Advanced Data-Driven Methods for IOT Sensor Data and Anomaly Analysis in Building Environments

by

Ashani Wickramasinghe

A Thesis submitted to the Faculty of Graduate Studies of
The University of Manitoba
In partial fulfilment of the requirements of the Degree of

Doctor of Philosophy

Department of Statistics
University of Manitoba
Winnipeg

Abstract

In commercial buildings, maintaining and controlling the indoor environment while balancing thermal comfort, building health, and energy consumption can be challenging. This thesis focuses on developing statistical and machine learning methods to study the indoor environment of commercial buildings using IoT sensors to identify issues that could affect building health or tenant comfort. Controlling the temperature within an acceptable thermal comfort range is crucial. To address this, we developed and applied statistical methods to identify locations with similar environmental conditions, which helps optimize the HVAC (Heating, Ventilation, and Air Conditioning) system. In this study, we introduced a novel method for identifying time series clusters using community detection in network analysis. Additionally, buildings may have significantly high-temperature 'hot spots' and low-temperature 'cold spots'. To identify these locations, we developed an improved hot spot analysis approach. We also proposed a model to detect locations with abnormal behaviors due to environmental issues. Finally, we developed a Bayesian predictive model using Bayesian networks by introducing a novel method for learning network structures to enhance prediction accuracy and estimate the probability of encountering abnormal locations in the future.

Keywords: Anomaly Detection; Bayesian Network; Building science; Conditional Dependencies; Hotspot Analysis; Machine Learning; Spatial Autocorrelation; Structure Learning; Time series clustering.

Acknowledgment Page

Completing this PhD thesis has been a challenging yet immensely rewarding journey, and it would not have been possible without the support and encouragement of many individuals. First and foremost, I am deeply grateful to my advisor, Prof. Saman Muthukumarana, for his invaluable guidance, encouragement, and support. Your insightful feedback and expertise have been helping me navigate the challenges of this journey. Thank you for believing in me and for being an inspiration.

I would also like to thank my committee, Dr. Po Yang and Dr. Cuneyt Akcora for their time commitment and dedication in serving on my committee. I would like to thank the external examiner Dr. Cindy Feng for her time and valuable feedback on the success of this thesis. I would also like to thank Matt Schaubroeck of ioAirFlow and his R&D team for providing the data and domain knowledge that was instrumental in developing the models for this thesis.

Many thanks go to the faculty, the staff members, and the colleagues within the Department of Statistics at the University of Manitoba. I also thank the Faculty of Graduate Studies for the financial support.

On a personal note, my family and friends for their unwavering support, love, and encouragement. To my mother and father, your belief in me and your sacrifices have been a constant source of strength. I love you both a lot. I am especially grateful to my sister and brother for their continuous support throughout this endeavor.

Last but certainly not least, I would like to thank my husband Dhanushka for his love, understanding, and continuous support, and for never letting me give up.

Dedication Page

This thesis is dedicated to my husband, father, mother, father-in-law, mother-in-law, sister, and brother, who have supported me with heart and soul.

Contents

List of Tables

List of Figures

Contribution of Authors

The thesis is structured as a "sandwich" thesis, with chapters 2, 3, 4, and 5 containing material from journal articles that have either been published or are under review. Following are the titles of the prepared research articles, the manuscript review status at the time of the thesis submission, the authors' contributions, and the corresponding chapters in the thesis:

1. Chapter 2 includes the content published in the Energy Informatics Journal titled "Temperature clusters in commercial buildings using k-means and time series clustering". DOI: https://doi.org/10.1186/s42162-022-00186-8. Author contributions:

    - **Ashani Wickramasinghe** *(Thesis author)*: Conceptualization, Visualization, Methodology, Software, Draft the Manuscript.
    - **Saman Muthukumarana** *(Advisor)*: Conceptualization, Methodology, Supervision, Writing - review & editing.
    - **Matt Schaubroeck**: Conceptualization, Domain Knowledge, Writing - review & editing.
    - **Dan Loewen**: Conceptualization, Data Collection, Domain Knowledge, Writing - review & editing

2. Chapter 3 includes the content of the manuscript under review in the Journal of Building Engineering and titled "Hotspot analysis in commercial buildings using moran's i statistic".

