

Tobias Kristiansen

# Predicting annual illuminance and operative temperature in residential buildings using artificial neural networks

Master's thesis in Civil and Environmental Engineering

Supervisor: Mohamed Hamdy

Co-supervisor: Barbara Szybinska Matusiak

June 2021



Tobias Kristiansen

# **Predicting annual illuminance and operative temperature in residential buildings using artificial neural networks**

Master's thesis in Civil and Environmental Engineering  
Supervisor: Mohamed Hamdy  
Co-supervisor: Barbara Szybinska Matusiak  
June 2021

Norwegian University of Science and Technology  
Faculty of Engineering  
Department of Civil and Environmental Engineering





## PREFACE

---

This master thesis is written as the final semester project of the two-year Master of Science program Civil and Environmental Engineering, at the Norwegian University of Science and Technology (NTNU) during the spring of 2021.

My interest in building physics started in my freshman year at Østfold University College, back in 2015, which led me to further explore thermodynamics, energy management, indoor comfort and daylight in buildings. The development of this thesis started in the spring semester of 2020, with the introduction to building performance simulation, in a course held by my supervisor, Professor Mohamed Hamdy. Throughout this year I learned about the complexity of energy and environmental challenges that the building sector is facing. Building performance simulation and artificial neural networks is proving to be an effective approach for supporting the design and operation of high-performance buildings.

I would like to thank my supervisor Professor Mohamed Hamdy for guidance, advice and feedback on building simulations, energy and thermal comfort modelling and scientific writing. This project could not have been done without your insight in both building physics and computer science combined. It has been an exciting journey and inspirational working together on this project. I would also like to thank Professor Barbara Szybinska Matusiak, my co-supervisor, for the invaluable input on daylight modelling and technical aspects of the project. You have taught me the value of integrated building design and visual comfort.

A big thanks to all my friend at NTNU and especially Yong Bin Kwon for valuable input and endless hours of discussions on the topic of data science and machine learning. I would like to thank Multiconsult, and especially Helene Solvang, Wolfgang Kampel and Håkon Eggebø, for investing their time and effort in my project.

I would also like to thank my supportive family for contribution in reviewing this thesis, and moral support through my years as a student. Finally, I would like to thank my fiancé Anna Margrethe Eriksen and our dog Storm for moral support and motivation. Especially while working long hours from home under the pandemic.



## ABSTRACT

---

Humans spend 87 % of their time indoors, mostly in their own residence. The indoor environment is a crucial factor for people's health and welfare. There is an increasing challenge with overheating in buildings due to hotter climate. In addition, centralization and stricter building codes has led us to build more compact, making it more challenging to achieve daylight criteria in new building projects. Building Performance Simulation (BPS) is proving to be an effective approach for supporting the design and finding a balance between daylight availability, thermal comfort and energy performance. In current practice these aspects are treated separately, which leads to increased time and costs in building projects. Tools that address the problem are in short supply and the task is motivated by solving this challenge. The use of artificial neural networks (ANNs) promises great support and improved feasibility to BPS, due to a reduction in overall computation time.

This thesis investigates the potential for applying ANNs to predict both annual daylight illuminance and operative temperature. The main findings from deploying a simulation model is the importance of multi sensor-node calculations for operative temperature. Operative temperature is usually calculated for the room center, in contrast to daylight where illuminance is calculated for a grid of sensor-nodes. In this study, operative temperature including long- and shortwave radiation have been calculated for a grid of sensor-nodes. The results show a significant difference in operative temperature at different locations in the room where shortwave radiation has greatest impact on the results. It is therefore important to address operative temperature in the same way as daylight illuminance, using a grid of sensor-nodes when exploring multi-objective optimization performance. However, these calculations are computational demanding and increase simulation time by 2000 %. The author has therefore investigated the potential for applying machine learning techniques, to partially replace and reduce the time-consuming simulations methods in order to achieve multi-objective design targets.

A fully connected neural network is developed with five hidden layers and five different neuron structures. The ANN model for operative temperature performed overall best for predicting annual values, reaching a CV(RMSE) of 3.8 %, an accuracy of 98 % and an average prediction within 0.47 °C. The ANN model for daylight is less accurate. The results show that direct sun exposure is difficult to predict with a five-layer ANN structure and the model often underestimates these variations. The overall model is precise but not accurate, meaning it is following the same pattern, but is consequently predicting lower temperature and illuminance values.

In general, the ANN models are showing promising results which may be integrated in a multi-objective design workflow. The results show significant time saving potential by implementing ANNs. The overall time is reduced by 96 % by using ANN models for predicting annual temperature and illuminance values.





## SAMMENDRAG

---

Mennesker bruker 87 % av tiden sin innendørs, for det meste i sin egen bolig. Innemiljøet er en avgjørende faktor for folks helse og velferd. Det er en økende utfordring med overoppheting i bygninger på grunn av varmere klima. I tillegg har sentralisering og strengere byggekoder ført til at man bygger mer kompakt, noe som gjør det mer utfordrende å oppnå dagslyskriterier i nye byggeprosjekter. Building Performance Simulation (BPS) viser seg å være en effektiv tilnærming for å støtte designet og finne en balanse mellom dagslys, termisk komfort og energiytelse. Dagens praksis innebærer at disse aspektene blir behandlet hver for seg, noe som fører til økt tidsbruk og kostnader i byggeprosjekter. Verktøy som adresserer problemstillingen er mangelfulle og oppgaven er motivert av å løse denne utfordringen. Bruken av kunstige nevralt nettverk (ANN) viser seg å være lovende og gir god støtte til BPS, på grunn av en reduksjon i den totale beregningstiden.

Denne oppgaven undersøker potensialet for å anvende ANNs til å forutsi både årlig dagslysbelysning og operativ temperatur. De viktigste funnene fra å etablere en simuleringsmodell, er viktigheten av flere beregningspunkter for beregninger av operativ temperatur. Operativ temperatur beregnes vanligvis i midten av rommet, i motsetning til dagslysberegninger hvor beregningene blir gjort for et rutenett av beregningspunkter. I denne studien er operativ temperatur beregnet for et rutenett av beregningspunkter som inkluderer lang- og kortbølgestråling. Resultatene viser en signifikant forskjell i operativ temperatur på forskjellige punkter i rommet, og kortbølgestråling viser seg å ha størst innvirkning på resultatene. Det er derfor viktig å beregne operativ temperatur på samme måte som dagslysberegninger, ved hjelp av et rutenett av beregningspunkter, når man utforsker optimalisert ytelse med flere mål. Disse beregningene er imidlertid beregningskrevende og øker simuleringstiden med 2000 %. Det er derfor undersøkt potensialet for å anvende maskinlæringsteknikker, for å delvis erstatte og redusere de tidkrevende simuleringsmetodene, for å oppnå alle design målene.

Et fullt koblet nevralt nettverk er utviklet med fem skjulte lag og fem forskjellige nevronstrukturer. ANN-modellen for operative temperatur er generelt best til å forutsi årlige verdier og nådde et avvik med CV(RMSE) på 3,8 %, nøyaktighet på 98 % og gjennomsnittlig avvik innen 0,47 °C. Dagslys ANN-modellen er mindre nøyaktig. Resultatene viser at direkte soleksponering er vanskelig å forutsi med en fem-lags ANN-struktur, og modellen undervurderer ofte disse variasjonene. Den overordnede modellen er presis, men ikke nøyaktig, noe som betyr at den følger samme mønster, men beregner konsekvent lavere temperatur- og dagslysverdier.

Generelt viser ANN-modellene lovende resultater som kan integreres i en multi-objektiv arbeidsflyt. Resultatene viser betydelig potensial i tidsbesparende ved å implementere ANNs. Den totale tiden ble redusert med 96 % ved å bruke ANN-modeller for å forutsi verdier for årlig temperatur og lysstyrke.

# TABLE OF CONTENT

---

<b>PREFACE</b> .....	<b>I</b>
<b>ABSTRACT</b> .....	<b>III</b>
<b>SAMMENDRAG</b> .....	<b>V</b>
<b>TABLE OF CONTENT</b> .....	<b>VI</b>
<b>LIST OF FIGURES</b> .....	<b>VIII</b>
<b>LIST OF TABLES</b> .....	<b>IX</b>

<b>1 INTRODUCTION</b> .....	<b>1</b>
1.1 Background .....	1
1.2 Scope of the study .....	2
1.3 Limitations.....	2
<b>2 THEORETICAL FRAMEWORK</b> .....	<b>4</b>
2.1 Visual comfort.....	4
2.1.1 Daylight factor.....	4
2.1.2 Climate based daylight .....	5
2.1.3 Norwegian building code and standard .....	5
2.2 Thermal comfort.....	5
2.2.1 Operative temperature .....	6
2.2.2 Fanger method.....	6
2.2.3 Adaptive comfort model.....	6
2.2.4 Norwegian building code and standard .....	7
2.3 Machine Learning.....	7
2.3.1 Artificial neural networks.....	8
2.3.2 Training and validation.....	8
2.3.3 Activation functions .....	9
<b>3 LITERATURE REVIEW</b> .....	<b>10</b>
3.1 Structure and process.....	10
3.2 Previous work.....	11
<b>4 METHODOLOGY</b> .....	<b>15</b>
4.1 Building performance model.....	15
4.1.1 Grasshopper scripts .....	16
4.1.2 Simulation output results.....	16
4.1.3 Sensor-nodes .....	17
4.1.4 Energy plus parameters .....	18
4.1.5 Radiance parameters.....	18
4.2 Simulation and parametric run .....	19

4.2.1	Variables used for ANN .....	19
4.2.2	Parametric model.....	20
4.3	Data processing .....	21
4.4	Artificial neural network .....	22
4.4.1	Model architecture and layer structure .....	22
4.4.2	Model activation, accuracy, and error functions .....	24
4.4.3	Optimizer and training parameters .....	25
4.5	Final testing .....	26
4.6	Use.....	27
<b>5</b>	<b>RESULTS AND DISCUSSION.....</b>	<b>28</b>
5.1	Simulation models .....	28
5.1.1	Daylight illuminance results.....	28
5.1.2	Operative temperature results.....	29
5.2	Operative temperature ANN model.....	32
5.2.1	Limited compactional budget .....	32
5.2.2	Network 6-6 results .....	33
5.2.3	Qualitative results.....	34
5.3	Daylight ANN model .....	35
5.3.1	Limited compactional budget .....	35
5.3.2	Network 2-2 results .....	35
5.3.3	Qualitative results.....	36
5.4	General ANN aspects .....	38
5.4.1	Simulation time comparison.....	38
5.4.2	Model comparison.....	38
5.4.3	ANN aspects.....	39
<b>6</b>	<b>CONCLUSION.....</b>	<b>40</b>
6.1	Main findings .....	40
6.2	Further work.....	41
<b>7</b>	<b>REFERENCES .....</b>	<b>42</b>
<b>Appendix 1   Literature matrix .....</b>		<b>I</b>
<b>Appendix 2   Grasshopper script .....</b>		<b>V</b>
<b>Appendix 3   Python ANN Model.....</b>		<b>IX</b>
<b>Appendix 4   Simulation results for case 1 .....</b>		<b>XV</b>

## LIST OF FIGURES

---

Figure 1 PPD as a function of PMV .....	6
Figure 2 Voting scales of thermal sensation.....	7
Figure 3 Illustration of a fully connected ANN.....	8
Figure 4 Function curves of sigmoid, tanh and ReLU .....	9
Figure 5 Flow diagram of the process presented in this thesis. ....	15
Figure 6 Illustration of the simulation model .....	16
Figure 7 Illustration of sensor-nodes in Rhino .....	17
Figure 8 View variables illustrated for sensor node 34.....	19
Figure 9 Illustration of the simulation process .....	20
Figure 10 Illustration of data splitting and shuffling .....	21
Figure 11 Flow diagram of the ANN training process. ....	22
Figure 12 Illustration of data batching .....	24
Figure 13 Adaptive parameters .....	26
Figure 14 Capet plot of annual hourly illuminance for sensor node 18 (a) and 38 (b).....	28
Figure 15 Illustration of difference in direct sun exposure .....	29
Figure 16 Number of hours with operative temperature above 26 °C.....	30
Figure 17 Carpet plot of annual hourly operative temperature.....	31
Figure 18 Carpet plot of difference in operative temperature .....	31
Figure 19 Operative temperature ANN model performance .....	32
Figure 20 Network 6-6 error, momentum and accuracy performance .....	33
Figure 21 Network 6-6 predictions for annual hourly operative temperature .....	34
Figure 22 Network 6-6 predictions for hourly operative temperature.....	34
Figure 23 Daylight ANN model performance.....	35
Figure 24 Network 2-2 error, momentum and accuracy performance .....	36
Figure 25 Network 2-2 predictions for annual hourly illuminance .....	36
Figure 26 Network 2-2 predictions for hourly illuminance.....	37

# LIST OF TABLES

---

Table 1 Search hits from Oria and ScienceDirect ..... 10

Table 2 Energy plus design parameters ..... 18

Table 3 Radiance design parameters ..... 19

Table 4 Radiance simulation parameters..... 19

Table 5 Solution space of the parametric analysis. .... 20

Table 6 Overview of ANN architecture ..... 23

Table 7 Operative temperature error and accuracy performance ..... 33

Table 8 Illuminance error and accuracy performance ..... 35

Table 9 Time spent on simulations and running ANN model ..... 38

# 1 INTRODUCTION

---

The goal of this thesis is to explore the potential for applying artificial neural networks (ANNs) to predict both annual daylight illuminance and operative temperature, in order to reduce time-consuming simulation methods. Promising machine learning approaches from the literature are implemented and evaluated for performance.

This first section of this thesis provides an introduction and overview of visual and thermal comfort and machine learning. A literature study is conducted within this field related to ANNs. The next section consists of the methodology and the development process. Results and discussion are addressed for both the simulation results and the developed ANN. Lastly, a conclusion is presented with a recommendation on future work.

## 1.1 BACKGROUND

Humans spend 87 % of their time indoors, mostly in their own residence. The indoor environment is a crucial factor for people's health and welfare (Klepeis et al., 2001). It is well-known that windows have a considerable impact on both energy use and indoor environment. Study shows that long-term impact of attending a daylight school, could result in a 14 % increase in student performance in contrast to a classroom without windows (Bailey and Nicklas, 1996). Buildings also account for 40 % of energy use in the EU (EPBD, 2010). In recent years buildings in Norway have become more insulated and airtight due to more ambitious building projects with certifications such as BREEAM and ZEB. With hotter climate there has been increasingly challenge with overheating in buildings (Tian and Hrynyszyn, 2020). In addition, centralization and stricter building codes has led us to build more compact, making it more challenging to achieve daylight criteria in new building projects (Reinhart and Selkowitz, 2006, Chen and Yang, 2015).

Given the increasing complexity of energy and environmental challenges the building sector is facing, Building Performance Simulation (BPS) is proving to be an effective approach for supporting the design and operation of high-performance buildings, such as zero-energy buildings or zero-emission buildings (Clarke and Hensen, 2015, Wate et al., 2019). It is important to find a balance between daylight availability, thermal comfort and energy use, if we are to achieve both the goal of a nearly zero energy consumption and buildings with a healthy and comfortable indoor environment (Yu et al., 2020, Ruck et al., 2000).

Simulation-based multi-objective optimizations is widely applied when optimizing both thermal and daylighting performance. Some of these methods are genetic algorithm, weighted sum method and non-dominated sorting genetic algorithm II (NSGA-II). These algorithms offers a Pareto-optimal front that shows the best trade-offs between daylight and thermal comfort objectives. These methods

effectively shows the balance between the optimal objectives, but at the cost of a large number of calculations and computation (Yu et al., 2020). Time series forecasting are an active research area, which have received a considerable amount of attention in the literature (Rozenberg et al., 2012). In parametric design environments, the use of ANNs promises greater feasibility than simulations for exploring the performance of different solutions, due to a reduction in overall computation time (Lorenz et al., 2020).

### 1.2 SCOPE OF THE STUDY

This study investigates the potential for applying ANNs to predict both annual daylight illuminance and operative temperature in residential buildings. In current practice, both daylight illuminance and operative temperature are treated separately, which leads to increased time and costs for new building projects. Tools that address the problem are in short supply and the task is motivated by solving this challenge.

This study aims to establish a basis for a learning algorithm to partially replace time-consuming BPS tools in the design optimization processes. To the best of the authors' knowledge, there are no studies that use ANNs to predict annual climate-based metrics, for both daylight illuminance and operative temperature, in parametric design environments. This thesis aims to introduce a proof of concept, which predicts annual hourly illuminance and operative temperature for a grid of sensor-nodes in a zone, based on geometric data. The originality of this research lies within the following objectives:

1. Review state-of-the-art research, related to ANNs, thermal and visual comfort.
2. Deploy a BPS model with grid-based calculations, for annual daylight illuminance and operative temperature.
3. Develop a machine learning algorithm, which can partially replace and reduce the time-consuming simulations methods, in order to achieve multi-objective targets.

The methodology consists of generating training, validation and testing data by using parametric analysis and exhaustive search. The generated data is based on theoretical reference building and represent middle-class housing of the existing buildings in Norway. A custom ANN will be developed with suitable model architectures and hyper-parameters. Validating and testing the ANN results will also be addressed.

### 1.3 LIMITATIONS

There may be some possible limitations in this study. This thesis is conducted spring 2021 at NTNU and counts towards 30 credit and was conducted over 22 weeks. It is reasonable to assume that a longer timeframe would lead to more valid and representative results. Both visual and thermal comfort and ANNs are large research topics, there has been conducted and produced a lot within this field the last decade. Some relevant papers might not be included or thoroughly explored. Before starting the

thesis, the author had limited knowledge of ANNs. It has not been a part of the learning objectives for the study program.

It would be preferable to use real data in existing buildings for training and validation, but this would be cost and time-consuming, because it would require 216 different building solutions. The generated data is therefore based on simulation, which uses tools that are validated by real measurements. In addition, the study uses a simplified data model with limited input variables. This means that the model is restricted and sensitive to only these variables. The current simulations are focused on variables from continuous distributions, but it is unclear how well the results would generalize if variables were drawn from discrete distributions. Additional research would be required to determine how well the data with frequent variations would develop.

The Grasshopper-scripts for data generation, and the code associated with the machine learning model, is not validated. It is not guaranteed that all models are optimized to the full extent. However, effort has been made to reduce errors by implementing code standards and a thorough evaluation of the results.



## 2 THEORETICAL FRAMEWORK

---

In this chapter, the theoretical framework for thermal comfort, visual comfort and machine learning, will be detailed. The factors and metrics, which are commonly used for indoor comfort, and the regulated Norwegian laws for building comfort, is addressed. An introduction to machine learning and an overview of its structure, relevant for this thesis, will also be addressed.

### 2.1 VISUAL COMFORT

Visual comfort is a subjective reaction to the quantity and quality of light, within any given space at a given time, and for providing an adequate view to the outdoors. It can be categorized by daylight provision, quality of light, view to the outdoors and protection from glare. Daylight is described as the combination of all direct and indirect light, originating from the sun during daytime. Daylight design involves carefully balancing heat gain and loss, glare control, and variations in daylight availability. Visual parameters are directly related to the physiology of the eye (Ruck et al., 2000). It influences our performance (the visual system), as well as affect our health (the circadian system) and personal well-being (the perceptual system) (Andersen et al., 2014).

Illuminance is the measure of the amount of light received on a surface and is typically expressed in lux ( $\text{lm}/\text{m}^2$ ). It can either be measured by a luxmeter or calculated by simulation software. Typical minimum requirements range from 200 – 500 lux depending on the activities in the room. Luminance is the measure of the amount of light reflected from a surface and is typically expressed in  $\text{cd}/\text{m}^2$ . It can be measured with a luminance meter, high dynamic range techniques with digital camera, or by simulation software. Typical values ranges from cloudy sky with 2000  $\text{cd}/\text{m}^2$  to clear sky with 8000  $\text{cd}/\text{m}^2$  (Andersen et al., 2014).

#### 2.1.1 Daylight factor

Daylight factor (DF) is an availability metric, that expresses the ratio of natural light inside a room compared unobstructed natural light outside the room under CIE standard overcast sky conditions, expressed in percent (CIE, 1996). Often a grid of sensor-nodes, representing a whole room, is used. It is possible to calculate DF with a simulation software or luxmeter. Physical measurement is somewhat difficult and time consuming, due to the requirement of clouded sky, unobstructed horizon and measurement of two places at once (inside and outside). Typical average daylight factor values range from 2-5% depending on the usage. This metric does not account for mechanical shading devices, location of the building and time of the year (Andersen et al., 2014).

### 2.1.2 Climate based daylight

Daylight autonomy (DA) is an availability metric for annual occurrence of illuminance that corresponds to the occupied time. It is calculated by counting the number of hours when the target illuminance at a point in space is met by daylight, expressed in percentage of hours (Andersen et al., 2014). Threshold limits are usually set to 100, 300 or 500 lux, depending on the criteria and design goals (Norsk Standard, 2019a). Useful daylight illuminance (UDI) is also availability metric for annual occurrence of illuminance at a given point that falls within a threshold. It is similar to DA, but with an upward threshold limitation. The metric indicates the levels of daylight that are associated with occupant discomfort and unwanted solar gain based on the horizontal illuminance level (Andersen et al., 2014, Carlucci et al., 2015). Threshold values can range from a minimum of 100, 300 and 500 lux to a maximum of 2000, 2500 and 8000 lux (Mardaljevic et al., 2009, David et al., 2011, Olbina and Beliveau, 2009, Carlucci et al., 2015).

The climate-based daylight method gives a more realistic evaluation of the building performance compared to DF. This is because the DF evaluates illuminance for one time-step ( $T=1$ ), and the climate-based daylight evaluates hourly annual illuminance for multiple time-steps ( $T=8760$ ), using the local climate.

### 2.1.3 Norwegian building code and standard

It is stated in the Norwegian building code, TEK17, that rooms for long term stay shall have an average DF above 2.0 %, using simulation software validated according to CIE 171:2006. It can alternatively be documented with the 10 %-roule, which uses the window area in respect to the floor area and transmission factor of the window (TEK17, 2017 , CIE, 2006). The 10 %-roule method is not recommended by the norwegian Consulting Engineers Association (RIF), due to the lack of coherence with the average DF calculation. RIF has recently published a industry standard on how the daylight calculation should be conducted. This includes description of the model set-up, different metrics and also a checklist (RIF, 2020).

## 2.2 THERMAL COMFORT

Thermal comfort is a subjective reaction, where the condition of mind expresses satisfaction with the thermal environment (Norsk Standard, 2006). Designs should provide good thermal conditions based on energy efficient technologies like natural ventilation, solar shading and intelligent building design (Andersen et al., 2014). The environmental variables that influence thermal comfort are the air velocity, turbulence, relative humidity, air temperature and mean radiant temperature. Two other important variables are the person's activity level, commonly referred to as metabolic rate and clothing level, referred to as the clothing index (Mysen, 2017).

