	28	29

 $Y_{ac} = 3m$

SAT/Tin	27.5	26.5	26	25
26.41		30	31	32
26.81	33	34	35	36
27.75	37	38	39	40
30.33	41	42	43	44

 $Y_{ac} = 4m$

SAT/Tin	27.5	26.5	26	25
26.41		45	46	47
26.81	48	49	50	51
27.75	52	53	54	55
30.33	56	57	58	59

Figure 11: Data split and number

 $Y_{ac} = 1m$

SAT/Tin	27.5	26.5	26	25
26.41		0,86	0,81	0,31
26.81	0,88	0,86	0,78	0,12
27.75	0,92	0,65	0,55	-0,07
30.33	0,67	0,2	-0,11	-0,51

 $Y_{ac} = 2m$

SAT/Tin	27.5	26.5	26	25
26.41		0,85	0,81	0,40
26.81	0,86	0,84	0,76	0,20
27.75	0,88	0,76	0,57	0,19
30.33	0,72	0,30	0,07	-0,38

 $Y_{ac} = 3m$

SAT/Tin	27.5	26.5	26	25
26.41		0,85	0,82	0,38
26.81	0,86	0,85	0,70	0,36
27.75	0,87	0,75	0,47	0,16
30.33	0,66	0,32	0,08	-0,45

 $Y_{ac} = 4m$

SAT/Tin	27.5	26.5	26	25
26.41		0,84	0,79	0,39
26.81	0,91	0,87	0,75	0,33
27.75	0,92	0,82	0,64	0,21
30.33	0,73	0,42	0,13	-0,12



Figure 12: Mean of PMV field for all CFD simulations

4.2. Hyperparameters Optimization

Hyperparameter optimization for NN is crucial for achieving optimal model performance. It involves tuning parameters such as learning rate, optimizers, and number of neurons to enhance training efficiency and generalization. Techniques like grid search, random search, and Bayesian

optimization are commonly employed to systematically explore the hyperparameter space and find the configuration that yields the best results.

In this study, hyperopt (a python library) was used to find the best hyperparameters (18), it used a Bayesian optimization, which is more performant than RandomSearch or

GridSearch. [Table 5](#) shows the hyperparameter space used to find the best hyperparameters. The parameter 'neurons' corresponds to the number of neurons on each hidden layer.

Parameter	Values
neurons	16, 32, 64, 128
optimizers	Adam, RMSprop, SGD
nb_hidden	5, 10, 20
epochs	500
learning_rate	Uniform(0.001, 0.01)
activation	relu, sigmoid, tanh

Table 5
Hyperparameter Space for Optimization

200 models were trained and tested with only 500 epochs. Dagshub was used to display the results in [A. Table 6](#) shows the hyperparameters chosen.

After finding the best hyperparameters, the training epoch is set to 5000 to ensure the convergence of the NN.

Parameter	Values
neurons	64
optimizers	RMSprop
nb_hidden	5
epochs	5000
learning_rate	0.001
activation	relu

Table 6
Optimized Hyperparameters

4.3. Preprocessing Methods

Preprocessing methods for neural networks optimize data for effective model training. Many preprocessing methods exist such as Normalization, Standardization or Non-dimenzionalization. It can considerably improve the accuracy of a model, especially for NN with CFD simulation ([19](#)). Without data preprocessing, errors can become prominent, and the gradient descent algorithm may struggle to effectively reduce errors in variables that have smaller magnitudes during training.

4.3.1. Normalization

Normalization is a frequently employed preprocessing technique that adjusts data to fit within the [0, 1] range using the maximum and minimum values of each feature. It applies to all input data, here for example with SAT.

$$SAT_{norm} = \frac{SAT - min(SAT)}{max(SAT) - min(SAT)} \quad (25)$$

where $min(SAT)$, $max(SAT)$ are the minimum and maximum of SAT for the training cases.

4.3.2. Standardization

Standardization is a common data preprocessing technique in machine learning that centers the data by removing

the mean and scales it to have unit variance. Here for example with SAT.

$$SAT_{stan} = \frac{SAT - \mu}{\sigma} \quad (26)$$

where μ and σ are the mean and the variance of SAT for training cases.

4.3.3. Dimension Reduction

Dimension reduction techniques are essential in machine learning for simplifying datasets while retaining important information. Here the output variable is all the PMV field that correspond to 83 184 PMV values, so dimension reduction consist to divide the number of output variable. To do so, I divided the number of exported PMV values by 8 by taking the mean of eight adjacent blocks of PMV. In fact, it's the same as doing a different meshing 2 times bigger. So now, the PMV field that we want to predict has only 10479 values, which allows a faster computation.

4.4. Result Analysis

This section aims to compare the result of the different preprocessing methods and find the best possible model.

4.4.1. Error Quantification

To quantitatively assess the prediction accuracy of the neural network (NN), the following error is established which is also used in other papers ([19](#)).

$$Err_i = \left| \frac{PMV_i^{CFD} - PMV_i^{NN}}{\max(PMV_i^{CFD}) - \min(PMV_i^{CFD})} \right| \quad (27)$$

Where PMV_i^{CFD} is the true value from the CFD simulation and PMV_i^{NN} is the value predicted by NN, and $\max(PMV_i^{CFD})$ and $\min(PMV_i^{CFD})$ are the maximum and minimum of the PMV true value from this case.

This error is different than the usual relative error, because here PMV values are around 0, so with the usual relative error it will create unreasonable error.

4.4.2. Test on Training Dataset

[Figure 13](#) demonstrates the NN performance of PMV prediction on all the training dataset. It shows that only with dimension reduction the error for a simulation can be more than 5%. NN with normalization and standardization have the lowest error. [Table 7](#) shows the mean error for every case of the training dataset. Normalization and standardization divide by 2 the error compared to the basic case. And dimension reduction multiplies by 2 the error compared to the basic case.