   Author contributions:

   - **Ashani Wickramasinghe**: Conceptualization, Visualization, Methodology, Software, Draft the Manuscript.
   - **Saman Muthukumarana**: Conceptualization, Methodology, Supervision, Writing - review & editing.
   - **Matt Schaubroeck**: Conceptualization, Data Collection, Domain Knowledge, Writing - review & editing.

3. Chapter 4 includes the content published in the Scientific Reports Journal titled "An anomaly detection method for identifying locations with abnormal behavior of temperature in school buildings". DOI: https://doi.org/10.1038/s41598-023-49903-7

   Author contributions:

   - **Ashani Wickramasinghe**: Conceptualization, Visualization, Methodology, Software, Draft the Manuscript.
   - **Saman Muthukumarana**: Conceptualization, Methodology, Supervision, Writing - review & editing.
   - **Matt Schaubroeck**: Conceptualization, Data Collection, Domain Knowledge, Writing - review & editing.
   - **Surajith N. Wanasundara**: Conceptualization, Writing - review & editing.

4. Chapter 5 includes the content of the manuscript under review in the International Journal of Data Science and Analytics and titled "Enhanced anomaly detection through a bayesian framework with a novel network merging structure learning approach".

   Author contributions:

   - **Ashani Wickramasinghe**: Conceptualization, Visualization, Methodology, Software, Draft the Manuscript.
   - **Saman Muthukumarana**: Conceptualization, Methodology, Supervision, Writing - review & editing.

Chapter 1

Introduction

In this thesis, we focus on utilizing IoT sensor data to enhance the indoor environment of commercial buildings. By deploying these sensors, we can collect valuable information about indoor conditions, allowing us to uncover insights and develop statistical and machine-learning models to detect issues within the buildings. The primary objective of this research is to develop advanced models that can identify these issues effectively, allowing building owners and engineers to take necessary actions to improve the indoor environment. Currently, most issues are identified through manual auditing processes, which are time-consuming and prone to human error. To address this, we propose four improved approaches that work together to detect abnormal behaviors in the indoor environment.

The building's Heating, Ventilation, and Air Conditioning (HVAC) system controls the indoor environment. Thermostats play a crucial role by sending data about the indoor environment to the HVAC system. Therefore, optimizing the placement of these thermostats is essential for ensuring accurate data transmission and effective climate control. Identifying significantly hot or cold areas and understanding the

factors contributing to these conditions is important for tenant comfort, energy efficiency, and the overall health of the building. Furthermore, detecting abnormal locations based on unusual environmental behaviors and predicting abnormal conditions are key aspects of this research. These four findings will help building owners create energy-efficient, healthy buildings and comfortable environments for their tenants. The motivations behind proposing these approaches are discussed in the following section.

1.1 Motivations

1.1.1 Optimizing Thermostat Placement

In modern commercial buildings, maintaining an optimal indoor environment is crucial for both energy efficiency and occupant comfort. Thermostats play a key role in this process as they control the heating, ventilation, and air conditioning (HVAC) system. By accurately sensing the temperature, thermostats help ensure that spaces are neither too hot nor too cold. Properly positioned thermostats can lead to significant improvements in minimizing energy waste and enhancing the overall environment without the need for additional or more expensive units.

For instance, consider a commercial office building in which thermostats are placed randomly. This can lead to an uneven distribution of temperature in the surroundings, where some areas may get too warm, while others remain too cold. Because of this, the occupants may adjust their personal space heaters or cooling units, leading to more energy consumption and higher operational costs. In contrast, the optimization of thermostat location would mean that temperature readings can actually reflect

conditions in each area, and it would also provide a more balanced and efficient climate control system.

To address the challenge of identifying the most suitable locations for thermostat placement, we developed a solution focusing on locations with similar environmental behaviors. Since temperature exhibits temporal behavior, we used time series clustering techniques to analyze and group these locations effectively. This approach allows us to identify areas within the building that experience similar temperature patterns throughout the day. Furthermore, we enhanced the detection of time series clusters by using network community detection techniques with a correlation matrix. This approach is further explained in Section 1.2.1.

1.1.2 Hot and Cold Spot Detection

In a previous study, we focused on optimizing thermostat placements to enhance the performance of HVAC systems. However, many buildings still have areas with significant temperature discrepancies, resulting in some spaces being too cold or too hot compared to their neighboring locations. This can severely impact occupant comfort, lead to increased energy consumption, and harm the building over time.