### 2.2.1 Operative temperature

Operative temperature is calculated using dry-bulb air temperature, mean radiant temperature and air velocity. When air velocities are low (below 0.1 m/s), which is typical for spaces inside buildings, the operative temperature can be the average value of dry bulb temperature and mean radiant temperature. In buildings where surfaces may be heated or cooled, or there is significant thermal mass and solar radiation, air temperature and radiant temperatures may be very different (Myhren and Holmberg, 2006). Operative temperature can be a reasonable indicator of thermal comfort, this is usually measured by the maximum operative temperature, or number of hours with unacceptable operative temperatures calculated for one year. Usually operative temperatures between 19-26 °C is acceptable (Sintef and NTNU, 2007, Norsk Standard, 2019b).

### 2.2.2 Fanger method

A model is developed by Fanger in order to assess the thermal environment based on the different environmental variables. The predicted mean vote (PMV) index provides information on the degree of discomfort experienced in a thermal environment for a larger group of people, ranging from +3 to -3, representing too warm or too cold surroundings. The percentage of people dissatisfied (PPD) is based on the PMV index and includes clothing and activity level (Fanger, 1970, Shaw, 1972). PMV and PPD are illustrated in Figure 1, at least 5 % of the people in a group will be dissatisfied with the thermal environment, even if the PMV is zero (Norsk Standard, 2006, Mysen, 2017).

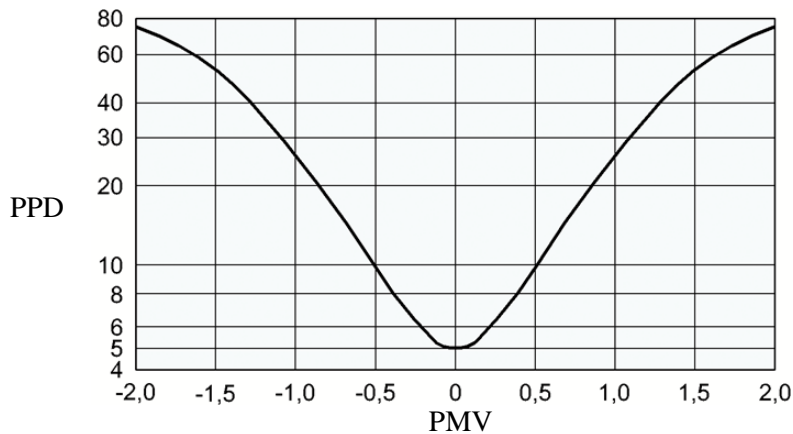


Figure 1 PPD as a function of PMV (Mysen, 2017).

### 2.2.3 Adaptive comfort model

It is found that occupants have a higher tolerance for temperature changes in naturally ventilated buildings in comparison to air-cooled buildings. Due to the lack of adaptability for both behavioral and physical adjustment in the Fanger comfort model, an adaptive comfort model is developed for use in buildings without mechanical cooling. This method is primarily applicable for conditions where the occupant have access to windows and can adapt clothing levels, according to indoor and outdoor temperature (Brager and de Dear, 1998, Norsk Standard, 2019b).

Instead of using models based on extensive laboratory tests, a new adaptive method is introduced by using a large number of thermal comfort data. This model includes environmental parameters, personal parameters, climatic types, and adaptive control measures. The model is trained in a machine learning algorithm which can predict occupants thermal comfort votes (TCV) and thermal sensation votes (TSV). These rating scales are similar to PMV and ranges from +3 to -3, see Figure 2 (Chai et al., 2020).

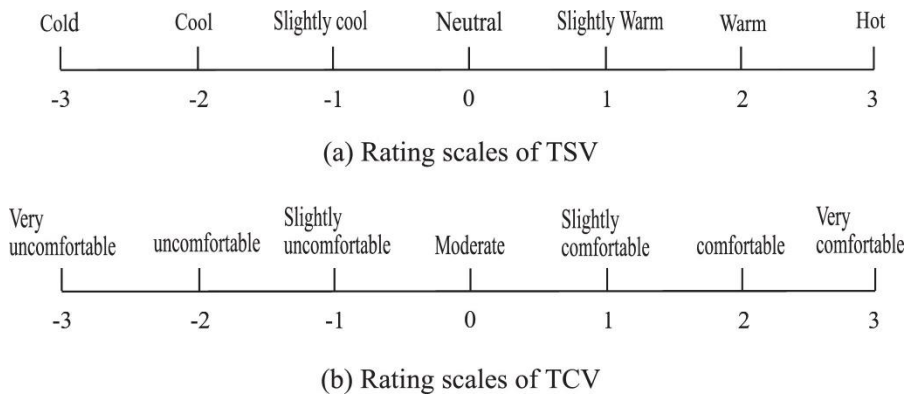


Figure 2 Voting scales of thermal sensation (a) and thermal comfort (b) (Chai et al., 2020).

#### 2.2.4 Norwegian building code and standard

It is stated in the Norwegian building code, TEK17, that occupied rooms for long term stay shall have an operative temperature between 19-26 °C for light work, 16-26 °C for medium work and 10-26 °C for hard work. For the same reasons stated in the adaptive comfort model, it is accepted to deviate from these values for hot summer periods, with an outdoor temperature that is exceeded by 50 hours in a normal year (TEK17, 2017).

The Norwegian Labour Inspection Authority, Arbeidstilsynet, states that operating temperature in the workplace should be within 19-26 °C. Furthermore, it is stated that exceedances of the highest limit should be acceptable during hot summer periods at outdoor air temperatures above 22 °C, it should not exceed 50 hours per year within the workhours (Arbeidstilsynet, 2016).

### 2.3 MACHINE LEARNING

Machine learning is a branch which is set out of artificial intelligence. A machine learning algorithm can identify and learn underlying patterns in observed data, without relying on a predetermined equation as a model. This is done to generate insight, make predictions and better decisions. The algorithms improve their performance as the number of samples available for learning increases (Patterson, 1996). It is essentially a form of applied statistics with increased emphasis on the use of computers to statistically estimate complicated functions (da Silva et al., 2016). Most machine learning algorithms can be divided into unsupervised learning and supervised learning. Unsupervised learning is an approach that learns from data that is unlabeled or classified. In supervised learning, the algorithm attempts to learn from informative examples of labeled data (Goodfellow et al., 2016).

### 2.3.1 Artificial neural networks

Artificial Neural Networks (ANNs) is a computational network based on how the biological nervous system work. The mathematical expression of a multiple-input neuron is shown in equation 1. A neuron  $z$  receives multiple input signals  $x_i$  from other neurons through a transfer function  $f$ , also called activation function. All the inputs are multiplied by a weight  $w_i$  which is computed based on their importance and is then summed together. A bias  $b$  is added to provide the ability to shift and increase the adaptability of neurons. The network is built upon numerous of these neurons, that work collectively to process data and is the basis of deep learning models (Zhang, 2010).

$$z = f\left(\sum_{i=1}^n w_i x_i + b\right) \quad (1)$$

The network can learn complex problems by arranging neurons into more layers consisting of multiple non-linear activation functions. This arrangement is called multilayer perceptron, where usually the structure consists of an input layer, an output layer and multiple hidden layers in between. The input and output are visible because the variables are defined in these layers. The value of each neuron is dependent on the output from the previous layers (Goodfellow et al., 2016). A fully connected ANN consists of a series of fully connected layers, that connect every neuron in one layer to every neuron in the next layer, illustrated in Figure 3.

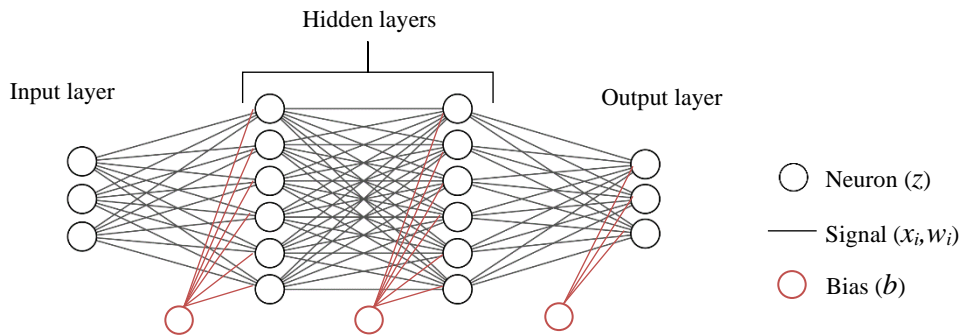


Figure 3 Illustration of a fully connected ANN with two hidden layers.

### 2.3.2 Training and validation

To be able to identify and learn the underlying patterns, the algorithm must train on rich and accurate data. This is done by initializing the model with random weights, and then calculate the error (loss) between predicted values and expected values (ground truth). The model is then optimized to minimize the loss by adjusting the weights for every iteration. A common optimization technique is stochastic gradient descent, where the impact of the optimized weight depends on the learning rate and momentum (Zhang, 2010).

When the model has iterated through all the training data, described as an epoch, the model is validated with a separate dataset. This is to measure the model accuracy after the weights are updated (Goodfellow et al., 2016). Overfitting occurs when a prediction corresponds too closely to a particular dataset. This can happen in cases where the model has limited training data, even if it is drawn from the same distribution, or if the model is training for too long (Srivastava et al., 2014). This can cause the model to adjust to very specific features of the training data, that have no relation to the target function. The performance on the training data will still increase, while the performance on unseen data (validation and testing data) becomes worse (Goodfellow et al., 2016). This can be solved with a dropout technique which randomly zeroes some of the elements of the input tensor while training. This method has proven to be an effective technique for regularization and preventing overfitting (Hinton et al., 2012).

### 2.3.3 Activation functions

Activation functions are mathematical equations that determine the output of a neuron. They are important for the ANN ability to converge and the convergence speed. The most frequently used activation functions are sigmoid, tanh and ReLU, illustrated in Figure 4 (Gao et al., 2019).

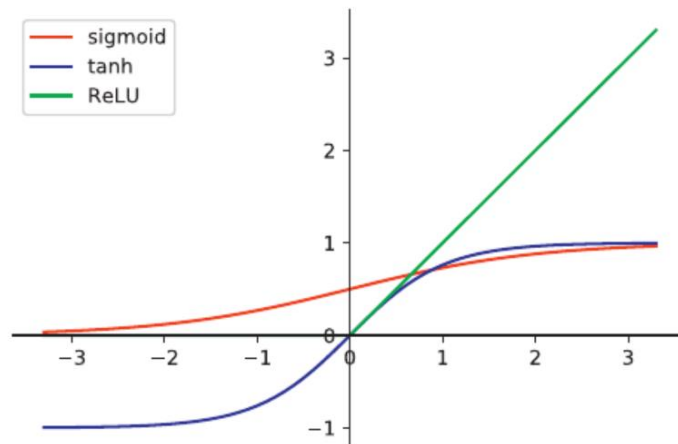


Figure 4 Function curves of sigmoid, tanh and ReLU (Gao et al., 2019).

One of the main drawbacks of sigmoid and tanh is that they can cause exploding gradients when stochastic gradient descent optimizers are used to train the network. Exploding gradients occurs when large error gradients accumulate, which results in very large updates to the ANN model weights during the training (Gao et al., 2019). To overcome this drawback, the ReLU function is applied to squash the output. This will make the ANN converge much faster when the stochastic gradient method is used to train the network (Krizhevsky et al., 2017).

### 3 LITERATURE REVIEW

---

The first part of this thesis aims to collect relevant state-of-the-art literature for indoor visual and thermal comfort, related to multi-objective optimizations and ANNs. This is to gather, define concepts and clarify a starting point for further discussion. This is done by a quantitative and qualitative literature study, where a large quantity of literature is gathered, and analyzed. The highest relevancy literature is further investigated and detailed.

#### 3.1 STRUCTURE AND PROCESS

Oria and ScienceDirect are used as main search engine with topics indicated in Table 1. These have been filtered with research from 2010 or later, language and peer review. In addition, Google was used for more general knowledge of a field or area of study, which was later implemented in the search words. For literature of high relevancy, cited papers was also investigated and used, even if it did not match the filtered criteria. All gathered literature is connected to visual and thermal comfort, it does not need to contain both comfort categories. Also, findings from research on specific ANN algorithms or structure is further studied.

Table 1 Search hits from Oria and ScienceDirect.

Search engine	Topic	Hits
ScienceDirect	Thermal comfort + artificial neural network	2403
ScienceDirect	Visual comfort + artificial neural network	1515
ScienceDirect	Thermal + visual + comfort + artificial neural network	734
Oria	Visual + thermal + comfort	6068
Oria	Operative temperature + artificial neural network	1738
Oria	Thermal comfort + Artificial neural network	1652
Oria	Daylight + Artificial neural network	1365
Oria	Machine learning + prediction + operative temperature	1152
Oria	Thermal and visual comfort + artificial neural network	413
Oria	Illuminance + artificial neural network	484
Oria	Machine learning + prediction + illuminance	254

Endnote X9 was used to organize all the relevant literature and cited papers. The information was rated on a five-point scale and given a short-summarized description and conclusion. This was later transferred to a literature matrix which can be found in Appendix 1. The matrix consists of information on aspect with regards to thermal comfort, visual comfort, and important geometric variables, such as window properties and window-wall/window-floor ratios. The matrix also consists of information on which methods that has been conducted with details on output metrics, ANN structure and simulation software. The rating system was divided into three categories from low (<1 point), medium (2-3 points) to high (>4 points) relevancy. The highest relevancy literature is described in more detailed the next section.

### 3.2 PREVIOUS WORK

Lorenz et al. found great time saving potential when using ANNs for daylight simulations. The study explored design solutions in an atrium, with the aim of bringing more daylight into its adjacent spaces on the lower floors. Variables included atrium geometry, orientation and Window to wall ratio (WWR) resulting in 54 possible solutions. Daylight Autonomy (DA) and spatial Daylight Autonomy (sDA) were used to assess the daylighting potential of the design variants. The ANNs consist of a Levenberg-Marquardt algorithm using a mean squared error (MSE) loss function and gradient descent optimizer. The data was divided into 65% training data, 20 % validation data and 15 % test data. Due to the achievable time-savings, ANNs offer a possibility to readapt the brute-force approach into the design process. This enabled all possible instances in the parameterized design solution space to be examined. The total simulation time was reduced by 65 % (Lorenz et al., 2020).

Bre et al. also found great time saving potential when using ANNs for energy and thermal comfort simulations. They propose an optimal way to generate samples used to train and validate the ANN, minimizing the total of building energy simulations necessary to train them. The ANN model consists of a Levenberg–Marquardt algorithm with a Bayesian regularization, which improves the capacity of generalization of the ANN. This was done by applying a regularization process, which prevents the model for overfitting. Results indicate that the presented method was able to reduce the number of building energy simulations up to 75 %, while keeping a good accuracy of the results (Bre et al., 2020).

Zhou and Liu studied three different machine learning algorithms with simulated data, in order to predict hourly results for climate-based UDI (Useful Daylight Illuminance). The study also conducted a literature review, which shows that currently only a very few tools integrate the machine learning techniques. The models consist of ANN, principal component analysis (PCA) and support vector machine (SVM). Results show that the neural network using PCA generated the highest accuracy compared to other algorithms, about 96 %. The model was limited to binary output that determined whether daylight levels fell within a specific illuminance range. The models could not only reduce their design and testing times, but could also provide sensibility analysis by allowing them to view the trend of different daylighting design strategies (Zhou and Liu, 2015).

In research connected to temperature forecasting, Tran et al. studied the relationship between ANN specifications and performance. 15 models with different layer structure was used, ranging from 3 to 5 hidden layers with 1 to 125 neurons in each layer. Root mean square error (RMSE) was used as loss function and a hyperbolic tangent was used as activation function. The data was normalized and divided into 70 % training data, 20 % validation data and 10 % test data. They discovered that if the neuron size were too large, the RMSE would be generally increased, and the model ability to predict was impaired. Too many hidden layers would also reduce the system prediction, if the number of



neurons was high. The result showed that the five-hidden layer model produced the smallest RMSE (Tran et al., 2020).

Yu et al. presents a novel multi-objective optimization model using ANN for energy and thermal comfort predictions. Important geometric variables that affect the building performance are building form factor, window transparency, orientation, window-to-wall ratio (WWR), thermo-physical properties, and the interior space layout. Mean squared error (MSE) was used as loss function and the training data was normalized. The ANN consist of one hidden layer with 13 neurons. The model was trained for a maximum of 2000 epoch. The relative error between prediction and simulation was below 1.7 % and 2.1 % for energy use and thermal comfort respectively. The prediction results suggest that the model is an effective tool for building optimization design (Yu et al., 2015).

Kazanasmaz et al. developed a prediction model for daylight illuminance. The model was based on 24 different office spaces divided on two floors. Three months of measured data at four positions was recorded for all rooms. The ANN was developed in the neural network software NeuroSolutions and consist of three layers with one output node. Input variables used for the ANN model was date, hour, distance to windows, number of windows, orientation of rooms, floor identification, room dimensions and point identification. Weather determinants such as outdoor temperature, solar radiation, humidity, UV index and UV dose were also used. The data was divided into 80 % training data and 20 % testing data. A simplex optimization function was used, and the model ran for 10 000 epochs. The model reached a 98 % accuracy and the optimum number of neurons was found to be seven, giving the model an average error of 2.2 % (Kazanasmaz et al., 2009).

Shaghaghian and Yan found good predictions for shape classification using Generative Adversarial Networks (GANs) and Convolutional Neural Networks (CNNs). The experiment was conducted by using Grasshopper for generating shapes and class labels, representing floor plans. The dataset consists of 6000 images of simple 2D shapes with labels. The CNN was trained for classifying labels for each shape and the GAN was trained to generate shapes from the labels. The results show 93 % accuracy when testing with manually drawn shapes. The trained GAN model can generate new images of façade pattern (window/wall), based on daylight performance simulation. Some of the generated patterns was not within the search space of the original model and can be regarded as novel design options that meet the demanded performance (Shaghaghian and Yan, 2019).

Energy savings potential in optimizing window size was found in research from Canada. The model was trained in an ANN to provide acceptable approximations of the simulation results. 450 training cases was needed to reach errors below 1 % for the total energy use, and below 4 % for the average PMV. The ANN results were then optimized in a multi-objective evolutionary algorithm, NSGA-II, to enable fast evaluations. Simulation time required to create the training data for the ANN was 3 weeks. If the NSGA-II was linked directly to the simulation software, each optimization would have taken

more than 10 years. The ANN performed very well in terms of predicting energy use, but generally underestimate the PMV value. It was stated that this should be further investigated (Magnier and Haghghat, 2010).

Ngarambe et al. developed and compared the performance of five common ML algorithms for predicting hourly illuminance based on generic simulated data. The algorithms consisting of generalized linear models, deep neural networks, random forest, long short-term memory networks and gradient boosting models. The models considered a total of 14 input variables consisting of window to wall ratio (WWR), wall reflectance (WR), and distance from the window (DFW), global horizontal irradiance (GHI), direct normal irradiance (DNI), diffuse horizontal irradiance (DHI) global horizontal illuminance (GHIL), direct normal illuminance (DNIL), relative humidity (RH), and sky cover (SC). The data was divided into 80 % training data and 20 % validation data. RMSE was used as loss function and Adaptive Momentum Estimation (Adam) was used as optimizer. Rectified linear units (ReLU) were used as activation functions and were reported to enable better performance than Tanh and Maxout functions. Deep ANNs outperformed the other network with a accuracy of 99 %. The results also indicated that distance from window, time of day, direct normal irradiance, and diffuse horizontal irradiance were the most important variables for distribution of indoor daylight illuminance (Ngarambe et al., 2020).

Palladino et al. used ANNs for assessment to thermal comfort for predicting PMV. Three input variables were used including indoor air temperature, relative humidity, and clothing insulation. The network consists of one hidden layer. The data was divided into 70 % training data, 15 % validation data and 15 % testing data. It was concluded that the model was suitable for the simplified calculation of PMV, reaching global regression value of 93 % (Palladino et al., 2020).

Lorenz et al. used a Levenberg-Marquardt algorithm for predicting Daylight Autonomy (DA) levels in an office building. The model consists one hidden layer with 3 to 25 neurons. The data was divided into 70 % training data and 30 % validation data. Mean Squared Error (MSE) was used as loss function and tanh was used as activation function. The model ran for 1000 epochs. The model predicted DA results within a 3 % range. The network with 13 neurons returned lowest error and had best predictions (Lorenz et al., 2018).

A paper from Denmark study the relationship between glazing-to-floor ratio, orientation, and glazing properties. The model consists of rooms with different geometries, representing Danish ‘nearly zero-energy’ single-family houses. The criteria used in the paper is based on the Danish Building code for nearly-zero residential buildings. For thermal comfort, the requirement is no more than 100 hours, where the operative temperature exceeds 26 °C. A modified version of the daylight factor metric was used. This created a connectivity for the diffuse daylight access at a specific location, which made it a climate dependent. On the assumption that the diffuse daylight access follows the same brightness as

the CIE overcast sky model, a target daylight factor for various locations was derived, based on the target for median illuminance indoors and the median diffuse illuminance available during daylight hours. The target illuminance indoors was set to 300 lx, which for Copenhagen gives a target daylight factor of 2.1 % across 50 % of the work plane. The target used for daylight evaluation was chosen to reflect a specific location, but does not take into account realistic sun and sky conditions, because it was based on evaluation of the daylight target under a CIE overcast sky (Vanhoutteghem et al., 2015).

A study from China presents a simulation optimization tool, which find the optimal trade-off between minimizing energy, maximizing thermal comfort and maximizing daylight probability. The results show various designs depending on which criteria one would like to use. The DA metric has an upper, limit which makes it difficult for comparison between different solutions. The optimization results indicate that all three typologies of spatial configuration have great potential for energy efficiency and comfort improvement. The overall best results are calculated by the value on the pareto front closest to the origin (Zhang et al., 2017).

The most common way to describe the geometry information of the opening in the façade is window-to-wall ratio (WWR), window-to-floor ratio (WFR), height-to-width ratio (HWR) and the height of windowsill (SH). WWR and WFR with different orientations is widely investigated and has been shown to have a significant impact on both thermal and daylighting performances. External screen can reduce operative temperatures, but openable windows are more effective in reducing overheating hours. Natural ventilation proves to be an effective measure to reduce high temperatures in summer and reduce overheating hours significantly for single-family houses in Norway (Haase and Grynning, 2017).

## 4 METHODOLOGY

A flow diagram of the conducted work is illustrated in Figure 5. The gray dashed rectangles represent a group of processes, which is further described in this chapter. It starts with developing a building performance model (1a) where the geometric properties, parameters and variables are defined. Then all the test cases are simulated and executed parametrically using exhaustive search (1b). After simulation the data is processed (2), where illuminance and operative temperature results are divided into three datasets where 70 % is training, 15 % is validation, and 15 % is testing. Then, the data is trained in an ANN (3). After training and reaching acceptable loss and accuracy, the model is tested (final validation) with the test dataset (4). This presents the final accuracy of the model by using a holdout dataset. Finally, the trained and tested model can be used (5) with a new building model, providing the user accurate and fast feedback on the building design.