4.4.3. Test on Testing Dataset

[Figure 14](#) demonstrates the NN performance of PMV prediction on all the testing dataset. It shows that the basic case and dimension reduction have almost the same error and normalization and standardization also have almost the

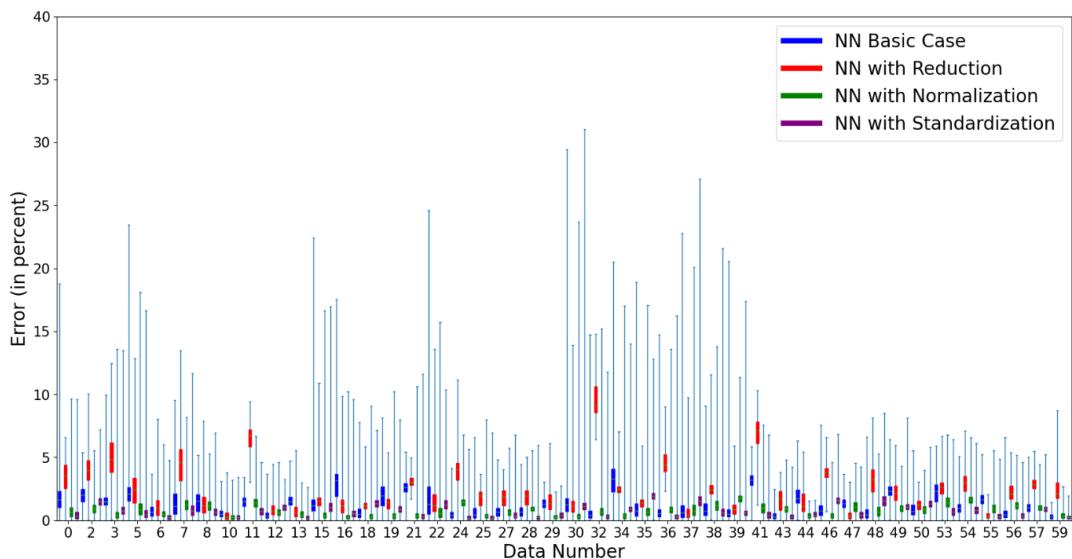


Figure 13: Comparison of error for the training dataset

Data Number	NN Basic Case	Reduction	Normalization	Standardization
0	1.944729	3.402949	0.751185	0.489582
2	1.992430	3.989606	0.929528	1.507180
3	1.541282	5.094008	0.483317	0.805463
5	2.202216	2.645071	0.956695	0.620827
6	0.758799	1.225766	0.535096	0.307003
...
55	1.700891	0.425259	0.922170	0.349086
56	0.525034	2.233801	1.193096	0.398604
57	0.993658	2.873039	1.004544	0.917018
59	0.307756	2.495006	0.378213	0.240268
Mean	1.313959	2.490473	0.793327	0.745977

Table 7

Mean error for the training dataset

same error. NN with normalization and standardization have the lowest error. Indeed, with preprocessing methods the maximum mean error is around 15%. Whereas for the basic case and dimension reduction case, the maximum mean error can go up to 50%. Table 8 shows the mean error for every case of the testing dataset. Normalization and standardization have the mean error around 6%, whereas the basic case and dimension reduction have the mean error around 14%. Here the best case is with normalization and is the model chosen in the following sections.

4.4.4. Feature importance

Feature importance in neural networks can be determined by calculating the sum of the absolute weights connecting input features to neurons in the first hidden layer. Higher values indicate greater influence of those features on the network's initial processing and decision-making. This method helps identify which input features contribute more significantly to the model's predictions, aiding in feature

selection and understanding model behavior. It's crucial for interpreting neural network operations and optimizing feature engineering efforts.

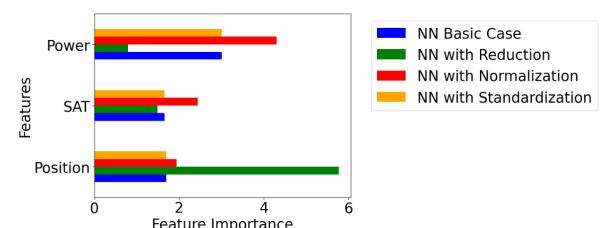
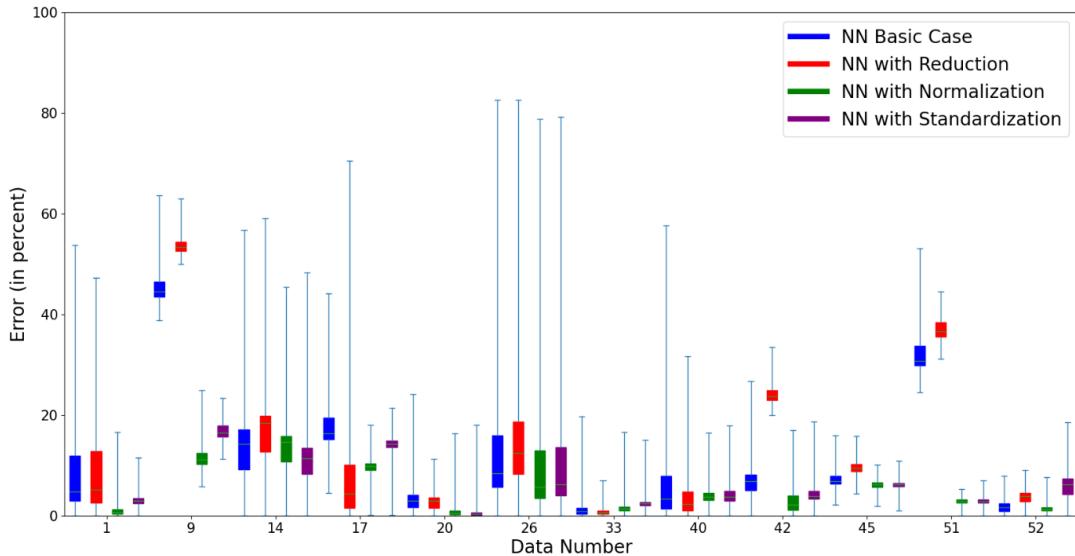