Identifying these problematic locations early can allow building owners to take action before discomfort becomes a bigger issue. If these hotspots are not identified, they can lead to increased energy consumption, higher maintenance costs, and potential damage to the building's infrastructure due to uneven heating and cooling. For example, chronic cold spots might cause condensation and mold growth, while hot

spots can lead to equipment overheating.

Currently, building engineers use thermal cameras to identify these hotspots, but this can be costly and not always practical. In this study, we propose a more affordable way to find these temperature issues using IoT sensor data for hotspot analysis. This approach is innovative, as it has not been extensively applied to building data in the literature.

We also suggested a new approach to determine the optimum number of neighbors for hotspot detection, which further increased the accuracy of our results. Our suggested approach is discussed in greater detail in Section 1.2.2. We were also able to apply the random forest algorithm to determine some possible causes for these hot and cold spots. This approach will enable the detection of temperature imbalances at an early stage, and also provide valuable insights into determining the factors that cause these imbalances, therefore contributing a great deal toward improving occupant comfort and energy efficiency in buildings.

1.1.3 Environmental Anomalies Detection

Our third study investigates anomaly detection in building environments. Anomaly detection is an important task enabling a better indoor atmosphere that will contribute to occupant comfort and energy savings. Much of the existing literature has focused on detecting contextual anomalies within a single time series, known as intra-time series anomalies. In building science, it was assumed that those are mainly caused by hardware issues. However, we will need to consider the concept of

collective anomalies to detect anomalies based on environmental causes.

By analyzing all data series within a building rather than just a single one, we can better pinpoint locations exhibiting unusual environmental behaviors compared to the rest of the building. For example, suppose we detect an abnormal temperature spike in one location. In that case, this spike may also appear across other areas for acceptable reasons, such as a sudden increase of customers during the happy hour leading to higher indoor temperatures due to body heat. Such a spike should not be considered an abnormal event when evaluating locations.

In contrast, when we focus on inter-time series anomalies and identify a spike in one series that is quite different from the rest, it may indicate a real abnormal event; thus, it will be important for building owners to take on because it allows them to treat actual environmental problems, not typical fluctuations

We introduced a novel scoring method to identify locations where this type of abnormal event occurs. We applied a Dynamic Time Warping-based collective anomaly detection method to effectively detect collective anomalies. The proposed scoring method can be used to identify the most and least abnormal locations. More information about this method will be discussed in Section 1.2.3.

1.1.4 Enhanced Anomaly Detection using Labeled Data

In our previous study, we presented a scoring-based method for detecting abnormal locations in building environments. This initial work showed that most abnormalities significantly affect occupant comfort and energy efficiency. However, we realized that simply identifying these abnormalities was insufficient; understanding the underlying

causes is crucial for effective action. More precisely, we were interested in how labeled data would enable us to enhance our anomaly detection and whether the environmental features of neighboring locations have any impact on the abnormality of a location. This investigation is important because building owners need clear information to address potential issues proactively, ultimately creating a more comfortable indoor and eco-friendly environment.

To address these challenges, we have further developed our collective anomaly detection method with the inclusion of a Bayesian framework. First, we classify time points as normal or abnormal using this new collective anomaly detection method, then use a Bayesian network with a novel structure learning method to predict the likelihood of abnormality for specific locations. The methodological contribution of this study is the proposal of this new merging structure learning method. More details about this can be found in Section 1.2.4.

Additionally, another Bayesian network model was developed that includes features from its neighboring locations. These model assessments showed that this new model remarkably improved predictive accuracy by considering the environmental characteristics of nearby areas. In other words, the neighboring locations have a certain degree of influence on the indoor environment of the current location. This improvement not only increases our capability in terms of anomaly detection but also provides building owners with actionable insights to help them make informed decisions.

1.2 Main Contributions and Thesis Organization

In this thesis, we make several key contributions by proposing various methods to study the indoor environment of commercial buildings using IoT sensors and by identifying issues that may impact building health or tenant comfort. The research presented here aims to assist building owners and engineers in enhancing indoor environmental quality while also advancing the fields of statistics and machine learning through the introduction of novel approaches and improving existing methods. All computations, simulations, and related experiments were conducted using Python. The following is a summary of the our contributions, which are presented in Chapters 2 to 5.