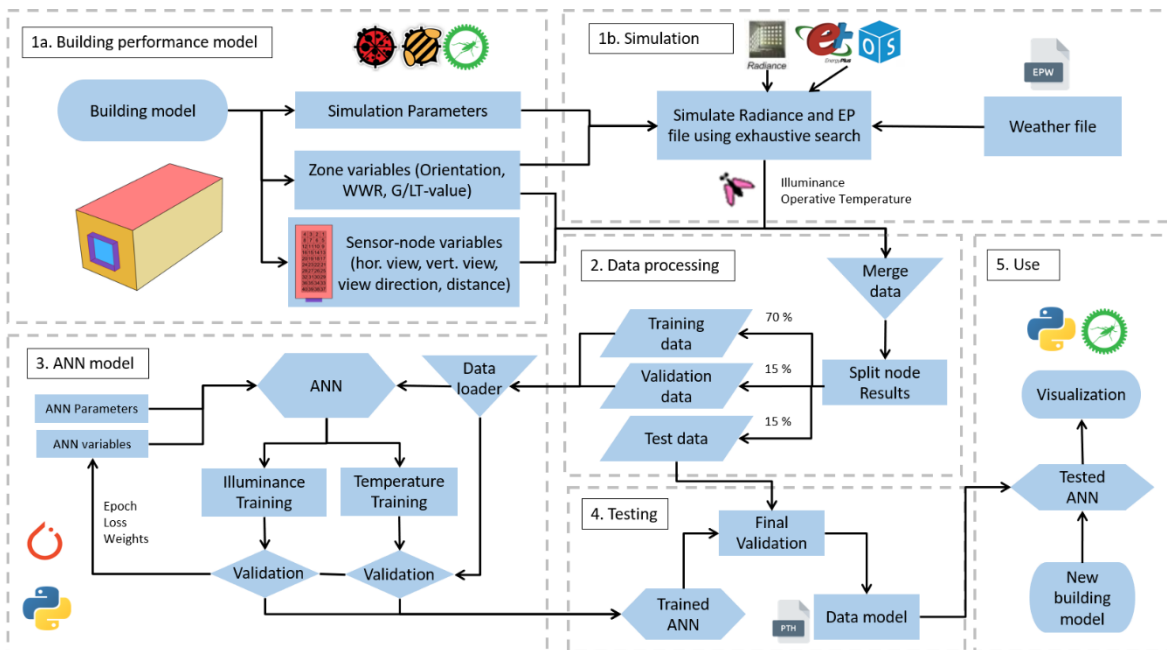


Figure 5 Flow diagram of the process presented in this thesis.

### 4.1 BUILDING PERFORMANCE MODEL

The first step of the methodology, numbered “1a” in Figure 5, is to develop the BPS model to generate training, validation and test data. It would be preferable to use real measured data from existing buildings for training and validation, but this would be cost- and time-consuming. A building model is therefore developed using BPS tools, which is validated with real building models. Explorative use of BPS tools, for research and development, might be a potential pitfall (Loonen et al., 2019). Although these tools are mainly used for generating generic room models and not extreme model cases.

The generic simulation model is based on theoretical reference building and represent a room form a middle-class housing in Norway. The basic geometric properties are based on test-cases from the CIE 171:2006 standard and the ANSI/ASHRAE Standard 140-2001 (CIE, 2006, ASHRAE, 2001). The model is slightly modified, in order to represent a combination for daylight illuminance and operative temperature. This is done for the purpose of comparison between simulated and predicted results. The generic model consists of one rectangular room (3.0 m wide, 6.0 m long and 2.7 m high), with no interior partitions and one-sided window, se illustration in Figure 6.

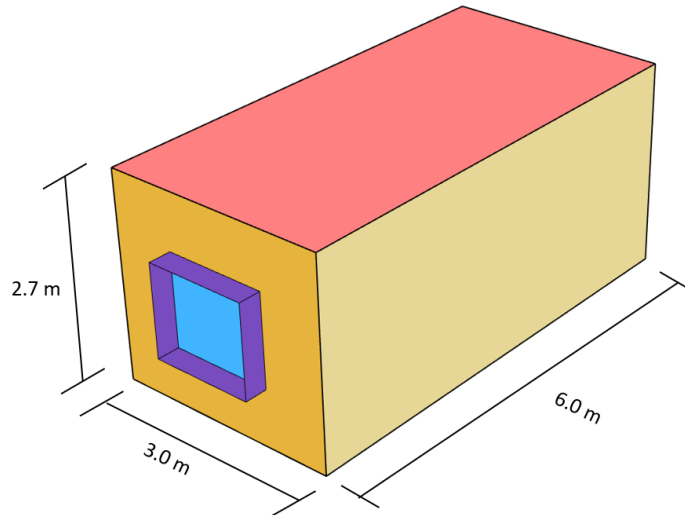


Figure 6 Illustration of the simulation model in Rhino.

#### 4.1.1 Grasshopper scripts

Building geometry and parametric variables are designed in Grasshopper graphical algorithm editor version 1.0. The editor is a built-in plugin in Rhinoceros 3D modeling tool, which is widely used by architects and engineers, and is interoperable with BIM tools. Rhinoceros version 6 SR34 is used. Ladybug tools version 1.2.0, which is integrated into Grasshopper, is used for BPS.

Two scripts are developed for generating training data, one for visual comfort and calculation daylight illuminance, and another for thermal comfort and calculation operative temperature. The graphical color-coding standard (Dynamo Standard) for Grasshopper, developed by Vladimir Ondejcik, is implemented as a part of the code validation and readability (Ondejcik, 2016). An overview of the scrips, developed in Grasshopper, can be found in Appendix 2.

#### 4.1.2 Simulation output results

Operative temperature is a common thermal comfort metric and is chosen as output result for this study. This is because it is connected to the Norwegian building standard and used for several other common thermal comfort metrics, such as PMV, TCV and TSV, as described in the theory section. Operative temperature is calculated based on dry bulb air temperature, mean radiant temperature and air velocity.

Annual daylight illuminance is a common visual comfort metric and is chosen as output result for this study. This is because it is connected to several common daylight metrics such as UDI and DA and provides the model with a more flexible usage. Illuminance is the measure of the amount of light received on a surface, and is typically expressed in lux ( $\text{lm}/\text{m}^2$ ) (Andersen et al., 2014).

### 4.1.3 Sensor-nodes

It is found from the literature study that most ANNs are trained to predict a single output value. This can be metrics such as DA and PMV. In this study the goal is to utilize ANNs and use annual hourly data for several sensor nodes, representing every position in the room. This is because in buildings where there is significant solar radiation, air temperatures and radiant temperatures may be very different (Myhren and Holmberg, 2006). The operative temperature is therefore calculated as a grid of sensor-nodes in the same way as daylight illuminance. This makes it possible to take account for radiant surface temperatures (longwave radiation) at different positions in the zone. In addition, the effect of direct sun exposure (shortwave radiation) is included for the operative temperature. This is done by calculating mean radiant temperature (MRT) for each position using the SolarCal model of ASHRAE-55 (ASHRAE, 2017). This method estimates the effects of shortwave solar and sky exposure to determine longwave radiant exchange. It is assumed that the whole body is irradiated if the sensor-node is irradiated.

Sensor-nodes have been set with 0.5 m distance between nodes, and 0.5 m from walls based on NS-EN 12464-1:2011 (Norsk Standard, 2011), resulting in a total of 40 sensor-nodes for each model. Figure 7 illustrates the location of the sensor-nodes. The Norwegian building research design guides from SINTEF suggest a distance of 0.6 m from walls for thermal models (Mysen, 2017). This is reduced to 0.5 m, in order to compare the results with annual daylight illuminance. There is no furniture present in the room.

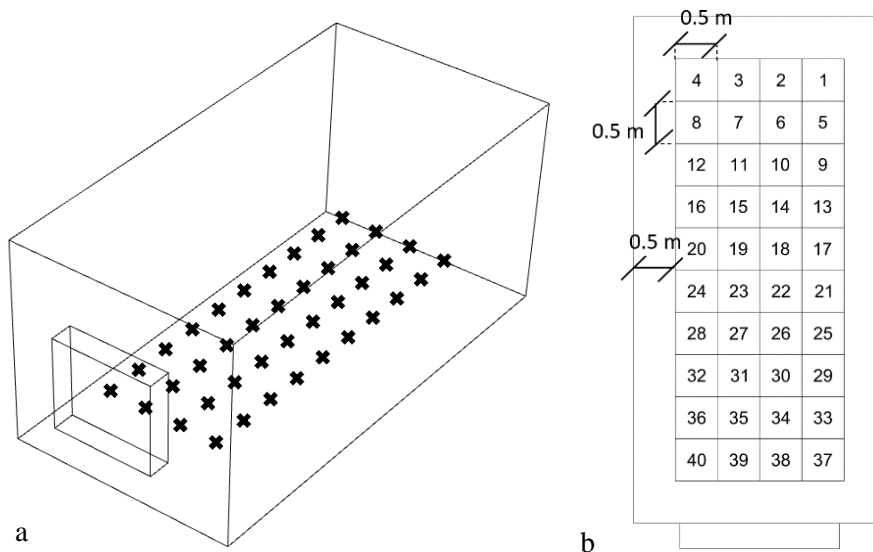


Figure 7 Illustration of sensor-nodes in Rhino, with overview of the 3d zone (a) and sensor-node spacing (b).

#### 4.1.4 Energy plus parameters

Model input parameters for energy is applied according to Table 2. There is a continuous internal load of 132 W for light and equipment, based on NS 3031:2014 in addition to one adult person, which is always present (Norsk Standard, 2014). The building is free-running and does not make any use of mechanical heating or cooling. Window ventilation is activated above 22 °C, and the operable area is 50 % of the window area. Only one wall is facing outdoors, all other surfaces are adiabatic. The U-value for the wall is set to 0.16 W/m<sup>2</sup>k, and to 0.80 W/m<sup>2</sup>k for the windows. Inner surfaces for walls are built of gypsum, floor and ceiling is set to normal weight concrete. The measurement height is set to 0.6 m and is based on Norwegian building research design guides, and represents the body center for a seated adult person (Mysen, 2017).

Table 2 Energy plus design parameters for the building simulation model.

<b>Parameter</b>	<b>Value</b>
Internal load	132 W
Occupancy	Always present
U-value of the external wall	0.16 W/m <sup>2</sup> k
U-value windows	0.80 W/m <sup>2</sup> k
G-factor glass	0.4-0.6
Infiltration rate at 50 Pa [n50]	0.6
Heating or cooling	free-running
Window ventilation set-point	>22 °C
Window ventilation fraction	50 %
Measurement height (sensor-point)	0.6 m
Thermal absorptance floor/wall/ceiling	0.9/0.9/0.9

When air velocities are low (below 0.1 m/s), which is typical for spaces inside buildings, the operative temperature can be the average value of dry bulb temperature and mean radiant temperature (Mysen, 2017). This is assumed for the building model presented in this thesis and might deviate from a real building case. Because window ventilation will affect air velocities, especially when connected to multiple adjacent zones with window ventilation, causing draft. This can be a crucial factor for cross-ventilation in natural ventilated residential buildings (Prakash and Ravikumar, 2015). It is therefore assumed that the zone is a closed volume without cross-ventilation and low outdoor wind speeds.

#### 4.1.5 Radiance parameters

Radiance design parameters for the building simulation model is set according to Table 3. The height of the sensor-nodes is set to 0.85 m for daylight calculations and is based on EN 17037:2018 and represents the work plane for a seated adult person. The indoor reflectance is set according to the same standard, and represent normal surface reflections (Norsk Standard, 2019a).

Table 3 Radiance design parameters for the building simulation model.

Parameter	Value
Reflectance floor	0.2
Reflectance wall	0.5
Reflectance ceiling	0.7
Wall thickness	0.3 m
Light transmittance glass	0.5-0.7
Measurement height (sensor-point)	0.85 m

Radiance simulation parameters used in this study are set to high precision and are shown in Table 4.

Table 4 Radiance simulation parameters.

Parameter	Value
Ambient bounces (ab)	7
Ambient divisions (ad)	2048
Ambient super-samples (as)	1024
Ambient accuracy (aa)	0.1

## 4.2 SIMULATION AND PARAMETRIC RUN

The next step is to simulate the models and run the parametric analysis, numbered “1b” in Figure 5. Processes 1a and 1b is strongly connected because Ladybug tools have a direct connection to the simulation engines. The plugin uses Radiance version 5.3 and Energy plus with OpenStudio version 3.1.0, which are validated tools (Roudsari and Smith, 2013).

### 4.2.1 Variables used for ANN

Two sets of variables are defined for the ANN model and is integrated in the simulation model. The first set is zone variables, which depends on the window geometry and window properties. Zone variables are described as glass properties (LT-value and G-value), window size and window surface orientation. The second set is sensor-node variables, which depends on specific location of the sensor-nodes. Sensor-node variables are described based on the different WWR variations. These are vertical view, horizontal view and view direction, measured in degrees. Figure 8 illustrate the different view variables. The last sensor-node variable is window distance, which is measured in meters.

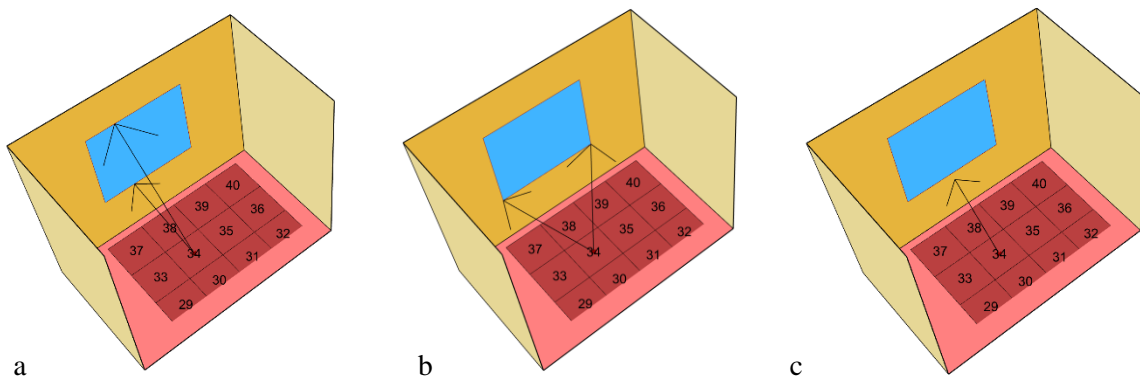


Figure 8 View variables illustrated for sensor node 34 with vertical view (a), horizontal view (b) and view direction (c).



Figure 8 Illustrate the vertical view (a), which is defined as the angle between upper and lower frame of the window, measured at the closest point on the frame. Horizontal view (b), is defined as the angle between the vertical window frames, also measured at the closest point on the frame. View direction (c) is defined as the angle between the sensor-node and the center of the lower window frame. These variables are integrated in order to have a detailed description of the window for each individual sensor-node. This helps the model differentiate between window shapes, window ratios and different angles to the window.

#### 4.2.2 Parametric model

The geometric model and associated parameters are imported into the energy simulation software Energy Plus and the lighting software Radiance. ASHRAE weather file with Oslo Fornebu IWEC climate is used for both Radiance and Energy Plus. The solution space can be found in Table 5 and consists of room orientation, glass properties and WWR, which is the most common way to describe the geometry information of the opening in the façade (Hee et al., 2015).

Table 5 Solution space of the parametric analysis.

Variable	Minimum	Maximum	Step	Number of steps
<b>Orientation</b>	0	315	45	8
<b>G-value/LT-value</b>	0.4 / 0.5	0.6 / 0.7	0.1	3
<b>WWR</b>	0.1	0.9	0.1	9
<b>Total simulations</b>				<b>216</b>

The model is executed parametrically using exhaustive search, distributing the variables continuously. This combination result in a solution space of 216 different building variations and a total of 151 million hours of operative temperature and illuminance. Figure 9 illustrates process of data generation from test-cases to hourly data results.

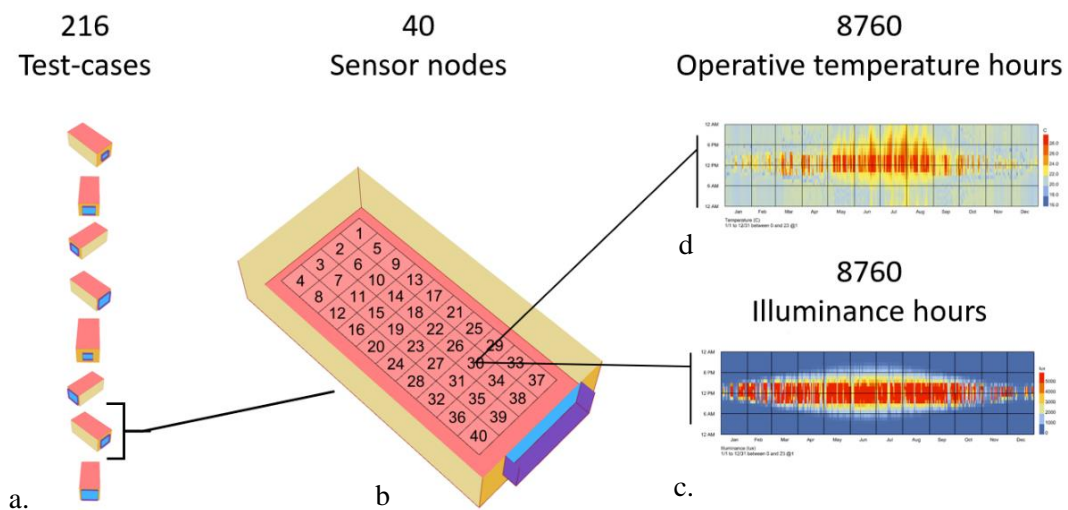


Figure 9 Illustration of the simulation process where all 216 test-cases (a) are simulated with 40 sensor-nodes (b) which generates two sets of 8760 hourly values shown as carpet plot for illuminance (c) and operative temperature (d).

The total simulation time was 4.7 hours and 6.2 hours for daylight illuminance and operative temperature respectively. A virtual machine with 2.3 GHz Intel E5 V4 with 12 cores is used. The simulation is executed in parallel where the number of sensor-nodes is divided on the total number of CPU cores, resulting in 4 sensor-nodes for each core.

### 4.3 DATA PROCESSING

After simulation and the parametric run, Illuminance and Operative temperature results are divided into three datasets, referred to “2” in Figure 5. This is to ensure that the model is tested and validated with a separate dataset. The split is based on research found in the literature and consist of 70 % training, 15 % validation, and 15 % testing. The validation dataset is a smaller subset of the data and is used for evaluation during training. The testing data, common referred to as the holdout dataset, is also a smaller subset of the data, and is used for evaluation after all the training is done. This represents the final accuracy and is used to ensure that the final model is properly generalized. This will provide the model with stable data where it is less likely to have a bias for a certain model-case. The data is shuffled before splitting, ensuring all datasets represents a mix of all variations including window size, window properties and orientation. This process is illustrated in Figure 10.

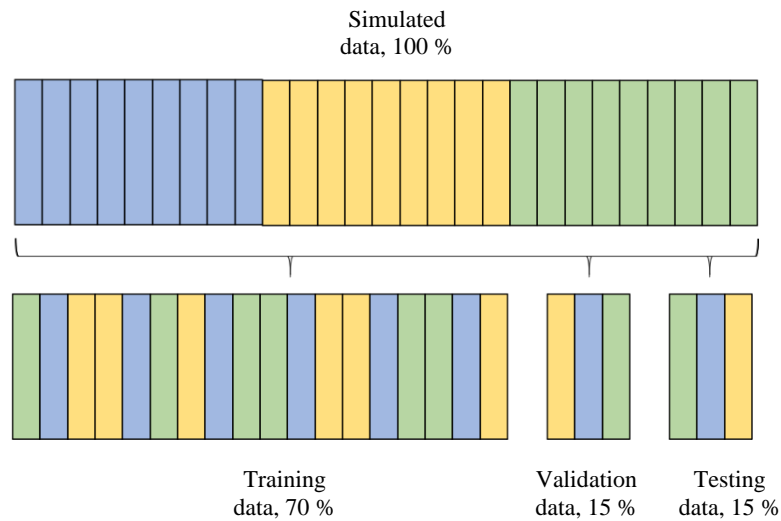


Figure 10 Illustration of data splitting and shuffling.

#### 4.4 ARTIFICIAL NEURAL NETWORK

In this step the ANN is developed and trained, illustrated as numbered “3” in Figure 11. Findings from the literature indicate that there are few tools available for conducting fully automated ANN training in combination with measured or simulated data. There are some plug-in tools available for Rhino and Grasshopper, such as LunchBoxML, Crow and Octopus, but they often lack proper detailing, or is not open-sourced. A custom machine learning algorithm is therefore developed in this study, using Python and the open source machine learning framework, Pytorch (Paszke et al., 2019).

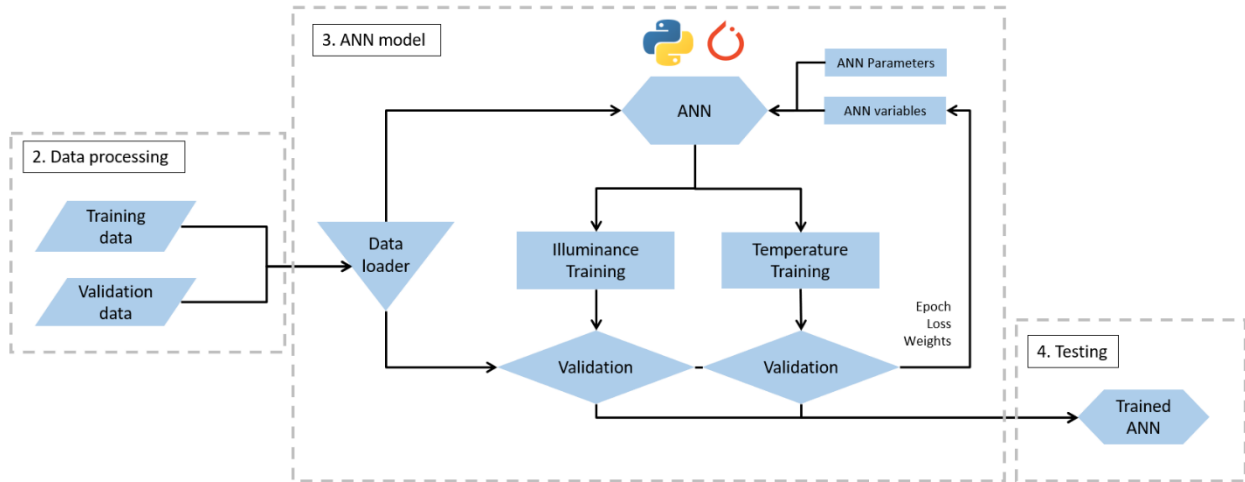


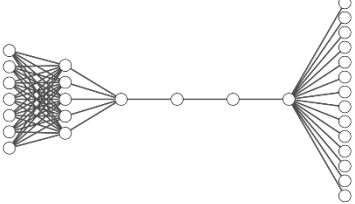
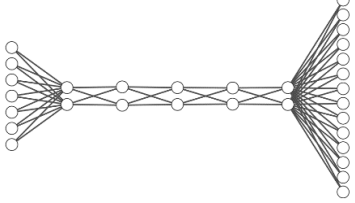
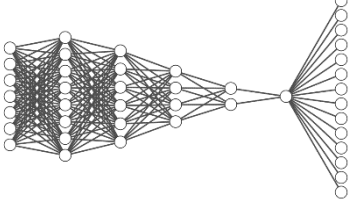
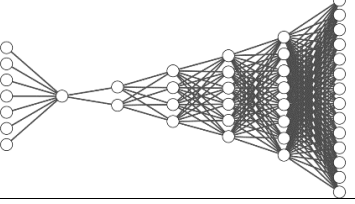
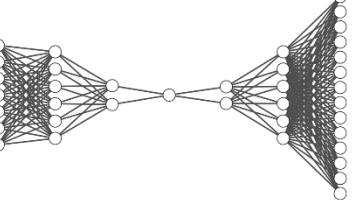
Figure 11 Flow diagram of the ANN training process.