Figure 15: Feature importance

Figure 15 shows the feature importance depending of the different case. It demonstrates that for the worst case which is NN with reduction the main feature is the AC position,

**Figure 14:** Comparison of error for the testing dataset

Data Number	NN Basic Case	Reduction	Normalization	Standardization
1	8.139474	9.048435	0.973443	3.026753
9	45.553793	53.700150	11.691819	16.889560
14	13.791778	16.913194	13.625349	11.434766
17	17.692809	8.137301	9.674165	14.228794
20	3.439434	2.732197	0.713181	0.437385
26	12.865245	15.449526	10.615810	11.013294
33	1.204223	0.779730	1.366375	2.393399
40	6.222906	3.881778	3.817003	4.160705
42	6.863333	24.147201	2.914599	4.293493
45	7.227953	9.490665	6.152362	6.171570
51	32.309255	37.029692	2.829195	2.877835
52	1.660559	3.602963	1.347347	5.912622
Mean	13.080897	15.409403	5.476721	6.903348

Table 8

Mean error for the testing dataset

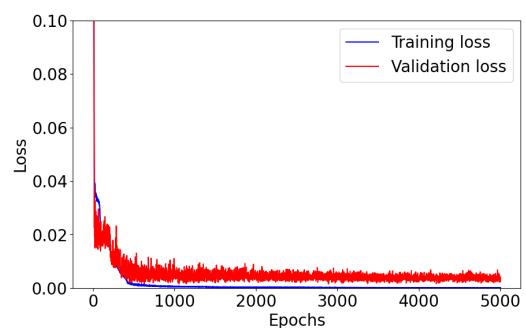
whereas for the best case which is NN with normalization the main feature is the AC power. So it means that PMV distribution depends more on the AC power than on the AC position.

4.5. Best model results

This section focuses on the results of NN with normalization.

Figure 16 compares the PMV field from the CFD and result from the NN on some cases of the training dataset at $Z = 1\text{m}$. And Figure 17 compares the PMV field from the CFD and result from the NN on some cases of the testing dataset at $Z = 1\text{m}$. It shows that overall PMV distribution is about the same between CFD and result from NN for training dataset. However, there are still some small differences for testing dataset, especially for data number 14.

Moreover, Figure 18 shows the training and validation loss for NN with normalization. It demonstrates that there is no overfitting, the validation loss never goes up.