##### 4.4.1 Model architecture and layer structure

The ANN is developed in programming language python version 3.8 and Pytorch version 1.8.1 (Paszke et al., 2019). The environment is set up with Anaconda Navigator version 1.10, which has proven to be useful when working with multiple remote virtual machines, by importing custom existing environments directly. The python ANN model code can be found in Appendix 3.

ANNs require training data to learn relationships between design parameters and corresponding daylight illuminance and operative temperature. There is some ongoing research on automating the neural architecture search, which speeds up the process of developing an ANN (Kyriakides and Margaritis, 2021). Small networks with few numbers of hidden neurons have shown good performance predicting temperature (Tran et al., 2020). Based on this research, a seven-layer fully connected ANN is used. The architecture is structured with 7 input neurons and 8760 output neurons using 5 hidden layers with 1 to 8 neurons in each layer. There are in total five model architectures with different layer structure as shown in Table 6.

Table 6 Overview of ANN architecture with ANN id, layer structure and visualization of network.

Artificial neural network id #	Layer structure (hidden layer)	Visualization of network
5-1	7-(5-1-1-1-1)-8760	
2-2	7-(2-2-2-2-2)-8760	
8-1	7-(8-6-4-2-1)-8760	
1-8	7-(1-2-4-6-8)-8760	
6-6	7-(6-2-1-2-6)-8760	

The input variables are glass properties (LT-value and G-value), window size, window surface orientation, vertical view, horizontal view, view direction and distance to the window, resulting in a total of 7 input variables. The merged results and input variables are defined as the dataloader, which is used for training and validating the ANN. The training data from the dataloader is shuffled randomly and divided into batches with size of 64 for every epoch, the process is illustrated in Figure 12. This ensures that every training is unique, which helps the ANN to converge faster and not fall into a local minimum.

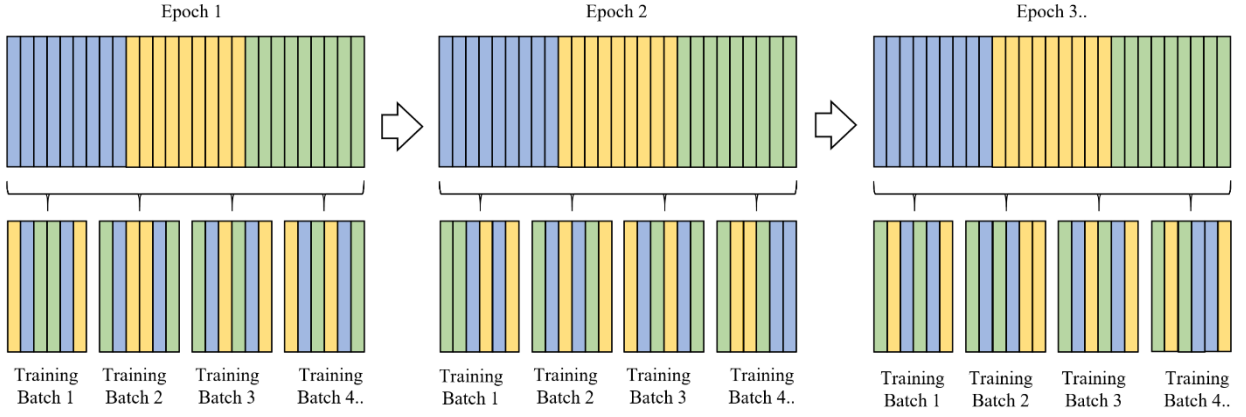


Figure 12 Illustration of data batching, where the data is shuffled and divided into batches of 64 for each epoch.

#### 4.4.2 Model activation, accuracy, and error functions

Rectified Linear Unit (ReLU) activation function is used in each layer because of the drawbacks of sigmoid and tanh when using stochastic gradient descent optimizers. The ReLU function is shown in equation (2), and is known as a ramp function where  $x$  is the input to a neuron (Shang et al., 2016). When the input is greater than zero, ReLU acts as a linear function in order to maintain the gradient stability. When the input is smaller than zero, it eliminates the output to carry on the non-linear features (Krizhevsky et al., 2017).

$$ReLU(x) = \max(0, x) \quad (2)$$

In addition to the activation function, each layer is provided with a batch normalization. This allows the model to use much higher learning rates. Batch normalization is shown in equation (3) where mean  $E(x)$  and standard-deviation  $Var(x)$  are calculated per-dimension over the mini-batches (Ioffe and Szegedy, 2015).

$$BN(x) = \frac{x - E(x)}{\sqrt{Var(x)}} \quad (3)$$

Root mean squared error (RMSE) is used as loss function while training, shown in equation (4). The error is measured between each element  $n$  in the input prediction  $y$  and target ground truth  $\hat{y}$ , and is often used when matching simulation models to measured data (Coakley et al., 2014). In this study each element consists of 8760 hourly data points, representing each hour of one year.

$$RMSE = \sqrt{\left[ \sum_{i=1}^n (y_i - \hat{y}_i)^2 \right] / n} \quad (4)$$

Coefficient of Variation of the Root Mean Square Error (CV(RMSE)) is used for validating and testing the training outputs. This indicator of error can be found in equation (5). CV(RMSE) is found to be the most robust error function for different hourly calibration datasets for building energy models

(Martínez et al., 2020). According to ASHREA the acceptance limit for CV(RMSE) is below 30% (ASHRAE, 2002). However, this acceptance limits are used for calibrating with respect to measured energy data. It is reasonable to expect the ANN validation and testing to be lower than the acceptance limit, since the input parameters are fixed. In addition, this study is using annual temperature and illuminance values which is not calibrated for the respective models.

$$CV(RMSE) = \frac{\sqrt{\sum_{i=1}^n (y_i - \hat{y}_i)^2 / n}}{E(\hat{y}_i)} \quad (5)$$

A common measure used for correlation association is Pearson's product–moment correlation coefficient (PCC) shown in equation (6), and is the ratio between the covariance of the two variables and their standard deviations. It describes the strengths of associations between variables and assumes that there is always linear relationship between the variables which might not be the case at all times (Kurtz and Mayo, 1979, Bishara and Hittner, 2012).

$$PCC = \frac{\sum_{i=1}^n [(y_i - E(y_i))(\hat{y}_i - E(\hat{y}_i))]}{\sqrt{\sum_{i=1}^n (y_i - E(y_i))^2} \sqrt{\sum_{i=1}^n (\hat{y}_i - E(\hat{y}_i))^2}} \quad (6)$$

One should be aware that it can be easily misinterpreted as a high degree of correlation from large values of the correlation coefficient does not necessarily mean very high linear relationship between the two variables (Puth et al., 2014).

#### 4.4.3 Optimizer and training parameters

With the use of RMSE, the weight and biases are optimized by implementation of stochastic gradient descent (SGD) utilizing Nesterov momentum (Sutskever et al., 2013). This is a common optimization technique for training machine learning algorithms. The goal is to adjust the weights to minimize the RMSE loss. The gradient descent algorithm starts with random model parameters and calculates the error for each learning iteration. How much the weights are adjusted each iteration depends on the scalar hyper-parameter, known as momentum and learning rate. These hyper-parameters can be adaptive, meaning they change while training. Some optimizer algorithms, such as Adaptive Momentum Estimation (Adam) does this automatically, but often generalizes significantly worse than Stochastic Gradient Descent (SGD) (Xie et al., 2020, Moreira and Fiesler, 1995). We have therefore in this study simplified the adaptive process to change with the accuracy function. Initially the learning rate is set to 0.01 and momentum is set to 0.99. While training both parameters are updated depending on the accuracy as shown in Figure 13. When accuracy increases the momentum decreases with its previous value divided by 25, and learning rate decreases with its previous value divided by 50. This allows us to give the optimizer more impact on the adjusted weight early in the training process. Momentum and learning rate are logged while training.

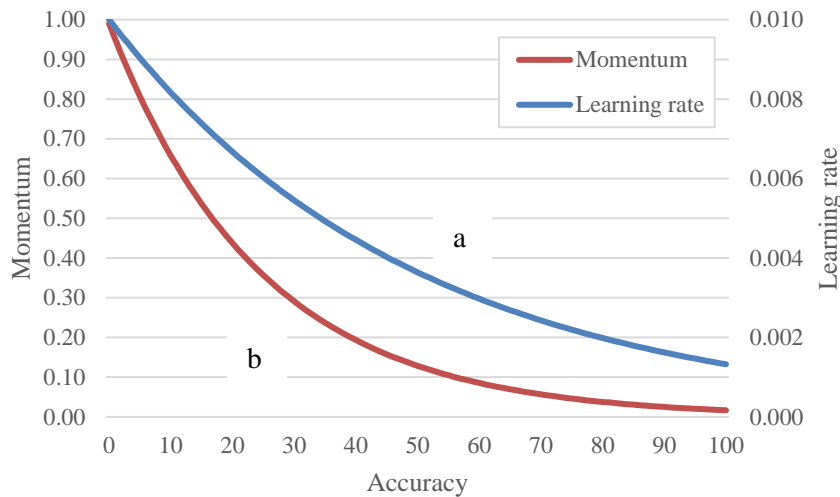


Figure 13 Adaptive parameters for learning rate (a) and momentum (b) based on the accuracy.

To avoid co-adaptation (overfitting) of neurons during the training process an early-stop mechanism is implemented. This is solved by implementing batch normalization and early-stop function. After each epoch, the algorithm measures the performance of the model over the validation data after the weights and biases are updated. If the error of the validation data grows over each epoch the training is stopped. The validation data does not intervene in the adjustment of the weights and biases during the training process. Stop mechanisms are integrated while training, if CV(RMSE) decreases to less than 1.0 %, accuracy exceeds 90 % or if the model reaches 250 epochs. For the limited budget results, 50 epochs are used.

For training the neural network a Nvidia tesla T4 tensor core GPU with 2560 CUDA cores is used. Cudatoolkit 10.2 is used for enabling CUDA utilization (Cook, 2012, Vingelmann and Fitzek, 2020).

#### 4.5 FINAL TESTING

After reaching acceptable accuracy and error limits, the trained model is tested with a hold-out dataset, shown as step number “4” in Figure 5. This dataset contains 15 % of the total data and is applied after the final network optimization update. Using a hold-out dataset assures that the model is tested with never-seen data and gives feedback on the overall accuracy. For testing, the same procedure as for the validation is conducted. Loss, CV(RMSE) and accuracy is logged.

After training and testing the model is saved using a .pth file extension. Learnable parameters such as weight and biases for the current ANN architecture are stored to disk. Each object is mapped to each layer and its parameter tensor. This allows the objects to load back to the current network architecture for either usage, further training or validation. Leveraging trained parameters by using an existing network for training, even if only a few are usable, will help the training process and hopefully help the model converge much faster than training from scratch. This can be valuable and a timesaving measure.

## 4.6 USE

Finally, after training and testing, the model can now be used. This is done by using the same Grasshopper script which is used for generating the generic models and allow us to skip the simulation step. The model objects will be loaded onto the ANN architecture in the same way as the training dataset.

Grasshopper run on IronPython, which is an open-source implementation of the Python programming language. It is also integrated with the .NET-platform, which is an open source developer platform for building many different types of cross-platform applications. IronPython uses python 2.7 which does not allow us to load the ANN model directly onto Grasshopper, since our models and libraries run on python 3.8. There are some workarounds, such as plug-in “Cpython” and “Python remote”, which allow us to integrate this directly into Grasshopper. In this study we export variables to a text file and run the python script outside Grasshopper, which then generated a new text file with the predicted hourly data. This file is then imported back to Grasshopper. These extra steps slowed down the process since the ANN needed to be activated with the respective .pth file extension for each model modification and run.



# 5 RESULTS AND DISCUSSION

In the following chapter, the generated training models and the findings from the artificial natural network is presented. The first section will address the simulated models and discuss how they impact the overall concept. This is done by comparing different sensor-node results from different test-cases, which represent a standard room geometry according to the standards which are used. In the last section we will address how the ANN performed using a limited computational budget and further investigate the best performing networks.

## 5.1 SIMULATION MODELS

### 5.1.1 Daylight illuminance results

As stated previously, one of the objectives is to deploy a simulation model, which accounts for multi-node calculations for daylight illuminance and operative temperature. Annual daylight illuminance calculations are already usually conducted with a grid of sensor-nodes for a room, which is also implemented and integrated in the workflow of Ladybug tools. However, the data is formatted in such way where only illuminance values is generated when the sun is up, usually above zero lux. Therefore, a custom script is made to read and append illuminance values with the simulated data. This is a slower process since the model need to work with a larger dataset, but it is crucial for training the neural network and comparing results. The carpet plot in Figure 14 illustrates the annual hourly illuminance values for sensor-node 18 (a) and 38 (b) in one of the simulated test-cases. The presented room parameters consist of LT-value of 0.7, WWR of 0.4 and orientation of 180 degrees (south). There is difference in illuminance when comparing the two sensor nodes where sensor node 18 and 38 have a DA(300 lux) of 17% and 40% respectively. Which indicate that sensor-node 38 receives more than 300 lux for 40 % of the occupied time.

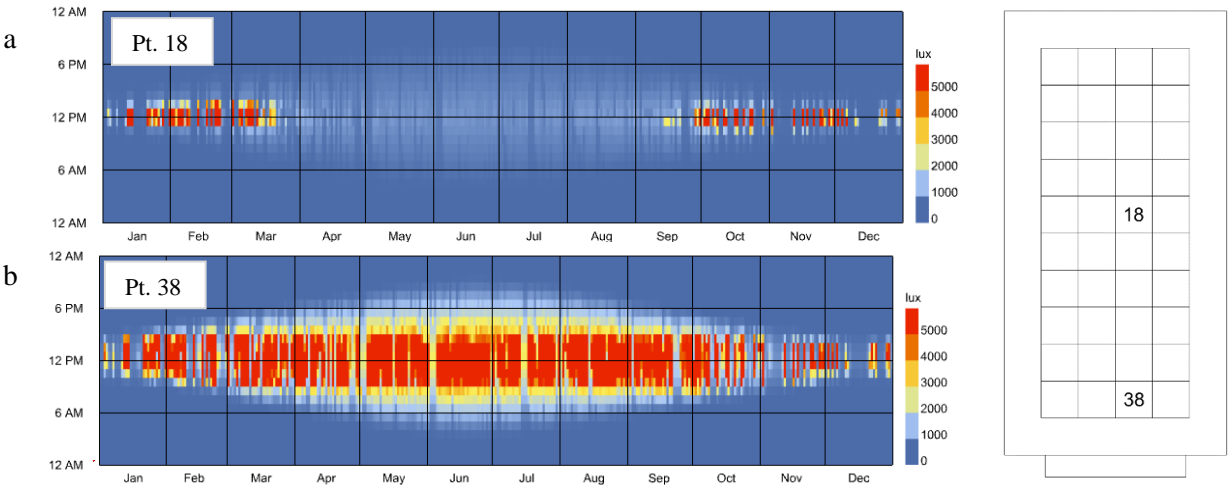


Figure 14 Carpet plot of annual hourly illuminance for sensor node 18 (a) and 38 (b).

Sensor node 18, shown in Figure 14, receives most illuminance in winter between October and March, when the sun is typically low in Oslo. Sensor node 38 receives most illuminance in summer between May and August. These are important factors to consider, because the ANN model needs to distinguish between different seasons and the effect this has on the different sensor-nodes. Carpet plot for all sensor-nodes for this test-case can be found in Appendix 4.

Another important factor to consider is directional variables for separating between sun direction throughout the day. The carpet plot in Figure 15 illustrates annual hourly fraction of direct sun exposure for sensor node 40 (a) and 38 (a). The presented room parameters are the same as above. The results show that there is a difference in direct sun exposure for different sensor-nodes with same distance from the window. Although they both receive the same amount of illuminance annually, which for these two sensor nodes are DA(300 lux) of 40%, it occurs at different hours of the day. This effect would not be considered if we only used horizontal and vertical view to the window. Because the view fraction would be the same for sensor-node 37 and 40.

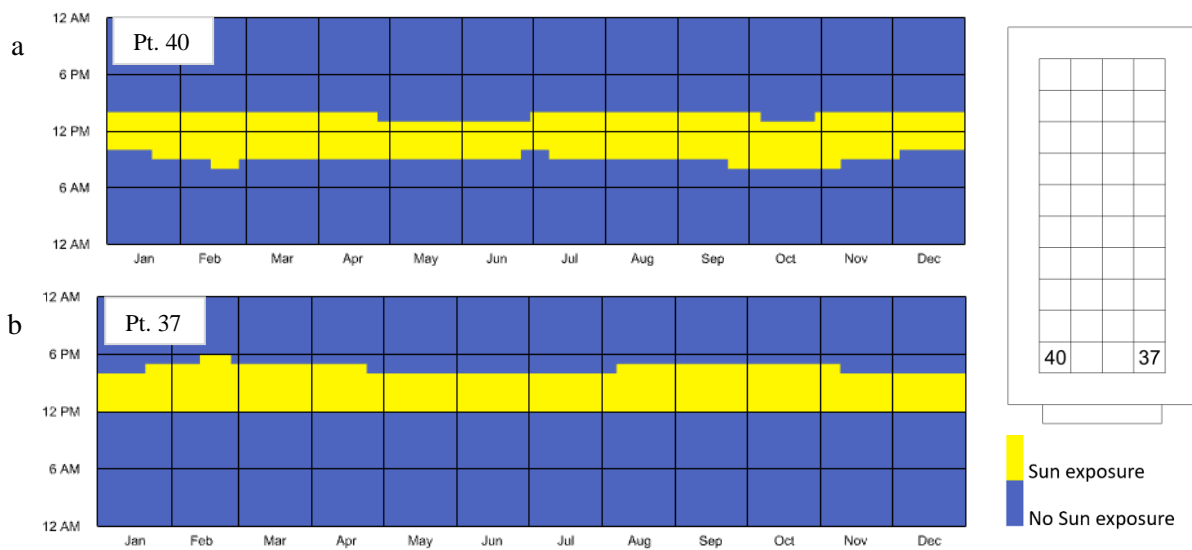


Figure 15 Illustration of difference in direct sun exposure for sensor-node 40 (a) and 37 (b).

### 5.1.2 Operative temperature results

Grid-based calculations for thermal comfort and operative temperature is not so common. To illustrate the difference, the annual operative temperature is calculated, shown in Figure 16. The calculation is based on Norwegian regulations stated in Chapter 2.2.4, where the indoor operative temperature should not exceed 26 °C when the outdoor temperature is below 23.9 °C (highest number after excluding the top 50 hours of the year). This is done for the zone center (a) and for the whole grid (b), including short- and longwave for each sensor node. The presented room parameters are the same as above and consist of g-value of 0.5, WWR of 0.4 and orientation of 180 degrees (south).

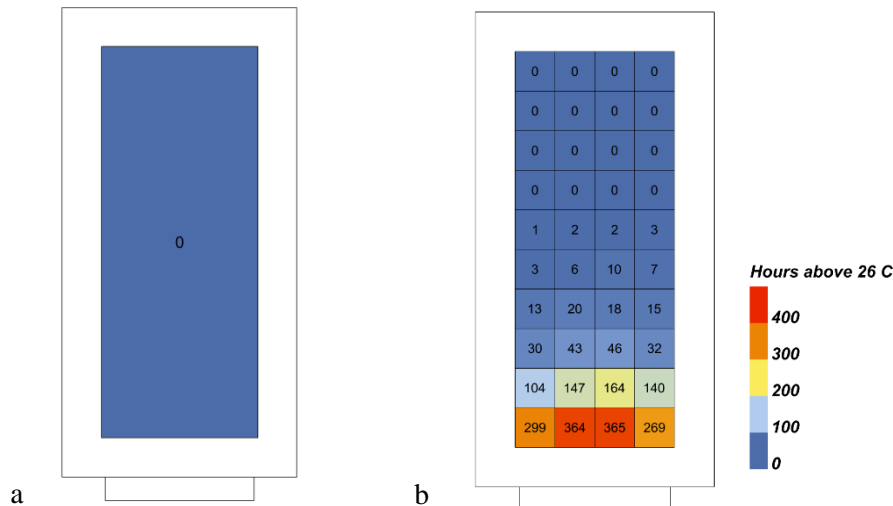


Figure 16 Number of hours with operative temperature above 26 °C when the outdoor temperature is below 23.9 °C for single node in zone center (a) and multi-node (b).

By comparing the results from Figure 16 we can see that the room to the left (a) has zero hours above 26 °C, and never excides the Norwegian building regulation. The room to the right (b) have an average of 53 hours above 26 °C and a maximum of 365 hours above 26 °C, which is the sensor-node closest to the window. The Norwegian building regulations is not necessary developed for multi-node evaluations and might use different threshold values if it were to be used. These values might be similar to DA and UDI, which is used for daylight availability, where the measurement of area within a certain limit is acceptable. It should also be noted that the measurement height of the sensor-nodes is 0.6 m and 0.85 m for operative temperature and illuminance, representing body center and work plane respectively.

This test-case and illustration shows the importance of multi-node calculations and how room evaluations, according to the regulations, might be quite different. However, this type of calculation is time and computational demanding, since we need to address heat gain from long- and shortwave for each sensor-node. The time used for calculating operative temperature was 1.8 and 51.9 seconds for single node (a) and multi-node (b) respectively. This is almost 30 times more even with parallel processing for calculations. It is therefore very interesting to look at machine learning techniques, to partially replace and reduce the time-consuming simulations methods, in order to achieve multi-objective design targets.

The carpet plot in Figure 17 illustrates the annual hourly operative temperature for sensor-node 18 (a) and 38 (b) in one of the simulated test-cases. The presented room parameters consist of g-value of 0.5, WWR of 0.4 and orientation of 180 degrees (south).

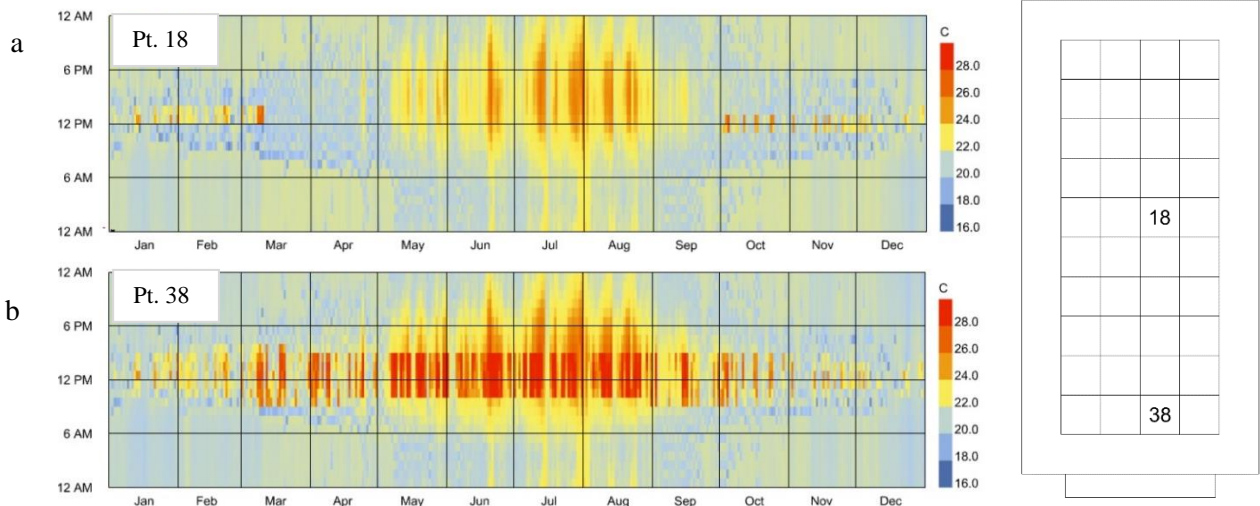


Figure 17 Carpet plot of annual hourly operative temperature for sensor node 18 (a) and 38 (b).

From the results we can clearly see that there is a difference in direct sun exposure for different sensor-nodes. Sensor node 18 (a) represents a typical zone center calculation. Using zone center as stated in standards and Figure 16a, will give a small portion of shortwave radiation, which is heat gained from direct sun exposure. The longwave radiation, which is heat gained from surfaces, had less impact on the overall performance. Carpet plot for all sensor-nodes for this test-case can be found in Appendix 4.

The carpet plot shown in Figure 18 illustrate the difference in operative temperature for sensor-node 18 and 38 for longwave (a) and shortwave (b) radiation. There is a clear heat contribution from the glass surface for the sensor-node close to the window, direct sun exposure had significant impact on the operative temperature.

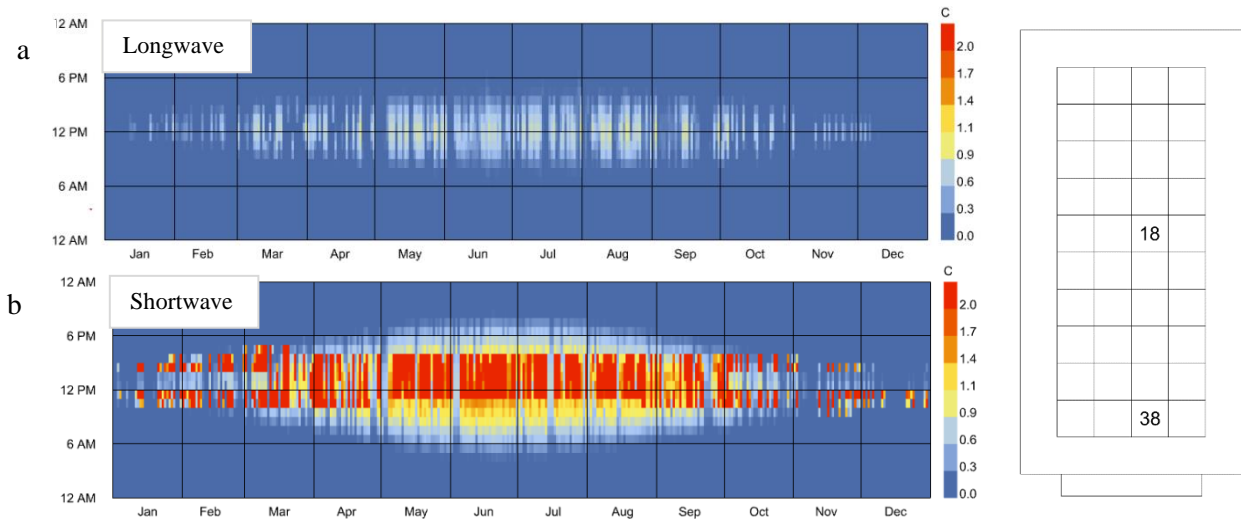


Figure 18 Carpet plot of difference in operative temperature for sensor node 18 and 38 for longwave (a) and shortwave (b) radiation.

It should also be mentioned that the longwave surface heat flux only considers one temperature for each surface per time-step. Our model consist of four wall faces, one ceiling, one floor and one window, resulting in a total of seven surface calculation. This is a good enough approximation when

most of the surfaces has a homogenic heat distribution. In some cases, it would be preferable to include a more realistic and detailed approach, where we divide each surface into smaller subsurfaces. This could for instance be done for a floor section close to the window, where a certain area is heated directly by the sun, or a window where vegetation context is shading a part of it. Surface heat flux calculations is where the majority of the computational time is spent. The same consideration is set for shortwave surface heat flux where it is assumed that the whole body is irradiated, if the sensor-node is irradiated. This could also be considered as a whole-body model with different absorbance for different clothing situations, or if the model is partly shaded/exposed to direct sun. Furniture could also have significant impact on the results because it can obstruct direct sun, absorb heat and reflect heat.

## 5.2 OPERATIVE TEMPERATURE ANN MODEL

### 5.2.1 Limited compactional budget

The ANN is first trained on the operative temperature dataset using limited budget according to chapter 4.4. The compactional budget is reduced to a maximum of 50 training epochs, where all networks are evaluated. The error and accuracy performance based on the validation data of each network is illustrated in Figure 19.

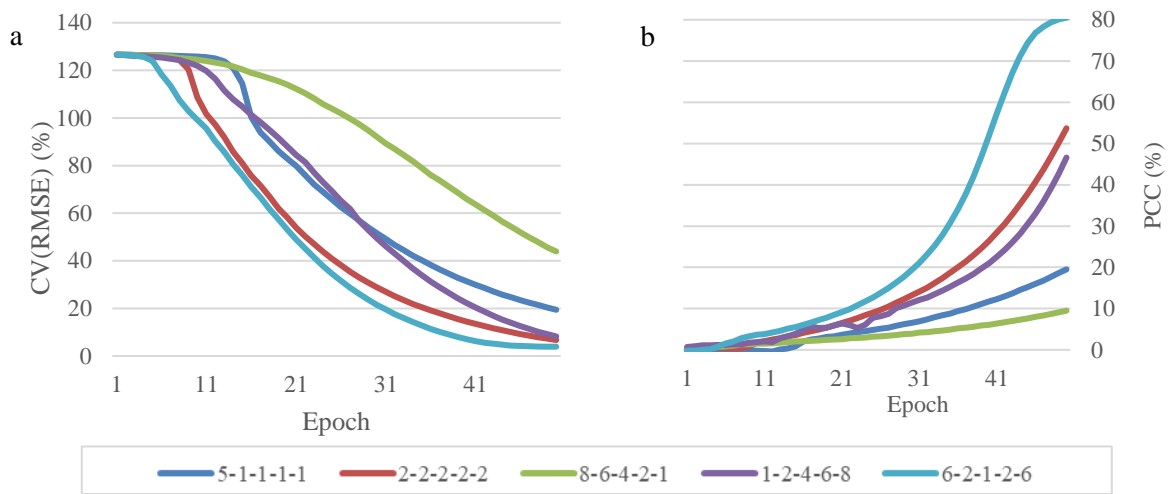


Figure 19 Operative temperature ANN model performance with Error (a) and accuracy (b) performance after 50 epochs for all networks.

As we can see in Figure 19 all networks have a reduction in error after 15 epochs of training. Network 5-1 and 8-1 are the slowest to reduce initial error and network 6-6 is the fastest. Network 8-1 is the overall slowest. The final accuracy and error score can be found in Table 7. Based on the results, the best performing network 6-6 is used for further training and evaluation. This does not necessarily mean network 6-6 have best performance when training with unlimited compactional budget, because all networks might learn better over time and the final adjusted weights and loss might be more accurate. We can also observe all networks have a stable reduction in error and increasingly accuracy over time, which indicate the possibility where all networks can reach acceptable accuracy.

Table 7 Operative temperature error and accuracy performance for ANN models after 50 epochs.

Artificial neural network id #	5-1	2-2	8-1	1-8	6-6
CV(RMSE) (%)	19.4	6.7	43.9	8.2	3.9
PCC (%)	19.5	53.7	9.5	46.6	80.5

5.2.2 Network 6-6 results

After the initial testing with limited budget the best model is trained further and stopped if CV(RMSE) fall under 1.0 % or PCC exceeded 90 % or epoch reached 250. The new ANN model is trained with the pre-trained network where the initialized weight is already adjusted. Figure 20 show the error and accuracy for network 6-6. The model stopped after 100 epochs with a CV(RMSE) of 3.8 % and PCC of 82 %. The momentum is also included in the plot illustrating how it adapts while accuracy is increasing, the learning rate has a similar plot. We can also observe that the slope of the momentum is stable even when the accuracy is changing drastically, seen at epoch 45. This is because the momentum and learning rate is evaluated and adjusted for each epoch depended on the accuracy.

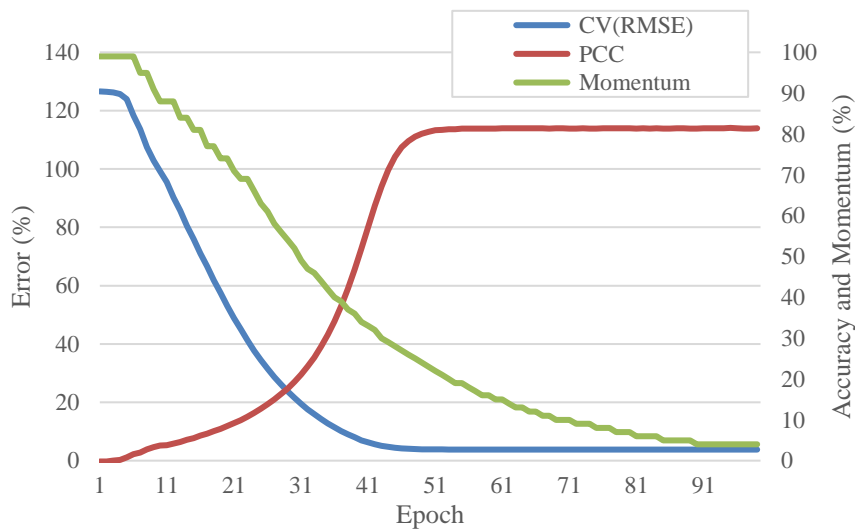


Figure 20 Network 6-6 error, momentum and accuracy performance while training.

Important factors to consider when making predictions are model error and accuracy. As stated previously, in Chapter 4.4.2, the ASHREA acceptance limit for CV(RMSE) is 30 %. This is used as limit for validating a simulation model with real building energy data. There are several limitations considered when validation real building data. For instance, the behavior of the occupants might deviate from the simulated data. This can be temperature set-points, different window ventilation and activity level etc. It is therefore reasonable to use lower acceptance limit for the ANN model, because the ANN model is validated against simulated data, which have fixed input for occupancy behavior.

5.2.3 Qualitative results

Based on the training presented in Figure 20, the test-cases which yielded greatest error is further investigated. The ANN model had difficulties with test-case g-value of 0.4, WWR of 10 % and oriented 270 degrees (West). The annual hourly operative temperature, for sensor-node 38, is presented in Figure 21. A more detailed view of the summer is presented Figure 22, the time slice is illustrated as a blue box on the figure.

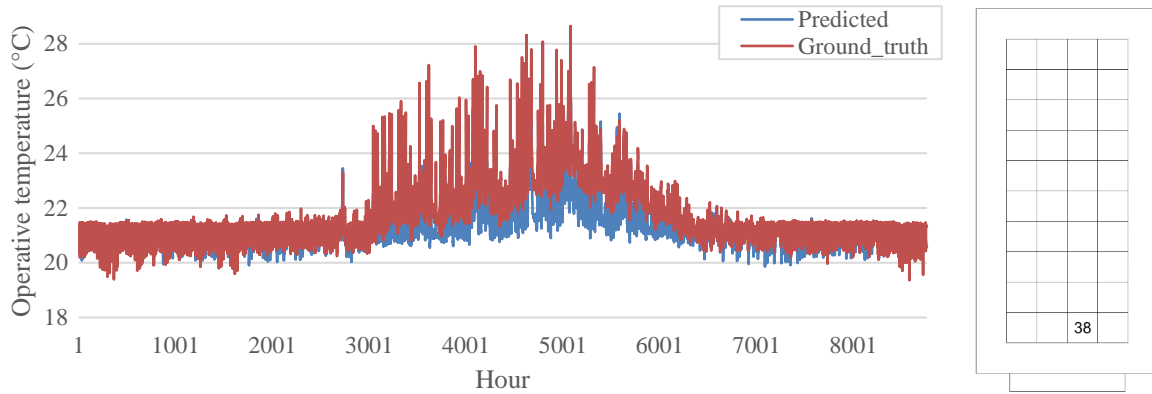


Figure 21 Network 6-6 predictions for annual hourly operative temperature and simulated results for sensor-node 38 (closest to window) in test-case g-value of 0.4, WWR of 10 % and oriented 270 degrees (West).

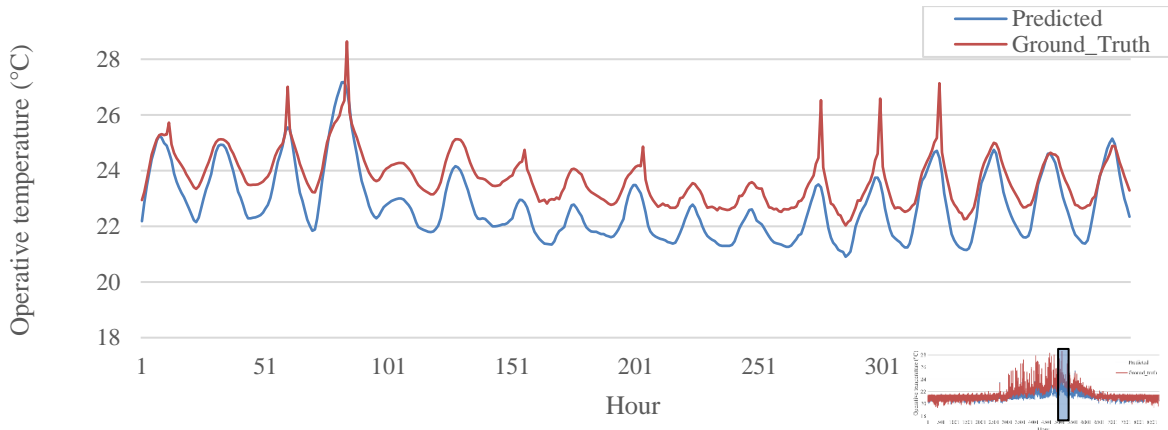


Figure 22 Network 6-6 predictions for hourly operative temperature between 5000 and 5400 hours (summer) and simulated results for sensor-node 38 (closest to window) in test-case g-value of 0.4, WWR of 10 % and oriented 270 degrees (West).

As we can see in Figure 22 the ANN model underestimates the temperatures in hot summer days. The overall model is precise but not accurate, meaning it is following the same temperature pattern, but is consequently predicting lower temperatures. There is also a mismatch for peak heat loads or where the temperature is changing rapidly. The overall average prediction error is 0.47 °C or 2.1 %, resulting in an accuracy of 98 %. The maximum operative temperature recorded is 27.2 °C and 28.6 °C for prediction and ground truth respectively. The minimum operative temperature recorded is 19.9 °C and 19.2 °C for prediction and ground truth respectively.

### 5.3 DAYLIGHT ANN MODEL

#### 5.3.1 Limited compactional budget

The second ANN is trained with the daylight illuminance dataset using limited budget according to chapter 4.4. The compactional budget is first reduced to maximum 50 training epoch. The error and accuracy performance based on the validation data of each network is illustrated in Figure 23.

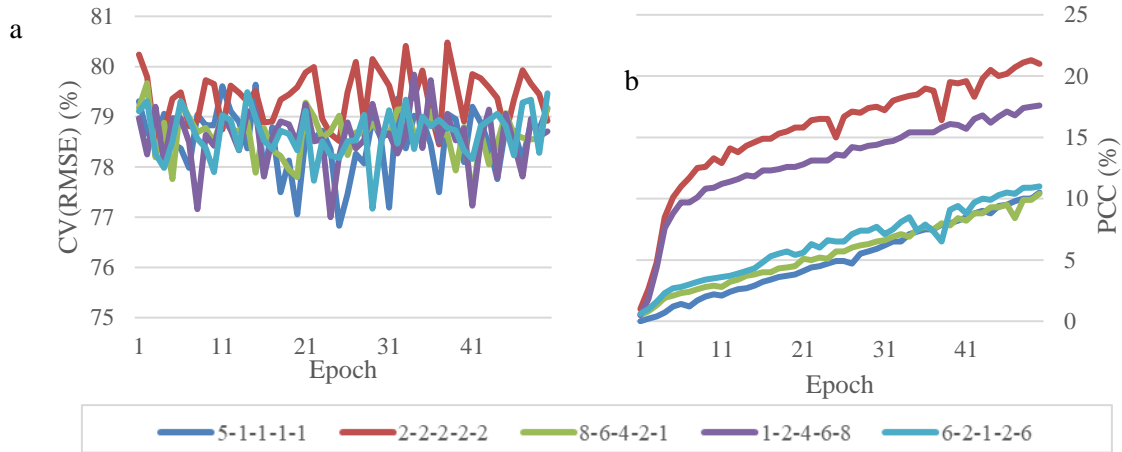


Figure 23 Daylight ANN model performance with Error (a) and accuracy (b) performance after 50 epochs for all networks.

In contrast to predicting operative temperature the illuminance values converged much slower. The error is not declining, and accuracy is slowly increasing after 50 epochs, as shown in Figure 23. Network 2-2 and 1-8 are the fastest to increase initial accuracy and network 8-1 is the slowest. The final accuracy and error score are found in Table 8. Based on the results, the best performing network 2-2 is used for further training and evaluation. It can be observed that all networks have no significant reduction in error.

Table 8 Illuminance error and accuracy performance for ANN models after 50 epochs.

Artificial neural network id #	5-1	2-2	8-1	1-8	6-6
CV(RMSE) (%)	79.2	78.9	79.2	78.7	79.5
PCC (%)	10.5	21	10.4	17.6	11.0

#### 5.3.2 Network 2-2 results

Because of high initial error the maximum epoch limitation is increased to 1500 epoch. The training stopped if CV(RMSE) decreased to less than 1.0 % or PCC exceeded 90 %. The new ANN model is trained with the pre-trained network where the initialized weight is already adjusted. Figure 24 show the error and accuracy for network 2-2. The model stopped after 1500 epochs with a CV(RMSE) of 61 %, PCC of 63 % and loss of 2541. We can observe that accuracy is increasing up to 50 % after 500 epochs and flattens to around 60 % the next 1000 epochs. The error on the other hand is stable and starts to decline after 500 epochs. There is no significant correlation between the error and accuracy like it is for operative temperature seen in Figure 20.



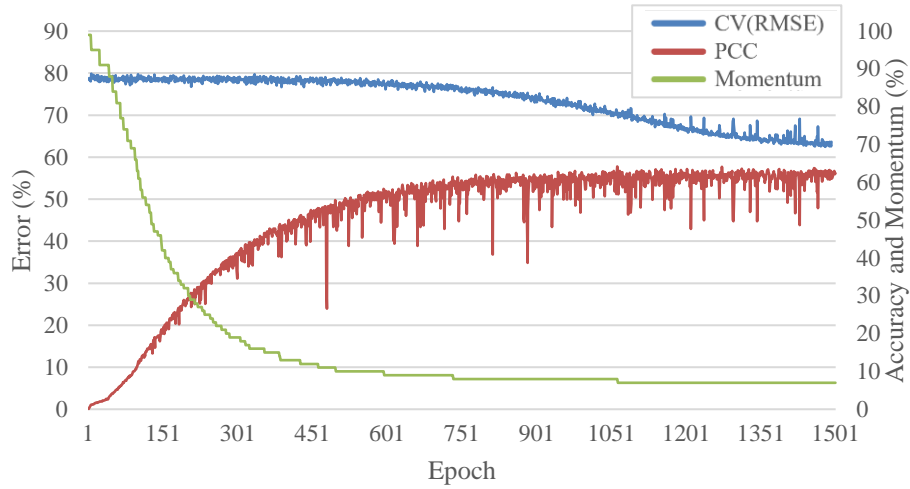


Figure 24 Network 2-2 error, momentum and accuracy performance while training.

### 5.3.3 Qualitative results

The test-cases which yielded greatest loss is further investigated based on the observations from the training. Test-case LT-value of 0.7, WWR of 50 % and oriented 180 degrees (South) with sensor node 18 (a) and 38 (b) is presented in Figure 25.