**Figure 18:** Training and validation loss

AC placement optimization with CFD and NN

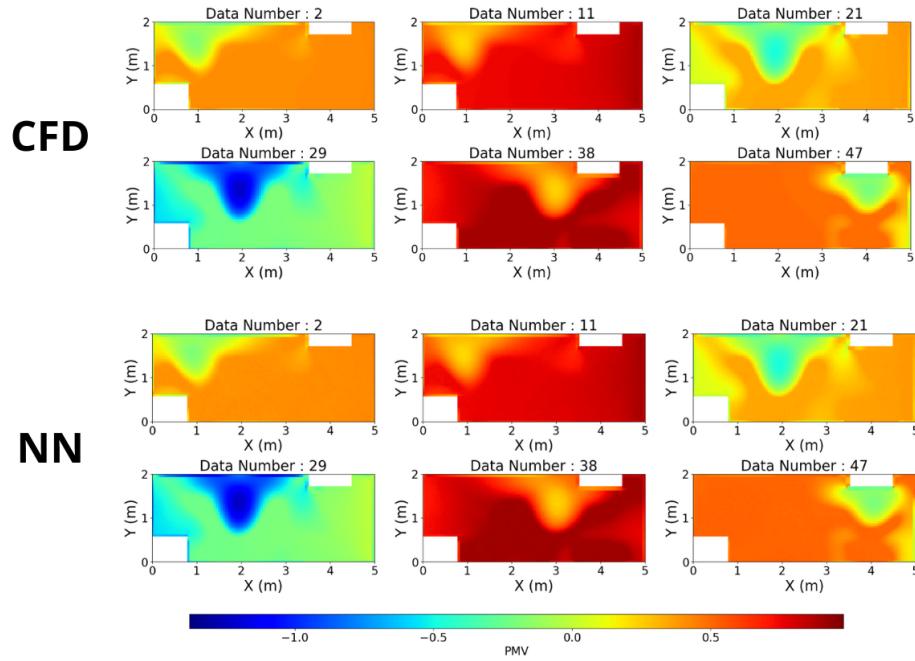


Figure 16: PMV field at $Z = 1\text{m}$ with CFD and NN for training dataset

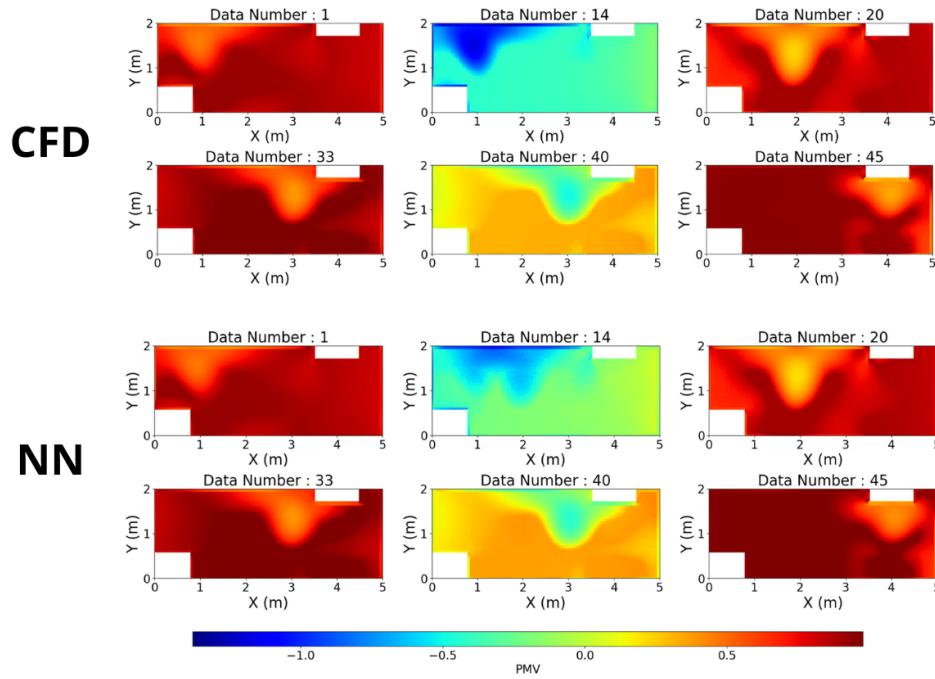


Figure 17: PMV field at $Z = 1\text{m}$ with CFD and NN for testing dataset

5. AC Position Optimization

5.1. PMV Distribution

Now that we've identified the top-performing NN, the next step is to envision the outcome and determine the most

suitable methodology to ensure consistent results.

This subsection focuses on the PMV distribution from the summer CFD simulations database to understand the

results and to define the methodology and the criteria used to find the best AC position.

Figure 19 is an example that shows that the PMV distribution vary significantly across different AC position. At $Y_{AC} = 1m$, the PMV values are widely spread with a peak around 0, indicating cooler conditions inside the bedroom. As the AC position increases to 2 and 3 meters, the distributions become narrower with peaks around 0.2, showing a trend towards warmer conditions. At 4 meters, the distribution is the narrowest with a sharp peak at 0.25, indicating the warmest and most consistent conditions. Overall, the data suggests that higher AC positions have less cooling effect than low AC positions but they have more stable thermal environments.

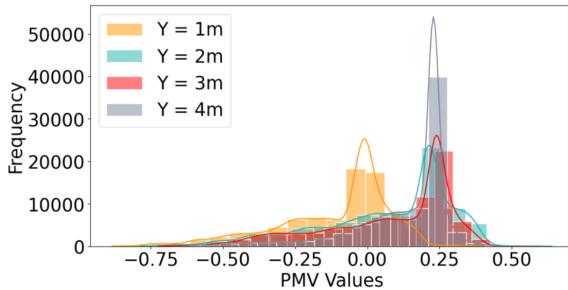


Figure 19: PMV distribution for the case $Q_{AC} = 86.51 \text{ W}$ depending on the AC position

Figure 20 shows that the standard deviation of PMV field depends on the AC power and the AC position. Indeed, higher AC power means that there are more temperature difference between inside and outside the bedroom. This figure demonstrates that at 4 meter the standard deviation is lower than other case, which is consistent with **Figure 19**.

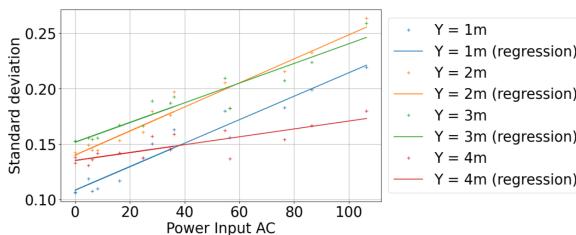


Figure 20: Standard deviation depending on the AC position

5.2. Methodology

The findings in the preceding section indicate that when evaluating whether an AC placement is effective, it is crucial to assess not only its cooling capacity but also the stability of the thermal conditions it provides. To do so, we define the PMV deviation in the following equation :

$$\Delta PMV_i = |PMV_i| - PMV_{range} \quad (28)$$

where $\Delta PMV_i = 0$ if $\Delta PMV_i < 0$.

Then, we can define the sum of PMV deviation for a PMV field with the following equation:

$$\sum_{i=1}^n \Delta PMV_i \quad (29)$$

where n is the number of PMV blocks in the PMV field.

Now that this criteria is defined, the methodology is the following one and **Figure 21** explained it more clearly :

1. For each hour during the year, we have the radiation and the outdoor temperature. When the outdoor temperature is superior to 27°C or inferior to 8°C ([subsubsection 2.3.2](#)) in the evening, it means that the AC is on.
2. With T_{out} and the radiation, we calculate the SAT ([subsection 3.2](#)).
3. For each AC position, we try multiple AC power and by using NN ([section 4](#)) we obtain in each case the PMV field.
4. For each PMV field, we calculate the sum of PMV deviations ([subsection 5.2](#)).
5. At the end, for each AC position, we keep the power that minimizes the sum of PMV deviation. Moreover, if there are multiple powers that minimizes it, we keep the minimum power.

5.3. Results analysis

This section focuses on summer and shows only the final results for winter.

5.3.1. Summer

Table 9 shows the result that correspond to the hourly sum of PMV deviation for summer with $PMV_{range} = 0.5$. Index date corresponds to an hour during the year.

Table 10 shows the hourly power consumption for summer that corresponds.

At the end, we can analyze the summer total power consumption and the summer total sum of PMV deviations to find the best position.

Figure 22 and **Figure 23** are results with $PMV_{range} = 0.5$.

Figure 22 shows the summer sum of PMV deviations depending on Y_{AC} and **Figure 23** shows the corresponding summer power consumption. In my opinion the best AC position for summer is for $Y_{AC} = 1m$. Indeed, at this position, it is the minimum power consumption that get a sum of PMV deviation that close to 0.

Appendix B shows the results in summer with $PMV_{range} = 0.2$ and $PMV_{range} = 0.8$. **Figure 33** and **Figure 34** show that to have a good confort with $PMV_{range} = 0.2$ the best position is $Y_{AC} = 3.4m$ even though the power consumption is high.