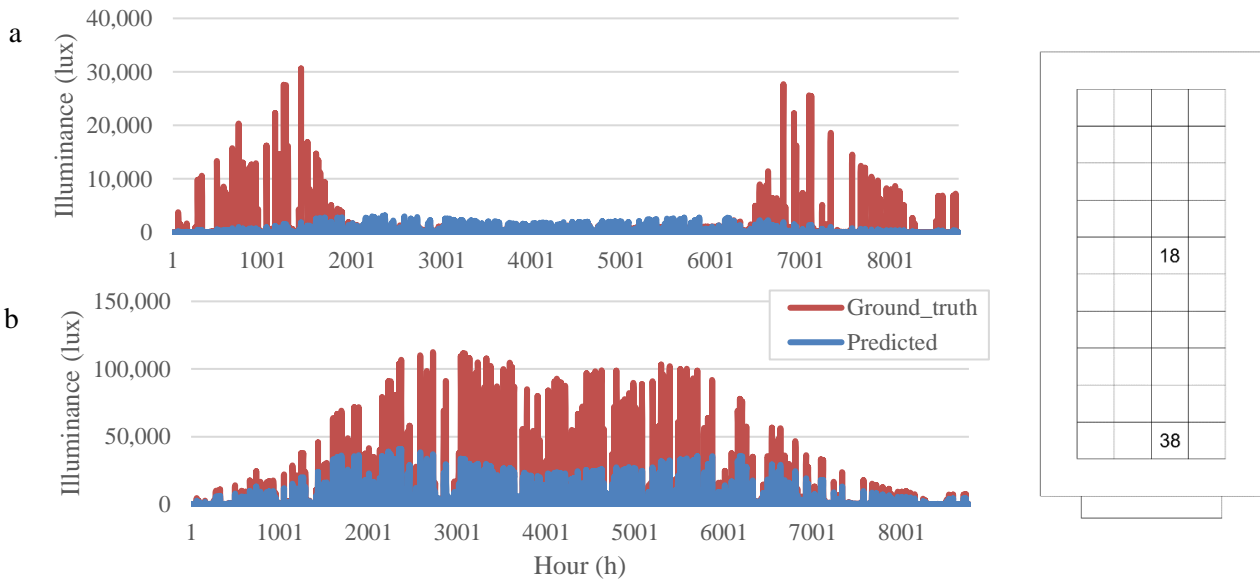


Figure 25 Network 2-2 predictions for annual hourly illuminance and simulated results for sensor-node 18 (a) and 38 (b) in test-case LT-value of 0.7, WWR of 50 % and oriented 180 degrees (South).

The predicted illuminance values are generally lower than the simulated results as seen for sensor-node 18 (a) and 38 (b) in Figure 25. This can indicate that the ANN model is generalizing the data by reducing most of the extreme values because of the MSE Loss function. This function is squaring the error for each element, making a higher error having more impact on the optimizer, which is reducing the overall loss while training. This issue was also addressed in the literature study by Tran et al. for using RMSE. It could be interesting to implement mean absolute error (MAE), which weight all the error equal while training. This might improve the overall precision but will most likely slow down the training process. Implementing both MSE and MAE at different training stages would also be very

interesting. Initially MSE could outline the overall predictions and MAE could be used for fine-tuning, making the training fast and predictions precise.

It can also be observed that the sensor-node 18 (a), which is more shaded, is having difficulties with predicting large illuminance values in the winter season. This can be due to the implementation of view direction, which is defined as the view to the window measured in degrees. When this data is normalized, the small difference between two sensor-nodes in the same room might not be sufficient for the ANN model. This will result in giving the view direction variable less impact on the predictions while training. This could be solved by normalize the view direction for each test-case, ranging the view from 0 to 1, instead of the actual orientation.

We can also see the predictions for sensor-node 38 (b) is following the same pattern as the simulated result, but is much smaller. Figure 26 show a more detailed plot of illuminance for sensor-node 18 (a) and 38 (b). The time slice is illustrated as a blue box on the figure to the right below.

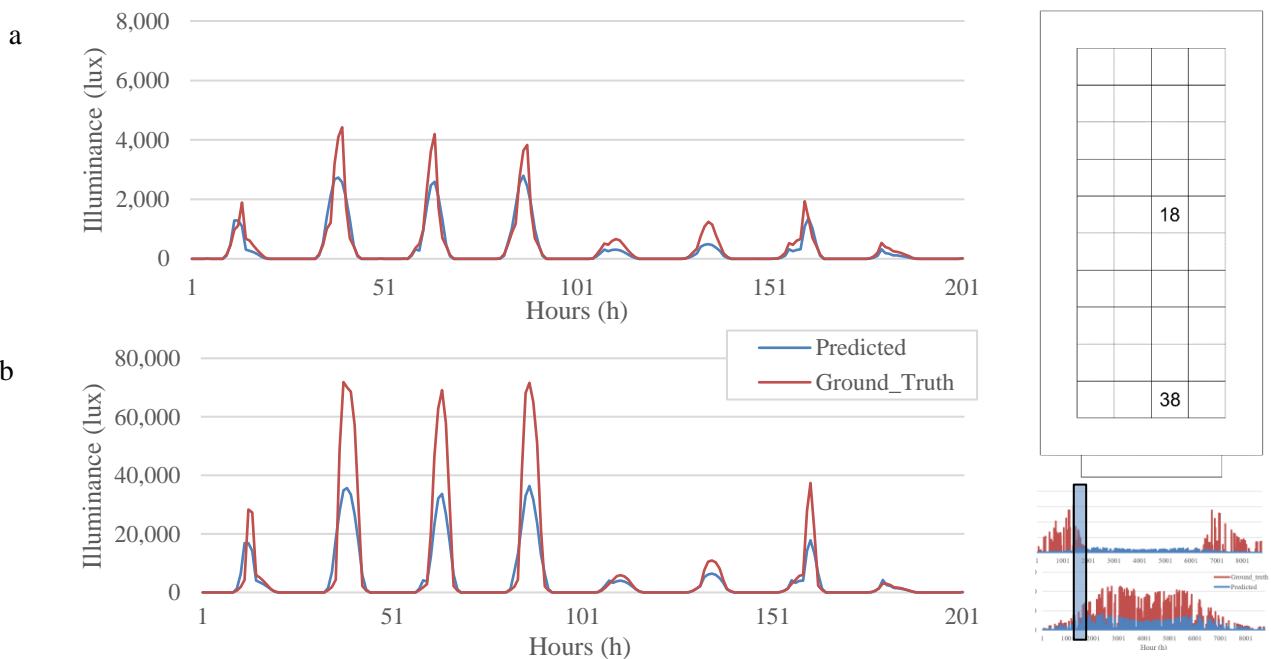


Figure 26 Network 2-2 predictions for hourly illuminance and simulated results between 1800 and 2000 hours (summer) for sensor-node 18 (a) and 38 (b) in test-case LT-value of 0.7, WWR of 50 % and oriented 180 degrees (South).

The daylight ANN model underestimates the illuminance and the same patterns as for operative temperature is observed. The model is precise but not accurate. As we also can see in Figure 26, the model is predicting low illuminance values very well. The overall average prediction for this specific test-case is within 300 lux and 3400 lux and PCC of 39 % and 83 % for sensor-node 18 and 38 respectively. This is somewhat counterintuitive because sensor-node 18 has smaller error margin, but much lower PCC accuracy. This can be caused by correlation from large values, which does not necessarily mean very high linear relationship between the two variables, which is stated in Chapter

4.4.3 and addressed by Puth et al. It might therefore be more suitable to use mean absolute accuracy, by calculating the difference in value for each time-step.

## 5.4 GENERAL ANN ASPECTS

### 5.4.1 Simulation time comparison

The average time used for activating the ANN models and generate annual results is 4.17 seconds, for 40 sensor-nodes. This includes reading variable file, separate data into chunks, activate ANN model, run variables, separate data, save data into files. The average time used for simulating daylight illuminance and operative temperature is 47.8 and 63.1 seconds respectively, using parallel processing. The same test is conducted for 1000 sensor-nodes, where the ANN model used 34.4 seconds, illuminance simulation used 282.8 seconds and operative temperature simulation used 1206.5 seconds. Table 9 show the computational time for each tested setup. For simulating, a virtual machine with 2.3 GHz Intel E5 V4 with 12 cores is used. The overall time is reduced by 96 % by using ANN models.

Table 9 Time spent on simulations and running ANN model.

<b>Model</b>	<b>Time used for 40 sensor-nodes (seconds/node)</b>	<b>Time used for 1000 sensor-nodes (seconds/node)</b>
<b>ANN</b>	0.10	0.03
<b>Daylight illuminance simulation</b>	1.20	0.28
<b>Operative temperature simulation</b>	1.60	1.20

### 5.4.2 Model comparison

The operative temperature ANN model performed overall best for predicting annual values. Shortwave radiation and sudden gain in illuminance are difficult to predict. These variations are often underestimated for both ANN models. This was also highlighted in the literature by Magnier and Haghghat, where ANN models performed very well in terms of predicting energy use, but generally underestimate the thermal comfort. Both models shared the same underlying algorithm, data-set and hyper-parameters. The only difference is the number of epochs and ANN structure. The illuminance model trained for 1500 epochs and consist of a 2-2 structure. The operative temperature model trained for 250 epochs and consist of a 6-6 structure. It could be argued that these two metricizes should have different hyper-parameters as well, because the illuminance model depends more on direct sun exposure, where the operative temperature model depends on the overall heat gain and heat loss.

A reason the operative temperature model performed better might be due to the annual and daily cycles. The hourly change in operative temperature is average 1.3 % and maximum 15 %. The hourly change in illuminance is average 63 % and maximum 3737 %. This indicate how different the hourly simulated data result actual is and will have impact on the predictions. Operative temperature is much more stable due to heat balance and thermal storage, which does not give much temperature fluctuations. Whereby illuminance rely on direct sun exposure and reflections, which will fluctuate

more. This was also addressed by Ngarambe et al. as factors affecting daylight indoor illuminance from the literature study.

### 5.4.3 ANN aspects

Another approach for training the ANN model could be to merge both the operative temperature and the illuminance data set. Making the ANN model output two datasets of annual predictions for every input. Research found in the literature study usually don't have complex output predictions, usually these are single output variables using PMV, UDI or DA metrics for one zone. In contrast to this study where we output over 350 000 hours of data (40 sensor-nodes with 8760 hours each) for one zone, see Figure 9 for the data generation. Tran et al. suggested smaller network architectures for simple outputs. It might therefore be necessary to increase the number of hidden layers and neurons in the ANN model for more accurate predictions. There is some ongoing research by Kyriakides and Margaritis on neural network search which will speed up the process of finding suitable network architecture.

Initially the model was trained without the view direction variable. This resulted in difficulties for the ANN model to distinguish between which side of the room the sensor-node is located, because the view fraction and distance from the window is the same. There is no evidence for which variables that had most impact on the overall results. The presented results indicate that sensor-nodes further from the window had difficulties to predict fluctuations in illuminance values in the winter season.

The models are also tested with various variations of ReLU (leaky ReLU and ELU) but did not have any significantly impact on the results. It would be interesting to test the model with different activation functions such as Tanh and Sigmoid. These functions usually arrange the weights between negative 1, 0 and 1, and could be useful for predicting illuminance and operative temperature because of the daily and annual cycle of these values. The batch split was conducted with random shuffling before splitting into training, validation and testing data sets. It would be preferable to split the data set into segments of 40 before splitting, representing one building simulation test-case with 40 sensor-nodes. This is to ensure every dataset represent one simulation model, and not randomly sensor-nodes from every test-case. This will however not impact the end results significantly because the generated result will be correlated to each other.

The adaptive momentum and learning rate had great impact on the performance while training, helping the model to reduce loss rapidly initially. There is however a potential pitfall where the model can be stuck in a local minimum, if the momentum and learning rate is reduced to fast. Re-running the ANN models resulted in different initial accuracy and error loss from time to time. This suggest that shuffling data for every epoch is an effective measure to help the model to converge and not fall into a local minimum.

## 6 CONCLUSION

---

This thesis investigates the potential for applying ANNs to predict both annual daylight illuminance and operative temperature. The use of ANNs promises great support and improved feasibility to BPS due to a reduction in overall computation time. However, few tools are available for conducting fully automated ANN training in combination with measured or simulated data.

### 6.1 MAIN FINDINGS

The main findings from deploying a simulation model is the importance of multi sensor-node calculations for both daylight illuminance and operative temperature. Daylight illuminance is commonly addressed by calculating a grid of sensor-nodes, representing light levels for the whole room. Operative temperature is in current practice calculated for the room center. In this study operative temperature is calculated for a grid of sensor-nodes, including long- and shortwave radiation. The results show a significant difference in operative temperature at different locations in the room where shortwave radiation has greatest impact on the results. It is therefore important to address operative temperature with a grid of sensor-nodes when exploring multi-objective optimization performance. However, these calculations are computational demanding and increase simulation time by 2000 %. It is therefore interesting to look at machine learning techniques to partially replace and reduce the time-consuming simulations methods in order to achieve multi-objective design targets.

Promising machine learning approaches from the literature are implemented and evaluated for performance. The operative temperature ANN model with 6-2-1-2-6-layer structure performed overall best for predicting annual values reaching a CV(RMSE) of 3.8 %, with an accuracy of 98 %, and average prediction error within 0.47 °C after 250 epochs. The daylight ANN model with 2-2-2-2-2-layer structure reached a CV(RMSE) of 61 % with an accuracy of 19 % after 1500 epochs. Direct sun exposure is difficult to predict and the ANN model often underestimates these variations. The overall model is precise but not accurate, meaning it is following the same pattern, but is consequently predicting lower temperature and illuminance values.

In general, the operative temperature ANN model is showing promising results which may be integrated in a multi-objective design workflow. The results show significant time saving potential by implementing ANNs. The overall time is reduced by 96 % by using ANN models for predicting annual temperature and illuminance values. Choosing a more complex layer and neuron structure might help the model to generalize the data resulting in increased accuracy and better predictions for shortwave radiation.

## 6.2 FURTHER WORK

There are many ideas and methodologies that would have contributed to enhancing the scope of this thesis but due to time and technical limitations they were not possible to implement. Here are some recommendations for further research in this field:

1. In this thesis a fully connected neural network is developed. It would be interesting to compare the results with different algorithms such as Levenberg-Marquardt algorithm and Recurrent Neural Network. These models have proven to be good at generalizing complex data structures and might perform well for annual hourly predictions. The model presented in this thesis is tested with variations of ReLU. Different activation function such as sigmoid and hyperbolic tangent should also be addressed in order to allow the model to learn different non-linear functions.
2. In addition to different algorithms, different layer structures and hyper-parameters should also be addressed. The model structure in this thesis consist of 5 hidden layers with 1-8 neurons in each layer. More complex structures should be tested and compared since the output layer consist of 8760 nodes which might be complex for such a small network used in this thesis. Different batch sizes for training, validation and testing can impact the training results. Also, different batch size while training might also impact the overall results.
3. Seven input variables are used for training and consist of LT/G-value, horizontal view, vertical view, view direction, distance, orientation and window size. For more complex calculations additional variables should be introduced, this can be window ratio, window height, reflections on different surfaces, different weather and climate data, size of room, shape of room, room height and occupancy schedules. Different simulation scenarios with additional windows can also be introduced. Usually the ANNs are developed for fixed number of input variables, which would make variable number of windows more complex to model. There are some workarounds which can be addressed.
4. Both daylight and operative temperature are highly connected to sky and direct sun exposure and therefore window design. It would be interesting to investigate the topic further by comparing sensor-node results and finding correlation between each variable. This could also be developed into a common integrated ANN model, which uses the correlated variables to predict both comfort metrics simultaneously.
5. Façade analysis for thermal radiation and daylight availability such as vertical sky component (VSC) have been more popular in recent years. These analyses are often calculated in early phase of building projects. It would be interesting to use these analysis results as an input variable for the ANN model. This would make the ANN model account for different weather locations and shading context, without needing to train the model for these specific locations.

## 7 REFERENCES

---

- ANDERSEN, P. A., DUER, K., FOLDBJERG, P., ROY, N., CHRISTOFFERSEN, J., ASMUSSEN, T. F., ANDERSEN, K., PLESNER, C., RASMUSSEN, M. H. & HANSEN, F. 2014. *Daylight, Energy and Indoor Climate Basic Book*, VELUX
- ARBEIDSTILSYNET 2016. Klima og luftkvalitet på arbeidsplassen. <https://www.urd-klima.no/getfile.php/132158-1574004907/Demo/Filer/luftkvalitet-pa-arbeidsplassen.pdf>
- ASHRAE 2001. ANSI/ASHRAE Standard 140-2001. *Standard Method of Test for the Evaluation of Building Energy Analysis Computer Programs*. American National Standards Institute and American Society of Heating Refrigerating and Air-Conditioning Engineers. [https://www.ashrae.org/file%20library/technical%20resources/standards%20and%20guidelines/standards%20addenda/140-2001\\_addendum-a.pdf](https://www.ashrae.org/file%20library/technical%20resources/standards%20and%20guidelines/standards%20addenda/140-2001_addendum-a.pdf)
- ASHRAE 2002. ASHRAE Guideline 14-2002. *Measurement of Energy and Demand Savings*. American Society of Heating Refrigerating and Air-Conditioning Engineers. [http://www.eepformance.org/uploads/8/6/5/0/8650231/ashrae\\_guideline\\_14-2002\\_measurement\\_of\\_energy\\_and\\_demand\\_saving.pdf](http://www.eepformance.org/uploads/8/6/5/0/8650231/ashrae_guideline_14-2002_measurement_of_energy_and_demand_saving.pdf)
- ASHRAE 2017. ANSI/ASHRAE Standard 55-2017. *Thermal Environmental Conditions for Human Occupancy*. American National Standards Institute and American Society of Heating Refrigerating and Air-Conditioning Engineers.
- BAILEY, G. B. & NICKLAS, M. H. 1996. Analysis of the Performance of Students in Daylit Schools. <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.962.1676&rep=rep1&type=pdf>
- BISHARA, A. J. & HITTNER, J. B. 2012. Testing the significance of a correlation with nonnormal data: Comparison of Pearson, Spearman, transformation, and resampling approaches. *Psychological Methods*, 17, 399-417.
- BRAGER, G. S. & DE DEAR, R. J. 1998. Thermal adaptation in the built environment: a literature review. *Energy and Buildings*, 27, 83-96. <http://www.sciencedirect.com/science/article/pii/S0378778897000534>
- BRE, F., ROMAN, N. & FACHINOTTI, V. D. 2020. An efficient metamodel-based method to carry out multi-objective building performance optimizations. *Energy and Buildings*, 206, 109576. <https://www.sciencedirect.com/science/article/pii/S0378778819323047>
- CARLUCCI, S., CAUSONE, F., DE ROSA, F. & PAGLIANO, L. 2015. A review of indices for assessing visual comfort with a view to their use in optimization processes to support building integrated design. *Renewable and Sustainable Energy Reviews*, 47, 1016-1033. <http://www.sciencedirect.com/science/article/pii/S1364032115002154>
- CHAI, Q., WANG, H., ZHAI, Y. & YANG, L. 2020. Using machine learning algorithms to predict occupants' thermal comfort in naturally ventilated residential buildings. *Energy and Buildings*, 217, 109937. <http://www.sciencedirect.com/science/article/pii/S0378778819314653>
- CHEN, X. & YANG, H. 2015. Combined thermal and daylight analysis of a typical public rental housing development to fulfil green building guidance in Hong Kong. *Energy and Buildings*, 108, 420-432. <https://www.scopus.com/inward/record.uri?eid=2-s2.0-84944203850&doi=10.1016%2fj.enbuild.2015.09.032&partnerID=40&md5=c8ba534e670e053893d9ca6d8c450881>
- CIE 1996. CIE S003 Spatial distribution of daylight - CIE standard overcast sky and clear sky. *CIE S003:1996*. International Commission on Illumination.
- CIE 2006. Test cases to assess the accuracy of lighting computer programs. *CIE 171:2006*. CIE.
- CLARKE, J. A. & HENSEN, J. L. M. 2015. Integrated building performance simulation: Progress, prospects and requirements. *Building and Environment*, 91, 294-306. <https://www.sciencedirect.com/science/article/pii/S0360132315001602>
- COAKLEY, D., RAFTERY, P. & KEANE, M. 2014. A review of methods to match building energy simulation models to measured data. *Renewable and Sustainable Energy Reviews*, 37, 123-141. <https://www.sciencedirect.com/science/article/pii/S1364032114003232>

- COOK, S. 2012. *CUDA Programming: A Developer's Guide to Parallel Computing with GPUs*, Saint Louis, Saint Louis: Elsevier Science & Technology.
- DA SILVA, I. N., HERNANE SPATTI, D., ANDRADE FLAUZINO, R., LIBONI, L. H. B. & DOS REIS ALVES, S. F. 2016. *Artificial Neural Networks: A Practical Course*, Cham, Cham: Springer International Publishing.
- DAVID, M., DONN, M., GARDE, F. & LENOIR, A. 2011. Assessment of the thermal and visual efficiency of solar shades. *Building and Environment*, 46, 1489-1496.  
<http://www.sciencedirect.com/science/article/pii/S0360132311000321>
- EPBD 2010. Directive 2010/31/EU of the European Parliament and of the Council of 19 May 2010 on the energy performance of buildings (recast). Brussels. <http://data.europa.eu/eli/dir/2010/31/oj>
- FANGER, P. O. 1970. *Thermal comfort. Analysis and applications in environmental engineering*, Copenhagen: Danish Technical Press.
- GAO, X., SHI, M., SONG, X., ZHANG, C. & ZHANG, H. 2019. Recurrent neural networks for real-time prediction of TBM operating parameters. *Automation in Construction*, 15, 130-140.
- GOODFELLOW, I., BENGIO, Y. & COURVILLE, A. 2016. Deep learning. *Genetic programming and evolvable machines*. <http://www.deeplearningbook.org>
- HAASE, M. & GRYNNING, S. 2017. Optimized facade design - Energy efficiency, comfort and daylight in early design phase. *Energy Procedia*, 132, 484-489.  
<http://www.sciencedirect.com/science/article/pii/S1876610217348130>
- HEE, W. J., ALGHOUL, M. A., BAKHTYAR, B., ELAYEB, O., SHAMERI, M. A., ALRUBAIH, M. S. & SOPIAN, K. 2015. The role of window glazing on daylighting and energy saving in buildings. *Renewable and Sustainable Energy Reviews*, 42, 323-343.  
<http://www.sciencedirect.com/science/article/pii/S136403211400793X>
- HINTON, G., SRIVASTAVA, N., KRIZHEVSKY, A., SUTSKEVER, I. & SALAKHUTDINOV, R. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint, arXiv*.
- IOFFE, S. & SZEGEDY, C. 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In: FRANCIS, B. & DAVID, B. (eds.) *Proceedings of the 32nd International Conference on Machine Learning*. Proceedings of Machine Learning Research: PMLR. <http://proceedings.mlr.press/v37/ioffe15.html>
- KAZANASMAZ, T., GÜNAYDIN, H. & BINOL, S. 2009. Artificial neural networks to predict daylight illuminance in office buildings. *Building and Environment*, 44, 1751-1757.
- KLEPEIS, N., NELSON, W., OTT, W. & ROBINSON, J. 2001. The National Human Activity Pattern Survey (NHAPS): A Resource for Assessing Exposure to Environmental Pollutants.
- KRIZHEVSKY, A., SUTSKEVER, I. & HINTON, G. E. 2017. ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60, 84-90.
- KURTZ, A. K. & MAYO, S. T. 1979. Pearson Product Moment Coefficient of Correlation. *Statistical Methods in Education and Psychology*. New York, NY: Springer New York.
- KYRIAKIDES, G. & MARGARITIS, K. 2021. Evolving graph convolutional networks for neural architecture search. *Neural Computing and Applications*. <https://doi.org/10.1007/s00521-021-05979-8>
- LOONEN, R. C. G. M., DE KLIJN-CHEVALERIAS, M. L. & HENSEN, J. L. M. 2019. Opportunities and pitfalls of using building performance simulation in explorative R&D contexts. *Journal of Building Performance Simulation*, 12, 272-288.  
<https://doi.org/10.1080/19401493.2018.1561754>
- LORENZ, C.-L., PACKIANATHER, M., SPAETH, A. B. & SOUZA, C. B. D. 2018. Artificial neural network-based modelling for daylight evaluations. *Proceedings of the Symposium on Simulation for Architecture and Urban Design*. Delft, Netherlands: Society for Computer Simulation International.
- LORENZ, C. L., SPAETH, A. B., BLEIL DE SOUZA, C. & PACKIANATHER, M. S. 2020. Artificial Neural Networks for parametric daylight design. *Architectural science review*, 63, 210-221.
- MAGNIER, L. & HAGHIGHAT, F. 2010. Multiobjective optimization of building design using TRNSYS simulations, genetic algorithm, and Artificial Neural Network. *Building and*