Moreover, **Figure 35** and **Figure 36** show that since the range is higher, the condition to find the best position is only

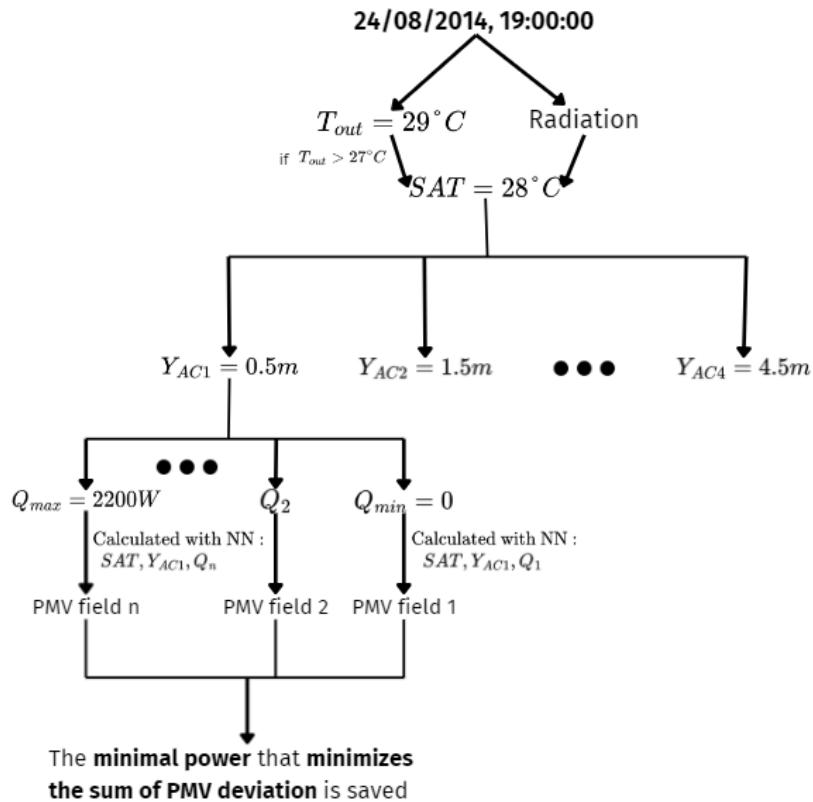


Figure 21: Methodology

Index Date	Yac = 0.4	Yac = 0.6	Yac = 0.8	...	Yac = 4.4	Yac = 4.6
4702	66.739	15.795	1.180	...	10.060	33.479
4745	116.730	94.233	17.630	...	19.838	29.031
4746	117.022	33.539	17.583	...	12.684	20.941
:	:	:	:	:	:	:
5825	73.340	20.480	1.734	...	7.721	27.134
Total	3037.360	1307.353	424.897	...	264.628	772.284

Table 9

Hourly sum of PMV deviations

Index Date	Yac = 0.4	Yac = 0.6	Yac = 0.8	...	Yac = 4.4	Yac = 4.6
4702	30.100	30.100	30.100	...	45.100	50.100
4745	65.100	65.100	70.100	...	80.100	80.100
4746	55.100	55.100	55.100	...	70.100	70.100
:	:	:	:	:	:	:
5825	35.100	30.100	30.100	...	45.100	50.100
Total	1233.100	1198.100	1198.100	...	1618.100	1703.100

Table 10

Hourly power consumption

the minimum of the summer power consumption, which is $Y_{AC} = 1m$.

So depending on PMV_{range} , the best position is different, if we set a low PMV_{range} the best position is close to the window, and if we set a high PMV_{range} the best position is far from the window.

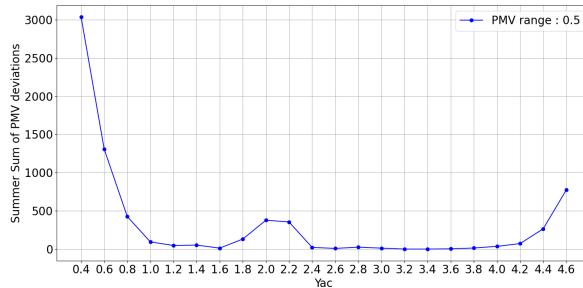


Figure 22: Summer Sum of PMV deviations depending on Y_{AC} with $PMV_{range} = 0.5$

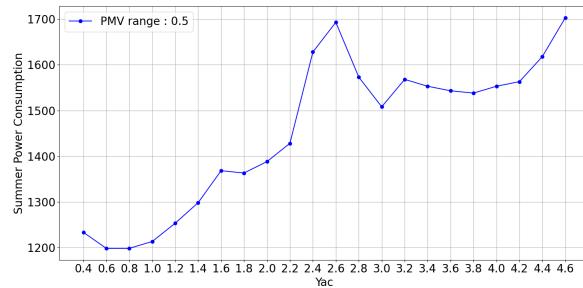


Figure 23: Summer Power Consumption depending on Y_{AC} with $PMV_{range} = 0.5$

5.3.2. Winter

Figure 24 and Figure 25 show that in winter the best position is $Y_{AC} = 3.8m$. Contrary to summer, in winter sum of PMV deviation and power consumption have the same minimum. The interpretation is in subsection 5.4.

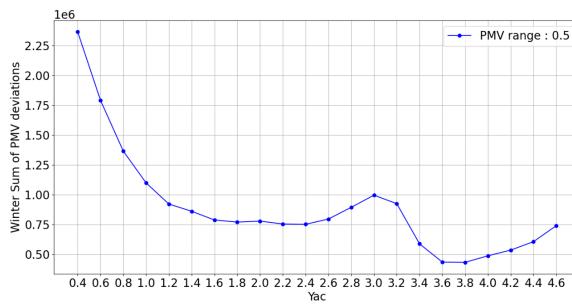


Figure 24: Winter Sum of PMV deviations depending on Y_{AC} with $PMV_{range} = 0.5$

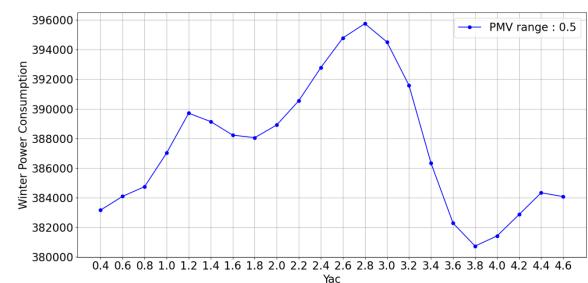


Figure 25: Winter Power Consumption depending on Y_{AC} with $PMV_{range} = 0.5$

5.3.3. Annual

For the annual result, the previous subsections show that the power consumption and the sum of PMV deviation in summer is negligible compared to winter.

Indeed, the differential of temperature between SAT and the comfortable temperature is higher in winter than in summer. Thus, the power consumption is higher and the sum of PMV deviation too. Moreover, in summer there are only 31 hours that satisfy the condition (evening and $T_{out} > 27^{\circ}\text{C}$), whereas in winter there are 944 hours that satisfy the condition (evening and $T_{out} < 8^{\circ}\text{C}$).

So, the best AC position with this methodology is $Y_{AC} = 3.8m$.

Moreover, the difference in annual power consumption between the AC position that minimize and maximize power consumption is around 5%.

5.4. Interpretation

The aim of this section is to grasp the physical essence of the obtained result. Figure 27 shows the airflow with the AC close to the window and Figure 26 shows the airflow with the AC far from the window in summer.

The idea is when the AC is close to the window there is a higher differential of temperature at the window between inside and outside. So heat gain is higher, thus it will need more AC power to cool down the room. Moreover, Figure 26

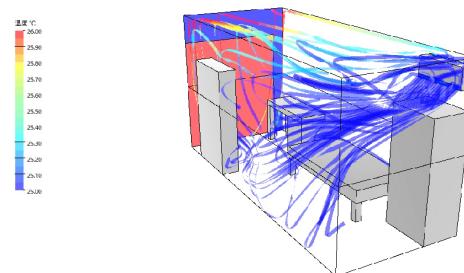


Figure 26: Temperature and airflow inside the bedroom for $Y_{AC} = 0.4m$

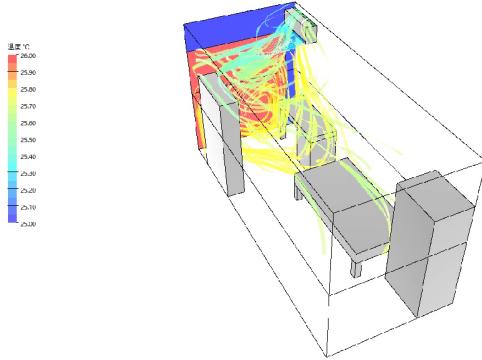


Figure 27: Temperature and airflow inside the bedroom for $Y_{AC} = 4.6m$

shows that the airflow stays in the same area compared to [Figure 27](#).

In summer, we can interpret the PMV distribution described in [subsection 5.1](#). When the AC unit is positioned near the window, the temperature differential inside the room is minimized, resulting in a lower standard deviation. However, the temperature differential between inside and outside near the window is greater, leading to a less effective cooling effect compared to when the AC unit is positioned farther away from the window.

In winter, the findings presented in [subsubsection 5.3.2](#) can be interpreted by the fact that improved airflow distribution in the room lower the PMV within the bedroom. However, during winter, the AC main objective is to warm the room. Therefore, when the AC is positioned to enhance airflow distribution, it necessitates greater power consumption to achieve efficient heating, which is the case when the AC is far from the window ([Figure 27](#)).

Therefore, despite the greater temperature differential when the AC is positioned near the window, this airflow effect suggests that the minimum power consumption occurs when the AC is close to the window. Furthermore, this position corresponds to the lowest sum of PMV deviations, which remains consistent between winter and summer conditions.

5.5. Calculation Time

This subsection demonstrates the time saved by using NN by counting the number of CFD simulations necessary to do the same study without NN. A CFD simulation takes approximately 10 minutes (t_{CFD}) to run and to export the data.

In summer, there are 31 hours that satisfy the condition (evening and $T_{out} > 27^{\circ}\text{C}$). For each hour we try 21 different AC positions, and for each AC position we try 30 AC power. So in summer, the number of CFD simulation done is :

$$N_{summer} = 31 \times 21 \times 30 = 19530 \quad (30)$$

In winter, there are 944 hours that satisfy the condition (evening and $T_{out} < 8^{\circ}\text{C}$). For each hour we also try 21 different AC positions, and for each AC position we also try 30 AC power. So in winter, the number of CFD simulation done is :

$$N_{winter} = 944 \times 21 \times 30 = 594720 \quad (31)$$

In total, the total number of CFD simulation done is :

$$N_{tot} = N_{winter} + N_{summer} = 614250 \quad (32)$$

So the necessary time to do this study without NN is :

$$t_{CFDs} = \frac{N_{tot} \times t_{CFD}}{60} = 102375 \text{ hours} \quad (33)$$

Therefore, without employing NN, it would have required 102,375 hours, equivalent to 4,265.625 days, or approximately 11.686 years.

In this study, with NN, 124 CFD simulations (N_{NN}) were done in total for winter and summer to create the database. Then, to train the two NN it took around 1 hour (t_{train}). And the time to do all the simulation with NN was around 30 minutes ($t_{NN_{simul}}$). So, the time to do this study with NN is :

$$t_{NN} = \frac{N_{NN} \times t_{CFD}}{60} + t_{train} + t_{NN_{simul}} = 22 \text{ hours} \quad (34)$$

So ultimately, the use of NN made the process **4,653** times faster.

6. Conclusion

In this study, two NN models (summer and winter) are constructed to realize fast and accurate predictions of PMV distribution inside a bedroom in Sendai to find the optimized position of the AC. Two CFD database are established to train, validate and test the NNs. The main conclusions are summarized as follows.