- Environment*, 45, 739-746.  
<http://www.sciencedirect.com/science/article/pii/S0360132309002091>
- MARDALJEVIC, J., HESCHONG, L. & LEE, E. 2009. Daylight metrics and energy savings. *Lighting Research and Technology*, 41, 261-283. <https://www.scopus.com/inward/record.uri?eid=2-s2.0-70149120713&doi=10.1177%2f1477153509339703&partnerID=40&md5=33d549546cf62baebb44dceefc644702>
- MARTÍNEZ, S., EGUÍA, P., GRANADA, E., MOAZAMI, A. & HAMDY, M. 2020. A performance comparison of multi-objective optimization-based approaches for calibrating white-box building energy models. *Energy and Buildings*, 216, 109942.  
<https://www.sciencedirect.com/science/article/pii/S0378778819336850>
- MOREIRA, M. & FIESLER, E. 1995. Neural networks with adaptive learning rate and momentum terms. Idiap. <http://publications.idiap.ch/downloads/reports/1995/95-04.pdf>
- MYHREN, J. & HOLMBERG, S. 2006. Comfort temperatures and operative temperatures in an office with different heating methods. 2, 47-52.
- MYSEN, M. 2017. Termisk inneklima. Betingelser, tilrettelegging og målinger. Byggforsk: Sintef. [https://www.byggforsk.no/dokument/193/termisk\\_inneklima\\_betingelser\\_tilrettelegging\\_og\\_maalinger](https://www.byggforsk.no/dokument/193/termisk_inneklima_betingelser_tilrettelegging_og_maalinger)
- NGARAMBE, J., IRAKOZE, A., YUN, G. Y. & KIM, G. 2020. Comparative Performance of Machine Learning Algorithms in the Prediction of Indoor Daylight Illuminances. *Sustainability*, 12, 4471. <https://www.mdpi.com/2071-1050/12/11/4471>
- NORSK STANDARD 2006. 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 (ISO 7730:2005). *NS-EN ISO 7730:2005*. Norsk Standard.  
<https://www.standard.no/no/Nettbutikk/produktkatalogen/Produktpresentasjon/?ProductID=158329>
- NORSK STANDARD 2011. Light and lighting - Lighting of work places - Part 1: Indoor work places. *NS-EN 12464-1:2011*. Norsk Standard.  
<https://standard.no/no/Nettbutikk/produktkatalogen/Produktpresentasjon/?ProductID=507703>
- NORSK STANDARD 2014. Calculation of energy performance of buildings - Method and data. *NS 3031:2014*. Norsk Standard.  
<https://www.standard.no/no/Nettbutikk/produktkatalogen/Produktpresentasjon/?ProductID=702386>
- NORSK STANDARD 2019a. Daylight in buildings. *NS-EN 17037:2018*. Norsk Standard.  
<https://www.standard.no/no/Nettbutikk/produktkatalogen/Produktpresentasjon/?ProductID=1020692>
- NORSK STANDARD 2019b. Energy performance of buildings - Ventilation for buildings - Part 1: Indoor environmental input parameters for design and assessment of energy performance of buildings addressing indoor air quality, thermal environment, lighting and acoustics (Module M1-6). *NS-EN 16798-1:2019*. Norsk Standard.  
<https://www.standard.no/no/Nettbutikk/produktkatalogen/Produktpresentasjon/?ProductID=1055687>
- OLBINA, S. & BELIVEAU, Y. 2009. Developing a transparent shading device as a daylighting system. *Building Research & Information*, 37, 148-163.  
<https://doi.org/10.1080/09613210902723738>
- ONDEJCIK, V. 2016. *Dynamo Graphic Standards at White arkitekter AB* [Online]. Autodesk, Inc. Available: <http://dynamobim.org/dynamo-graphic-standards-at-white-arkitekter-ab/> [Accessed 03 Feb 2021].
- PALLADINO, D., NARDI, I. & BURATTI, C. 2020. Artificial Neural Network for the Thermal Comfort Index Prediction: Development of a New Simplified Algorithm. *Energies*, 13, 4500.  
<https://www.mdpi.com/1996-1073/13/17/4500>
- PASZKE, A., GROSS, S., MASSA, F., LERER, A., BRADBURY, J., CHANAN, G., KILLEEN, T., LIN, Z., GIMELSHEIN, N., ANTIGA, L., DESMAISON, A., KÖPF, A., YANG, E., DEVITO, Z., RAISON, M., TEJANI, A., CHILAMKURTHY, S., STEINER, B., FANG, L. &

- CHINTALA, S. 2019. *PyTorch: An Imperative Style, High-Performance Deep Learning Library*, Advances in Neural Information Processing Systems 32.
- PATTERSON, D. W. 1996. *Artificial neural networks : theory and applications*, Singapore, Prentice Hall.
- PRAKASH, D. & RAVIKUMAR, P. 2015. Analysis of thermal comfort and indoor air flow characteristics for a residential building room under generalized window opening position at the adjacent walls. *International Journal of Sustainable Built Environment*, 4, 42-57. <http://www.sciencedirect.com/science/article/pii/S2212609015000047>
- PUTH, M.-T., NEUHÄUSER, M. & RUXTON, G. D. 2014. Effective use of Pearson's product-moment correlation coefficient. *Animal Behaviour*, 93, 183-189. <https://www.sciencedirect.com/science/article/pii/S0003347214002127>
- REINHART, C. & SELKOWITZ, S. 2006. Daylighting—Light, form, and people. *Energy and Buildings*, 38, 715-717. <https://www.sciencedirect.com/science/article/pii/S0378778806000661>
- RIF 2020. Dagslys i bygninger. *Beste praksis i byggeprosjekter og forslag til utvikling av regelverket*. Rådgivende ingeniørers forening. <https://www.rif.no/wp-content/uploads/2020/02/Dagslys-februar-2020.pdf>
- ROUDSARI, M. S. & SMITH, A. 2013. Ladybug: A parametric environmental plugin for grasshopper to help designers create an environmentally-conscious design. Available: [https://www.ibpsa.org/proceedings/bs2013/p\\_2499.pdf](https://www.ibpsa.org/proceedings/bs2013/p_2499.pdf).
- ROZENBERG, G., BÄCK, T., KOK, J. N. & ZHANG, G. P. 2012. *Neural Networks for Time-Series Forecasting*, Berlin, Heidelberg :
- RUCK, N., ASCHEHOUG, Ø., AYDINLI, S., CHRISTOFFERSEN, J., EDMONDS, I., JAKOBIAK, R., KISCHKOWEIT-LOPIN, M., KLINGER, M., LEE, E., COURRET, G., MICHEL, L., SCARTEZZINI, J.-L. & SELKOWITZ, S. 2000. *Daylight in Buildings - A source book on daylighting systems and components*, Lawrence Berkeley National Laboratory.
- SHAGHAGHIAN, Z. & YAN, W. 2019. Application of deep learning in generating desired design options: experiments using synthetic training dataset. *Building Performance Analysis Conference and SimBuild*. ASHRAE and IBPSA-USA.
- SHANG, W., SOHN, K., ALMEIDA, D. & LEE, H. 2016. Understanding and improving convolutional neural networks via concatenated rectified linear units. *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*. New York, NY, USA: JMLR.org.
- SHAW, E. W. 1972. Thermal Comfort: analysis and applications in environmental engineering, by P. O. Fanger. 244 pp. DANISH TECHNICAL PRESS. Copenhagen, Denmark, 1970. Danish Kr. 76, 50. *The Journal of the Royal Society for the Promotion of Health*, 92, 164 - 164.
- SINTEF & NTNU 2007. *ENØK i bygninger : effektiv energibruk*, Oslo, Gyldendal undervisning.
- SRIVASTAVA, N., HINTON, G., KRIZHEVSKY, A., SUTSKEVER, I. & SALAKHUTDINOV, R. 2014. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15, 1929-1958.
- SUTSKEVER, I., MARTENS, J., DAHL, G. & HINTON, G. 2013. On the importance of initialization and momentum in deep learning. In: SANJOY, D. & DAVID, M. (eds.) *Proceedings of the 30th International Conference on Machine Learning*. Proceedings of Machine Learning Research: PMLR. <http://proceedings.mlr.press>
- TEK17 2017. Forskrift om tekniske krav til byggverk (FOR-2017-06-19-840). <https://lovdata.no/dokument/SF/forskrift/2017-06-19-840>
- TIAN, Z. & HRYNYSZYN, B. 2020. Overheating risk of a typical Norwegian residential building retrofitted to higher energy standards under future climate conditions. *E3S Web of Conferences*, 172, 02007.
- TRAN, T., LEE, T. & KIM, J.-S. 2020. Increasing Neurons or Deepening Layers in Forecasting Maximum Temperature Time Series? *Atmosphere*, 11, 1072.
- VANHOUTTEGHEM, L., SKARNING, G. C. J., HVIID, C. A. & SVENDSEN, S. 2015. Impact of façade window design on energy, daylighting and thermal comfort in nearly zero-energy houses. *Energy and Buildings*, 102, 149-156. <http://www.sciencedirect.com/science/article/pii/S0378778815003898>

- VINGELMANN, P. & FITZEK, F. H. P. 2020. CUDA, release: 10.2.89. NVIDIA  
<https://developer.nvidia.com/cuda-toolkit>
- WATE, P., COORS, V., IGLESIAS, M. & ROBINSON, D. 2019. *Uncertainty assessment of building performance simulation*, London, United Kingdom :
- XIE, Z., WANG, X., ZHANG, H., SATO, I. & SUGIYAMA, M. 2020. *Adai: Separating the Effects of Adaptive Learning Rate and Momentum Inertia*.
- YU, F., WENNERSTEN, R. & LENG, J. 2020. A state-of-art review on concepts, criteria, methods and factors for reaching ‘thermal-daylighting balance’. *Building and Environment*, 186, 107330. <http://www.sciencedirect.com/science/article/pii/S0360132320306995>
- YU, W., LI, B., JIA, H., ZHANG, M. & WANG, D. 2015. Application of multi-objective genetic algorithm to optimize energy efficiency and thermal comfort in building design. *Energy and Buildings*, 88, 135-143.  
<https://www.sciencedirect.com/science/article/pii/S0378778814010305>
- ZHANG, A., BOKEL, R., VAN DEN DOBBELSTEEN, A., SUN, Y., HUANG, Q. & ZHANG, Q. 2017. Optimization of thermal and daylight performance of school buildings based on a multi-objective genetic algorithm in the cold climate of China. *Energy and Buildings*, 139, 371-384.  
<http://www.sciencedirect.com/science/article/pii/S0378778817301615>
- ZHANG, W. 2010. *Computational ecology: artificial neural networks and their applications*, Singapore, Singapore: World Scientific Publishing Co. Pte. Ltd.
- ZHOU, S. & LIU, D. 2015. Prediction of Daylighting and Energy Performance Using Artificial Neural Network and Support Vector Machine. *American Journal of Civil Engineering and Architecture*, 3, 1-8. <http://pubs.sciepub.com/ajcea/3/3A/1>

## APPENDIX 1 | LITERATURE MATRIX

A complete overview of the literature study with information on aspect, method, description and relevancy. Oria and ScienceDirect is used as main search engines. These have been filtered with research from 2010 or later, language and peer review. In addition, Google was used for more general knowledge of a field or area of study, which was later implemented in the search words. For literature of high relevancy, cited papers was also investigated and used, even if it did not match the filtered criteria. The rating system was divided into three categories from low (<1 point), medium (2-3 points) to high (>4 points) relevancy. All gathered literature is connected to visual and thermal comfort but does not need to contain both comfort categories. The main goal is to find information about implementation of ANNs with visual and thermal comfort. This is to gather and define concepts and clarify a starting point for further discussion.

Source	Aspect	Method	Description and conclusion	Relevance
(Lorenz et al., 2020)	Visual comfort, sDA	Grasshopper, fully connected artificial neural network, Levenberg-Marquardt	Lorenz et al. found great time saving potential when using ANNs for daylight simulations. Due to the achievable time-savings of 65%, ANNs offer a possibility to readapt the brute-force approach into the design process. This enabled all possible instances in the parameterized design solution space to be examined.	High
(Zhou and Liu, 2015)	Visual comfort, UDI	SketchUp, fully connected artificial neural networks, Support vector machines	Different ANN models and support vector machines on simulation data in order to predict hourly results for the climate-based UDI (Useful Daylight Illuminance) metric. Results show that the neural network using principal component analysis generated the highest accuracy (about 96%) compared to other algorithms	High
(Magnier and Haghghat, 2010)	Thermal comfort, PMV, WWR	TRNSYS, ANN, NSGA-II	This paper describes an optimization methodology for two (identical) residential houses. The study revealed a large number of potential designs. The methodology shows the impact of each investment in terms of energy use and thermal comfort. Most difference between the two optimizations results is the size of the north-west window, which was set in a significantly lower value in the second optimization. This can be explained by the fact that when the thermal mass of the house was smaller overheating was more likely to occur in summer. This case highlights the major advantage of multi-objective optimization, which is to bring to light the potentiality of each investment. The occupants of the house could be easily convinced to lower the average PMV from 0.8 (edit: should be raise from 0.08 to 0.11) to 0.11, in order to reduce energy use by up to 13%.	High

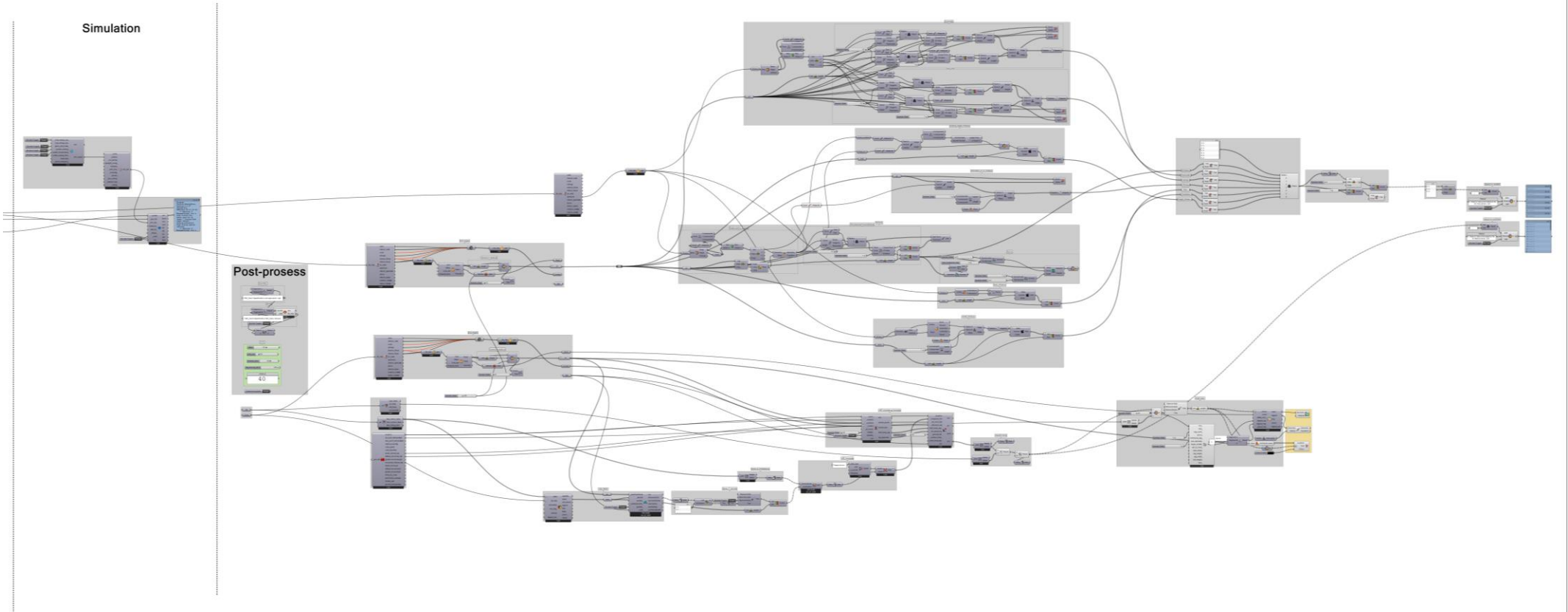
(Bre et al., 2020).	Thermal comfort and energy	ANN, Levenberg–Marquardt	Bre et al propose an optimal way to generate samples used to train and validate the ANN minimizing the total of building energy simulations necessary to train them. The ANN model consists of a Levenberg–Marquardt backpropagation algorithm with a Bayesian regularization which improves the capacity of generalization of the ANN by applying a regularization process. Results indicated that the presented method is able to reduce up to 75% of the number of building energy simulations needed to find the Pareto front while keeping a good accuracy of the results	High
(Ngarambe et al., 2020)	Visual comfort, Illuminance, WWR, DFW, WR, GHI, DNI, DHI, GHIL, RH, SC.	Generalized linear models, deep neural networks, random forest, and gradient boosting models, Radiance	Ngarambe et al. developed and compared the performance of five common ML algorithms for predicting hourly illuminance based on generic simulated data. The algorithms consisting of generalized linear models, deep neural networks, random forest, long short-term memory networks and gradient boosting models. The models considered in total 14 input variables. The data is divided into 80 % training data and 20 % validation data, RMSE is used as loss function and Adaptive Momentum Estimation (Adam) is used as optimizer. Rectified linear units (ReLU) is used as activation function and is reported to enable better performance of deep neural networks models than the Tanh or Maxout functions. Deep neural network outperformed the other network with a determination of coefficient of 99 %. The results also indicated that distance from window, time of day, direct normal irradiance, and diffuse horizontal irradiance were the three most important factors explaining the distribution of indoor daylight illuminance.	High
(Palladino et al., 2020)	Thermal comfort, PMV, indoor air temperature, relative humidity, and clothing insulation	Fully connected artificial neural network, using measured data.	Palladino et al. used ANNs for assessment to thermal comfort for predicting PMV. Three input variables are used including indoor air temperature, relative humidity, and clothing insulation. The network consists of one hidden layer. The data is divided into 70 % training data and 15 % validation data and 15 % testing data. It was concluded that the model was suitable for the simplified calculation of PMV, reaching global regression value of 93 % (Palladino et al., 2020).	High
(Shaghaghian and Yan, 2019)	Visual comfort, UDI	Convolutional Neural Networks, Generative Adversarial Networks, Grasshopper	Shaghaghian and Yan found good predictions for shape classification using Generative Adversarial Networks (GANs) and Convolutional Neural Networks (CNNs). The results show 93 % accuracy when testing with manually drawn shapes.	Medium

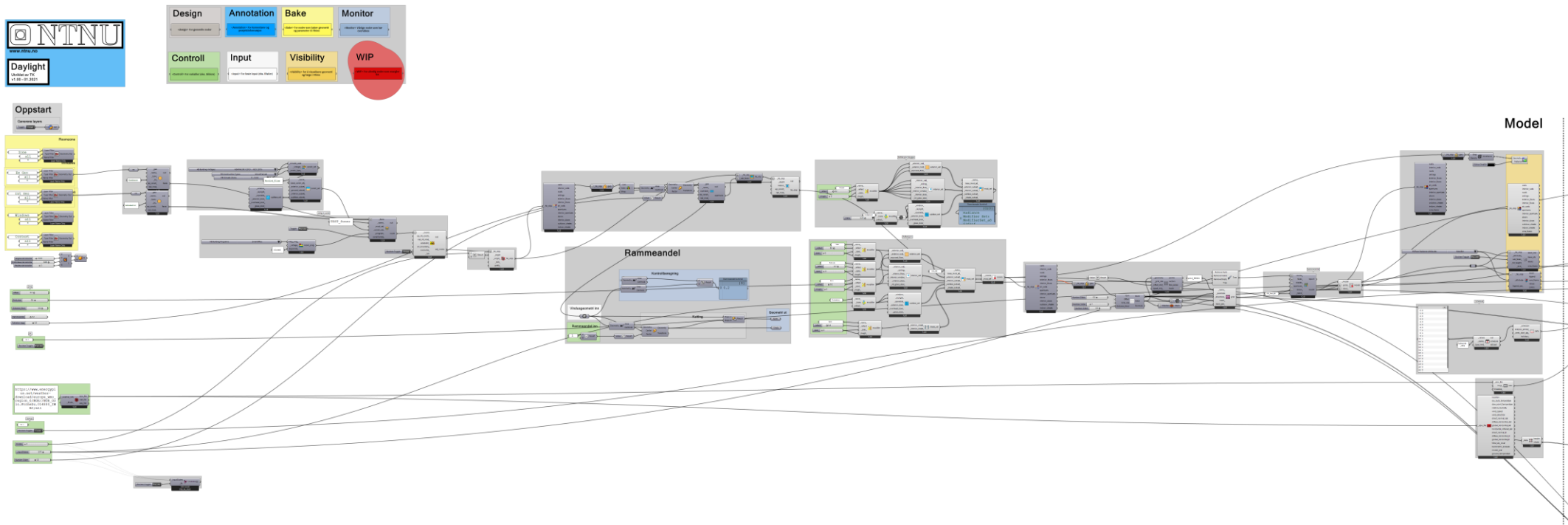
(Lorenz et al., 2018).	Visual comfort, DA	Fully connected artificial neural network, Levenberg-Marquardt, Grasshopper	Lorenz et al. used Neural networks for predicting Daylight Autonomy (DA) levels for an office building with variations in window location and shading. 90 % of the data was used for training and 10 % for validation. The model predicted DA results within a 3% range.	Medium
(Kazanasmaz et al., 2009)	Visual comfort, Lux	Fully connected artificial neural network, measured data	Kazanasmaz et al. have previously constructed a prediction model for daylight illuminance based on 3 months measured data and a three-layer feed-forward ANN model and one output node. 80 datasets were used, and the model had a 98% accuracy.	Medium
(Vanhoutteghem et al., 2015)	Visual and Thermal comfort, DF, degree-hours	Energy+ and DAYSIM	This paper focuses on the relationship between size, orientation and glazing properties of façade windows for different side-lit room geometries in Danish ‘nearly zero-energy’ houses. The effect of these parameters on space heating demand, daylighting and thermal environment. Findings in this paper showed that g-values above 0.3–0.4 have limited effect on decreasing the space heating demand for large WFR in south-oriented rooms. Permanent solar shading solutions, such as solar-coated glazing products with some degree of daylight efficiency, could be used as robust, user-friendly and cost-effective alternatives to dynamic solar shading devices to reduce overheating in south-oriented rooms. In north-oriented rooms, high g-values are recommended to reduce space heating demand. WFR can be chosen relatively freely to fulfil the daylight target. Optimum WFR of approximately 15–20% can be found in both north and south oriented rooms.	Medium
(Yu et al., 2015)	Thermal comfort	Fully connected artificial neural network	The study uses a ANN which is optimized by a genetic algorithm to characterize building behavior, so it can rapidly predict the energy use and indoor thermal comfort. The relative error between prediction and simulation value is 2.0-2.5 %. There is need for extensive testing or simulation data for training samples of each building type or building form.	Medium
(Zhang et al., 2017)	Energy, visual and thermal comfort, UDI, WWR	Rhino/Grasshopper	This paper described an optimization methodology for three different classrooms with parameters such as room depth, WWR, glazing and shading type and orientation. The criteria used is amount of summer hours above an adaptive comfort model and average UDI. In terms of the overall best solutions for each unit model, energy demand for heating and lighting is reduced by 24–28% and summer thermal discomfort by 9–23% while at the same time the UDI (100–2000 lx) increases with 15–63%.	Medium