1. The optimal performance of NNs in both winter and summer is achieved using the data preprocessing technique of normalization. The prediction accuracy of the PMV field is approximately 5% error when normalization is applied.
2. In summer, the optimal position is away from the window ($Y_{AC} = 1.0m$), whereas in winter, it is near the window ($Y_{AC} = 3.8m$).
3. Given this methodology and hypothesis, where summer's impact is minimal compared to winter, the ideal position overall is $Y_{AC} = 3.8m$, which results in an economy of 5% of power consumption and a better comfort.
4. Utilizing NNs accelerates this methodology by a factor of **4,653** compared to relying solely on CFD simulations.

A. Hyperparameters Optimization

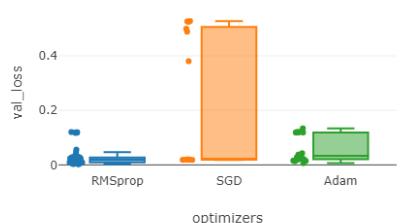


Figure 28: Optimizers

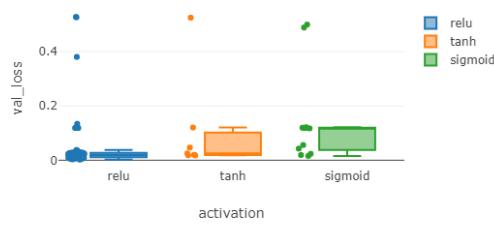
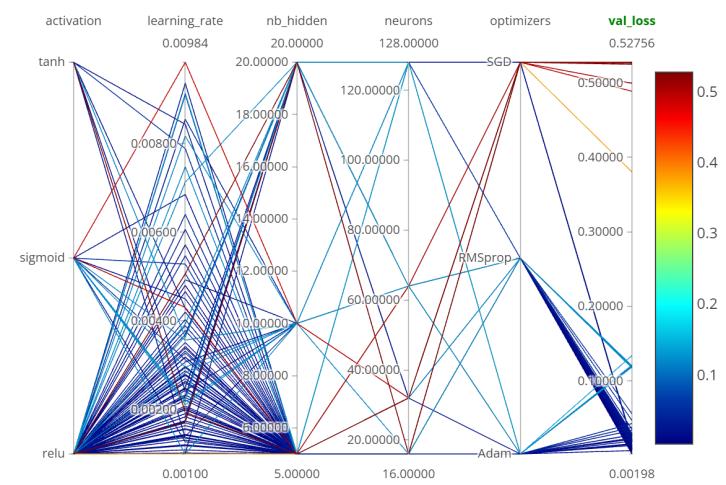


Figure 29: Activation Function

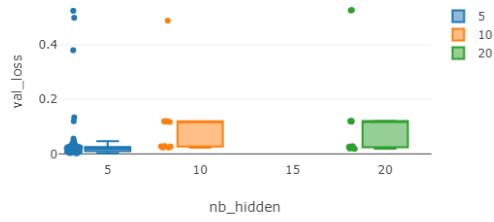


Figure 30: Number of hidden layers

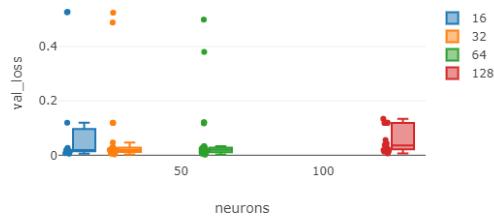


Figure 31: Number of neurons

Figure 32: Scatter Plot of hyperparameters

B. Summer results with different PMV_{range}

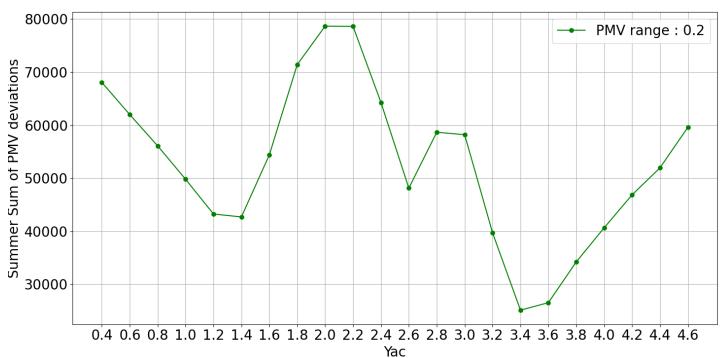


Figure 33: Summer Sum of PMV deviations depending on Y_{AC} with $PMV_{range} = 0.2$

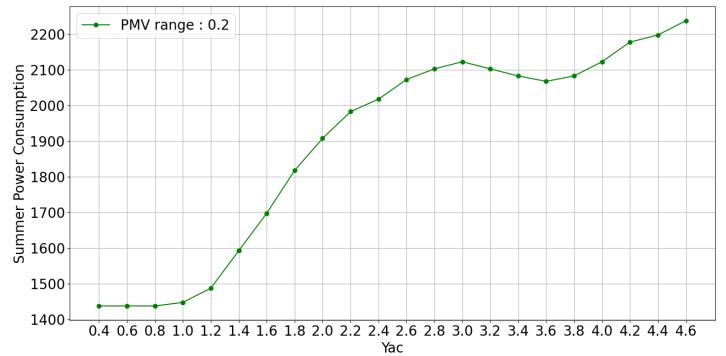


Figure 34: Summer Power Consumption depending on Y_{AC} with $PMV_{range} = 0.2$

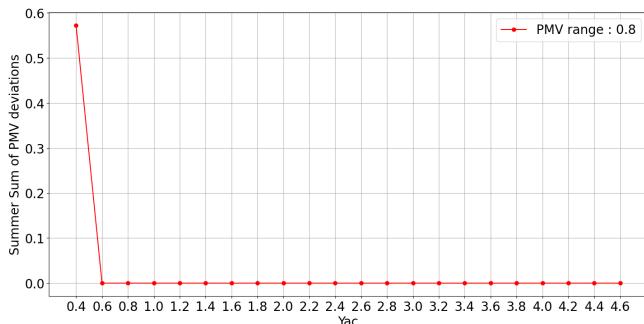


Figure 35: Summer Sum of PMV deviations depending on Y_{AC} with $PMV_{range} = 0.8$