(Yu et al., 2020)	Optimal visual and thermal comfort methods	Literature study	In this study five methods for achieving the thermal and daylighting balance has been established: Separate investigation (M1), separate investigation + overall comparison (M2), one-another (M3), together (M4), and multi-objective optimization (M5). M1 and M2 where frequently applied, while M5 was most effective.	Medium
-------------------	--	------------------	--	--------



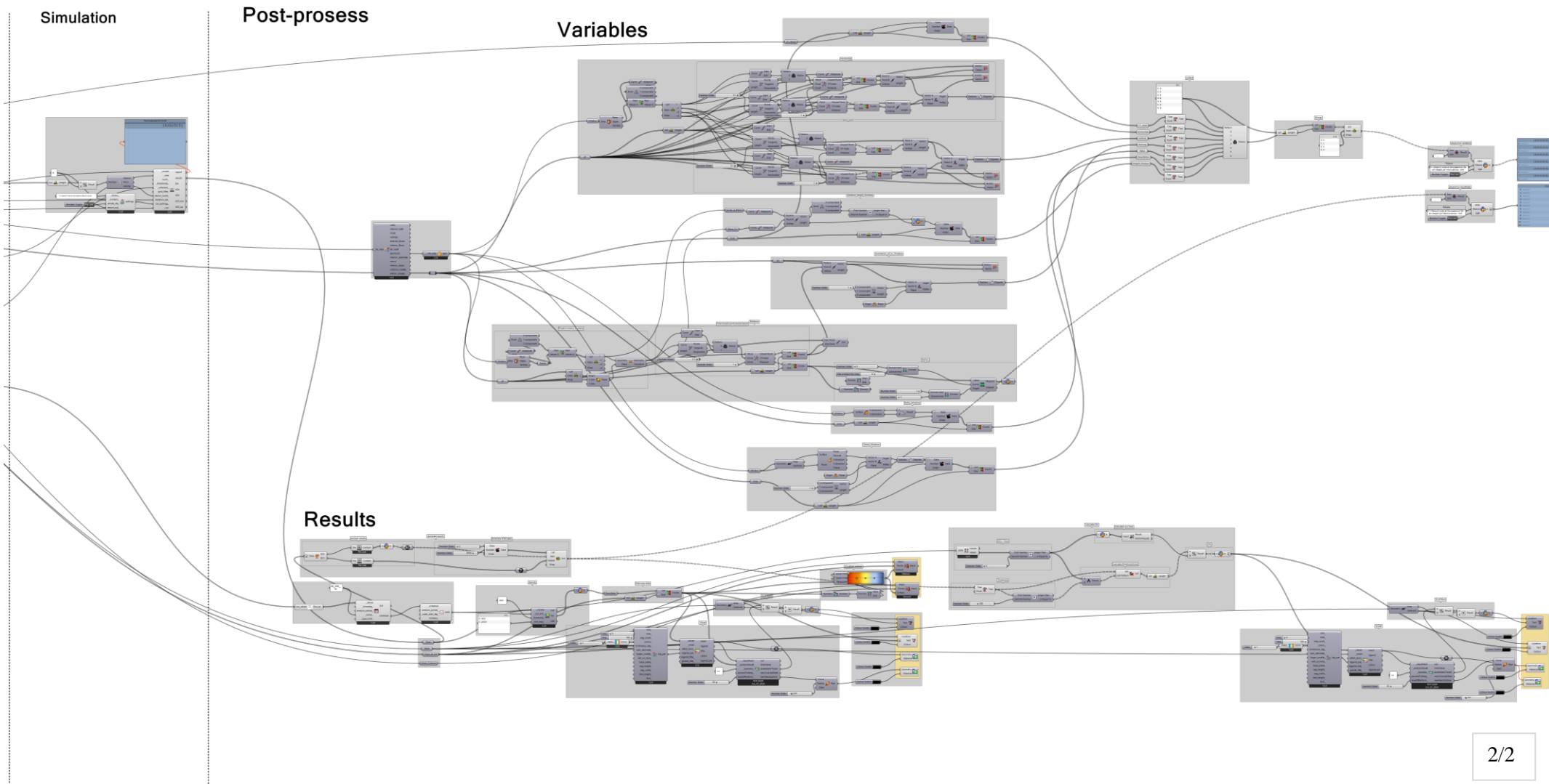






## Daylight illuminance script

This script was developed for generating training data for the ANN model using ladybug tools and custom python codes. The graphical color-coding standard (Dynamo Standard) for Grasshopper, has been implemented as a part of the code validation and readability.



## APPENDIX 3 | PYTHON ANN MODEL

The following custom machine learning algorithm is developed using Python and the open source machine learning framework, Pytorch.

```

import torch
import torch.nn as nn
import torch.nn.functional as F
from torch.utils.data import TensorDataset, DataLoader
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
import math

def read_file(file):
    # åpne fil og dele opp i elementer
    f = open(file, "r")
    x = []
    for line in f.readlines():
        for elem in line.split(","):
            x.append(float(elem))
    return x

def save_file(file, data):
    # lagre elementer til fil
    f = open(file, "w")
    for line in data:
        for elem in line:
            f.write(str(elem)+" ")
        f.write('\n')
    f.close()
    print("fil lagret")

def chunks(lst, n):
    # splitte elementer opp i repspektive variablgrupper
    for i in range(0, len(lst), n):
        yield lst[i:i + n]

def train_val_test_split(input, output, train_size, val_size, test_size, random_state=10):
    # Dele opp Treningsdata, Valideringsdata og Testdata
    X_train, X_val, y_train, y_val = train_test_split(input, output,
test_size=(val_size+test_size), random_state=random_state, shuffle=True)
    X_val, X_test, y_val, y_test = train_test_split(X_val, y_val,
test_size=(test_size/(val_size+test_size)), random_state=random_state, shuffle=True)
    return X_train, X_val, X_test, y_train, y_val, y_test

class Model(nn.Module):
    #Definere nettverk arkitektur
    def __init__(self, n_in, n_out):
        super(Model, self).__init__()
        # Sette opp Layers for nettverk (INN/UT)
        #neurons i layer:

```

```

self.fc_in = nn.Linear(n_in, n_layer0)
self.bn_in = nn.BatchNorm1d(num_features=n_layer0)
self.fc_hidden1 = nn.Linear(n_layer0, n_layer1)
self.bn_hidden1 = nn.BatchNorm1d(num_features=n_layer1)
self.fc_hidden2 = nn.Linear(n_layer1, n_layer2)
self.bn_hidden2 = nn.BatchNorm1d(num_features=n_layer2)
self.fc_hidden3 = nn.Linear(n_layer2, n_layer3)
self.bn_hidden3 = nn.BatchNorm1d(num_features=n_layer3)
self.fc_hidden4 = nn.Linear(n_layer3, n_layer4)
self.bn_hidden4 = nn.BatchNorm1d(num_features=n_layer4)
self.fc_out = nn.Linear(n_layer4, n_out)

def forward(self, x):
    #Definere gjemte Layers #F.ReLU()
    x = F.relu(self.bn_in(self.fc_in(x)))
    x = F.relu(self.bn_hidden1(self.fc_hidden1(x)))
    x = F.relu(self.bn_hidden2(self.fc_hidden2(x)))
    x = F.relu(self.bn_hidden3(self.fc_hidden3(x)))
    x = F.relu(self.bn_hidden4(self.fc_hidden4(x)))
    x = self.fc_out(x)
    return x

if __name__ == '__main__':
    # Check for CUDA else CPU
    use_cuda = torch.cuda.is_available()
    print("Har Cuda: ", use_cuda)
    #print("device: ", torch.cuda.device_count())
    device = torch.device("cuda:0" if use_cuda else "cpu")
    torch.backends.cudnn.benchmark = True

    # Model blir initialisert og __init__ funksjonen kalt
    model = Model(7, 8760).to(device)

    #Tren på eksisterende trent modell
    eksisterende_modell = False
    if eksisterende_modell:
        model.load_state_dict(torch.load("folder/modellname.pth"))
        model.train()
    print("Bruker eksisterende modell: ", eksisterende_modell)

    #Definere Loss og Optimizer
    loss_function = nn.MSELoss()
    momentum = 0.99
    lr = 0.01
    optimizer = torch.optim.SGD(model.parameters(), lr=lr, momentum=momentum)

    # System instillinger
    batch_size = int(64)
    num_epochs = 250

    # Parametere å bruke for dataloaderen.

```

```

params = {
    'batch_size': batch_size, # vanlig med 16, 32, 64 x**2
    'shuffle': True,
    'num_workers': 0,
    'drop_last': False
}

#Hente Treningsdata
folder = "folder"
variabler = list(chunks(read_file(folder+"variabler.txt"), 7))
resultater = list(chunks(read_file(folder+"resultater.txt"), 8760))

#Slette item ved behov
new_variabler = []
for item in variabler:
    new_variabler.append(item[:])
variabler = new_variabler

# Lese dimensjoner
print(f'variabler lastet inn: {len(variabler), len(variabler[0])}')
print(f'resultater lastet inn: {len(resultater), len(resultater[0])}')

#Splitte opp data
X_train, X_val, X_test, y_train, y_val, y_test = train_val_test_split(
    variabler, resultater, 0.75, 0.15, 0.15, 10)

print("Split: Train, val , test:", len(X_train), len(X_val), len(X_test))

# X og Y gjort om til dataset ("TensorDataset") og dataen fra dette datasettet blir
hentet ut av en dataloader ("DataLoader")
Dataloader_train = DataLoader(TensorDataset(torch.tensor(X_train, dtype=torch.float32,
device=device), torch.tensor(y_train, dtype=torch.float32, device=device)), **params)
Dataloader_val = DataLoader(TensorDataset(torch.tensor(X_val, dtype=torch.float32,
device=device), torch.tensor(y_val, dtype=torch.float32, device=device)), **params)
Dataloader_test = DataLoader(TensorDataset(torch.tensor(X_test, dtype=torch.float32,
device=device), torch.tensor(y_test, dtype=torch.float32, device=device)), **params)

#Generere størrelse på batcher
batches_in_epoch_train = len(X_train) / batch_size
batches_in_epoch_val = len(X_val) / batch_size
batches_in_epoch_test = len(X_test) / batch_size
print(batches_in_epoch_train, batches_in_epoch_val, batches_in_epoch_test, batch_size)

# Models
ANN_models = [[5, 1, 1, 1, 1], [2, 2, 2, 2, 2], [8, 6, 4, 2, 1], [1, 2, 4, 6, 8],
              [6, 2, 1, 2, 6]]
for model in len(ANN_models):
    n_layer0, n_layer1, n_layer2, n_layer3, n_layer4 = ANN_models[model]
# Generere Loggere
log_loss = []
log_epoch_loss_e = []

```

```

log_CVRMSE = []
log_NMBE = []
log_acc = []
logg_PCC = []
log_momentum = []
log_lr = []

#Trening
krav = 1
CVRMSE = 10000
accuracy = 0.0
PCC = 0.0
epoch_val_loss = float(len(X_val))
for epoch in range(num_epochs):
    # Iterere over hver epoch
    if CVRMSE <= 1.0 or PCC >=90.0:
        #Trene så lenge kriteriene ikke er tilfredstilt
        break
    else:

        if PCC > krav and epoch > 3:
            krav += 1
            momentum -= momentum/25
            lr -= lr/50
            optimizer = torch.optim.SGD(model.parameters(), lr=lr, momentum=momentum)

        batch_loss = 0.0
        epoch_loss = 0.0
        batch_størrelse = 0

        for input, ground_truth in Dataloader_train:
            input = input.to(device)
            ground_truth = ground_truth.to(device)
            #Iterere over hver batch
            optimizer.zero_grad()
            output = model(input)
            loss = torch.sqrt(loss_function(output, ground_truth))
            log_loss.append(loss.item())
            loss.backward()
            optimizer.step()

            #Logge_loss
            epoch_loss += (loss.item() * output.shape[0])

        if (epoch+1)%1==0:
            #Rapportere Fremgang
            epoch_val_loss = 0.0
            epoch_acc = 0.0
            correct = 0.0
            NMBE_step = 0.0
            CVRMSE_step = 0.0

```

```

pcc_step = 0.0
for input_val, ground_truth_val in Dataloader_val:
    # Rapportere validering
    input_val = input_val.to(device)
    ground_truth_val = ground_truth_val.to(device)

    #Modell validering
    output_val = model(input_val)
    optimizer.step()
    val_loss = torch.sqrt(loss_function(output_val, ground_truth_val))

    #Kalkulere Accuracy og Error
    #Abs error threshold
    correct += (abs(output_val -
ground_truth_val)).float().sum()/8760/ground_truth_val.mean()
    # Pearson Correlation Coefficient as cost function
    vx = output_val - torch.mean(output_val)
    vy = ground_truth_val - torch.mean(ground_truth_val)
    pcc_step += torch.sum(vx * vy) / (torch.sqrt(torch.sum(vx ** 2)) *
torch.sqrt(torch.sum(vy ** 2)))
    #NMBE: Normalized Mean Bias Error
    NMBE_step += (output_val-
ground_truth_val).float().sum()/output_val.float().sum()
    #CV(RMSE): Coefficient of Variation of the Root Mean Square Error
    CVRMSE_step += (torch.sqrt((torch.square(output_val-
ground_truth_val)).float().sum()/8760)/ground_truth_val.mean()) #len(ground_truth_val)
    #print(pcc_step.item(), (torch.sum(vx * vy) / (torch.sqrt(torch.sum(vx
** 2)) * torch.sqrt(torch.sum(vy ** 2))).item())
    #RMSE: Root Mean Square Error
    epoch_val_loss += (val_loss.item() * output_val.shape[0])

    #kalkulere loss og acc
    CVRMSE = 100 * CVRMSE_step / len(X_val)
    NMBE = 100 * NMBE_step / len(X_val)
    accuracy = 100 * (1-(correct /len(X_val)))
    PCC = 100 * pcc_step / batches_in_epoch_val
    #logge data
    log_CVRMSE.append(CVRMSE.item())
    log_NMBE.append(NMBE.item())
    log_acc.append(accuracy.item())
    logg_PCC.append(PCC.item())
    log_lr.append(lr*100)
    log_momentum.append(momentum*100)
    print("Epoch {}/{}", Loss: {:.3f}, Loss_val: {:.3f}, CVRMSE: {:.2f}, NMBE:
{:.1f}, Accuracy: {:.1f}, PCC: {:.1f}, Lr: {:.2f}, Momentum: {:.0f}".format(epoch + 1,
num_epochs, epoch_loss / len(X_train),epoch_val_loss/ len(X_val), CVRMSE, NMBE,
accuracy,PCC, lr*100, momentum*100))

    #Sette sammen logget loss
    log_epoch_loss_e.append(epoch_loss/len(X_train))

```



```

pcc_step = 0.0
epoch_test_loss = 0.0
epoch_acc = 0.0
correct = 0.0
NMBE_step = 0.0
CVRMSE_step = 0.0
for input_test, ground_truth_test in Dataloader_test:
    # Rapportere testing
    input_test = input_test.to(device)
    ground_truth_test = ground_truth_test.to(device)
    output_test = model(input_test)
    optimizer.step()
    test_loss = torch.sqrt(loss_function(output_test, ground_truth_test))

    # Kalkulere Accuracy og Error
    correct += (abs(output_test -
ground_truth_test)).float().sum()/8760/ground_truth_test.mean()
    NMBE_step += (output_test - ground_truth_test).float().sum() /
output_test.float().sum()
    CVRMSE_step += torch.sqrt(torch.square((output_test -
ground_truth_test)).float().sum() / 8760) / ground_truth_test.mean()
    epoch_test_loss += (test_loss.item() * output_test.shape[0])
    # Pearson Correlation Coefficient as cost function
    vx = output_test - torch.mean(output_test)
    vy = ground_truth_test - torch.mean(ground_truth_test)
    pcc_step += torch.sum(vx * vy) / (torch.sqrt(torch.sum(vx ** 2)) *
torch.sqrt(torch.sum(vy ** 2)))

    # kalkulere loss og acc
    CVRMSE = 100 * CVRMSE_step / len(X_test)
    NMBE = 100 * NMBE_step / len(X_test)
    accuracy = (1-(correct /len(X_test)))*100
    PCC = 100 * pcc_step / batches_in_epoch_test
    # logge data
    log_CVRMSE.append(CVRMSE.item())
    log_NMBE.append(NMBE.item())
    log_acc.append(accuracy.item())

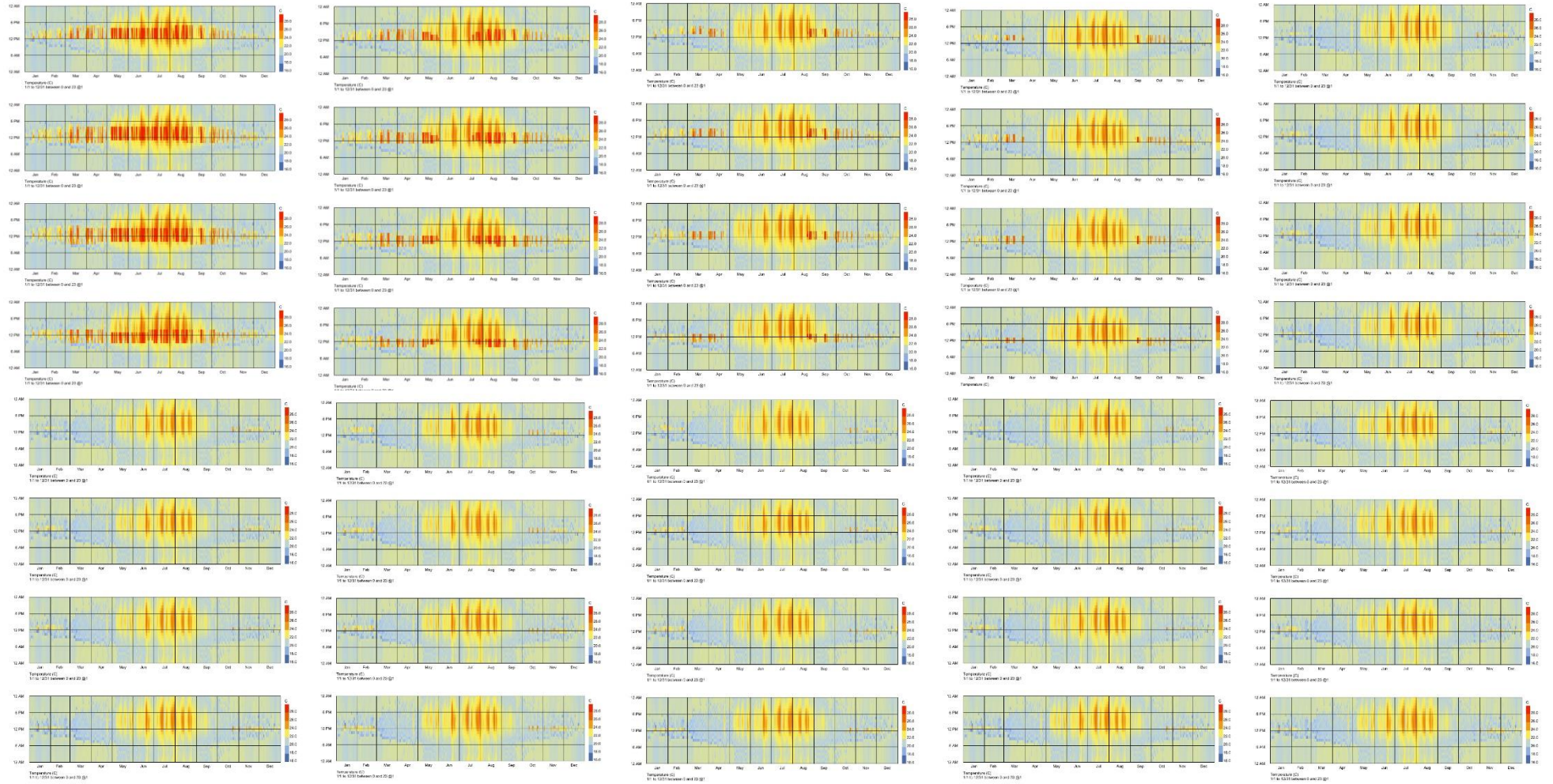
    print("Test, Loss_test: {:.3f}, CVRMSE: {:.1f}, NMBE: {:.1f}, Accuracy: {:.1f}, PCC:
{:.1f}".format(epoch_test_loss/len(X_test), CVRMSE, NMBE, accuracy, PCC))

#Lagre modell
torch.save(model.state_dict(), "folder/fullptThermmodell.pth")
print("done :)")

```

# APPENDIX 4 | SIMULATION RESULTS FOR CASE 1

Carpet plot of annual operative temperature results for all sensor nodes for test-case with parameters g-value 0.5, WWR 0.4 and orientation 180 degrees (south). Plots to the upper left is closest to the window and plots to the lower right is furthers from the window.



Carpet plot of annual daylight illuminance results for all sensor nodes for test-case with parameters LT-value 0.7, WWR 0.4 and orientation 180 degrees (south). Plots to the upper left is closest to the window and plots to the lower right is furthers from the window.