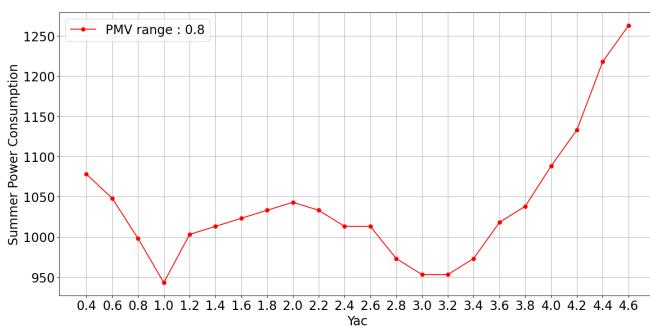


Figure 36: Summer Power Consumption depending on Y_{AC} with $PMV_{range} = 0.8$

Data and Code availability

The code implementation was done in Python on Google Colab. All the code used and the data used are available on <https://github.com/thoid/CFD-with-NN>

CRediT authorship contribution statement

Thomas IDIER: Conceptualization of this study, Methodology, Database establishment, Neural Network training.

Acknowledgement

This work was supported by the LBEE of the department of architecture and building science at Tohoku University, and especially Goto Lab.

References

- [1] K. Yoshida, H. B. Rijal, K. Bogaki, A. Mikami, H. Abe, Field study on energy-saving behaviour and patterns of air-conditioning use in a condominium, *Energies* 14 (24) (2021). doi:10.3390/en14248572.
- [2] J. Yuan, Z. Jiao, X. Xiao, K. Emura, C. Farnham, Impact of future climate change on energy consumption in residential buildings: A case study for representative cities in japan, *Energy Reports* 11 (2024) 1675–1692. doi:10.1016/j.egyr.2024.01.042.
- [3] Z. Zhang, Y. Gao, S. Nie, Y. Tian, C. Li, R. Gao, K. Yin, Y. Liu, B. Liu, H. Li, Study on the recommended placement and air distribution of split floor-standing room air conditioners, *International Journal of Architectural Engineering Technology* 9 (2022) 1–17. doi:10.15377/2409-9821.2022.09.1.
- [4] T. Yamamoto, A. Ozaki, M. Lee, Optimal air conditioner placement using a simple thermal environment analysis method for continuous large spaces with predominant advection, *Energies* 14 (15) (2021). doi:10.3390/en14154663.
- [5] W. Zuo, Q. Chen, Real-time or faster-than-real-time simulation of airflow in buildings, *Indoor air* 19 (2009) 33–44. doi:10.1111/j.1600-0668.2008.00559.x.
- [6] Q. Zhou, R. Ooka, Comparison of different deep neural network architectures for isothermal indoor airflow prediction, *Building Simulation* 13 (6) (2020) 1409–1423. doi:10.1007/s12273-020-0664-8.
- [7] Y. Zheng, X. Xu, Predicting indoor 3D airflow distribution using artificial neural networks with two different architectures, *Energy and Buildings* 303 (2024) 113841. doi:10.1016/j.enbuild.2023.113841.
- [8] Q. Zhou, R. Ooka, Implementation of a coupled simulation framework with neural network and Modelica for fast building energy simulation considering non-uniform indoor environment, *Building and Environment* 211 (2022) 108740. doi:10.1016/j.buildenv.2021.108740.
- [9] ASHRAE American Society of Heating Refrigerating and Air-Conditioning Engineers, *ASHRAE Handbook - Fundamentals*, ASHRAE, Atlanta, GA, 2017.
- [10] US Dept. of Commerce, *Building materials and structures Report 151*.
- [11] M. Schweiker, M. Shukuya, Comparison of theoretical and statistical models of air-conditioning-unit usage behaviour in a residential setting under japanese climatic conditions, *Building and Environment* 44 (2009) 2137–2149. doi:10.1016/j.buildenv.2009.03.004.
- [12] J. Smallcombe, S. Hodder, K. Kuklane, M. Mlynarczyk, D. Loveday, J. Eggeling, A. Halder, G. Havenith, Updated database of clothing thermal insulation and vapor permeability values of western ensembles for use in ashrae standard 55, iso 7730, and iso 9920 VC-21-011 (01 2021).
- [13] International Organization for Standardization, ISO 7730 2005-11-15 *Ergonomics of the Thermal Environment: Analytical Determination and Interpretation of Thermal Comfort Using Calculation of the PMV and PPD Indices and Local Thermal Comfort Criteria*, Vol. 3, 2005.
- [14] V. Draganova, H. Yokose, K. Tsuzuki, Y. Nabeshima, Field study on nationality differences in adaptive thermal comfort of university students in dormitories during summer in japan, *Atmosphere* 12 (2021) 566. doi:10.3390/atmos12050566.
- [15] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, T. E. Oliphant, Array programming with NumPy, *Nature* 585 (7825) (2020) 357–362. doi:10.1038/s41586-020-2649-2.
- [16] T. pandas development team, *pandas-dev/pandas: Pandas* (Feb. 2020). doi:10.5281/zenodo.3509134.
- [17] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng, *TensorFlow: Large-scale machine learning on heterogeneous systems*, software available from tensorflow.org (2015). URL <http://tensorflow.org/>
- [18] J. Bergstra, D. Yamins, D. Cox, Hyperopt: A python library for optimizing the hyperparameters of machine learning algorithms, 2013, pp. 13–19. doi:10.25080/Majora-8b375195-003.
- [19] Q. Zhou, R. Ooka, Influence of data preprocessing on neural network performance for reproducing cfd simulations of non-isothermal indoor airflow distribution, *Energy and Buildings* 230 (2021) 110525. doi:10.1016/j.enbuild.2020.110525.