- In Chapter 2, we optimized thermostat placement by identifying locations with similar environmental conditions using clustering. Given that environmental data exhibits temporal behavior, we demonstrated that time series clustering is more effective for grouping these locations. This work has been published in the *Energy Informatics* Journal [9]. An extension of this study focused on enhancing time series clustering through community detection in network analysis, resulting in improved outcomes compared to traditional time series clustering. Both this manuscript and its extension form the second chapter of this thesis.

- In Chapter 3, we propose an improved approach to identify hot and cold spots within a building. Hotspot analysis has been widely used in various research fields, but it has not yet been applied in building science. Through this study, we demonstrate that hotspot analysis can effectively identify significant hot

and cold spots inside buildings and is a more cost-effective method than using infrared cameras. Additionally, we introduce a method for determining the optimal number of neighbors using Moran's I statistic, which enhances the accuracy of the hotspot analysis results. This work is presented in a manuscript currently under review in the *Journal of Building Engineering* [10] and is the third chapter of this thesis.

- In Chapter 4, We developed a time series anomaly detection framework capable of identifying environmental anomalies, including collective anomalies, across multiple time series. These anomalies may indicate location-specific issues, such as problems with the building's HVAC system or other infrastructure concerns. This framework can highlight the most and least abnormal locations by analyzing the temperature behavior of the entire area. This work is presented in a manuscript and has been published in the *Scientific Reports* [11] and forms the fourth chapter of this thesis.

- In Chapter 5, We developed an anomaly detection framework using the Bayesian Network model, in which the network structure plays an important role in determining the outcomes. In this study, we present a new approach that combines classic network structures into a final network structure with an improved prediction capability. This work presents the idea that including environmental features from adjacent locations can improve the accuracy of anomaly prediction for the current location. This work is included in a manuscript that is currently under review and serves as the fifth chapter of this thesis.

Finally, in Chapter 6, we summarize this thesis with some concluding remarks

and discussion on future research directions. Note that each chapter is accompanied by its own bibliography. This is a "sandwich thesis", Chapters 2 to 5 consist of papers that are published or have been submitted for publication. We now briefly outline the methodological contributions presented in those chapters in more detail.

1.2.1 A Community Detection Based Time Series Clustering

In Chapter 2, we discuss our study on clustering building environments using various clustering methods. Based on our results, time series-based clustering performed better. This led us to develop a model aimed at enhancing this clustering capability. We explored the use of community detection from social network analysis for clustering. Community detection identifies communities based on the topological features of the network, making it important to create the most appropriate network structure. Wickramasinghe and Muthukumarana [6, 7] introduced a novel random graph generator using a mixture of Gaussian distributions for community detection in networks. In our approach, we considered sensors as nodes and created connections based on the cross-correlation matrix.

One of the previous studies, conducted by Li and the team [3], used the correlation matrix to determine the strength of connections; they didn't focus on choosing the most important ones. Because of that, it can make the network too complex for community detection. Therefore, this study focuses on identifying the best threshold for deciding whether to create a connection between two nodes. Details on the different threshold measures and case study results can be found in Chapter 2. Our findings show that community-based time series clustering is more effective than traditional time series clustering methods at identifying patterns over time.

1.2.2 Optimizing Number of Neighbors

While hotspot analysis has been widely used in various fields, it has not been applied to identify hot and cold spots within buildings. In Chapter 3, we explain how to implement hotspot analysis specifically for building environments. This analysis focuses on the variable of interest in neighboring locations. Hence, determining the optimal number of nearest neighbors is important for accurate results.

The optimization of the number of neighbors used in the hotspot analysis is one of the main methodological contributions of our study. We introduced a novel method to identify the optimal number of neighbors using Moran's I statistic [4], a well-known measure of spatial autocorrelation. A higher Moran's I value indicates clusters with clearer boundaries. We used that information to refine the optimal number of neighbors in the selection process.

Through experimental analyses, we show that hotspot analysis can be applied inside commercial buildings and that the results make sense. We then found that identifying the optimum number of neighbors greatly improved the accuracy of hotspot detection. This advancement enhances our understanding of temperature variations within buildings and gives useful insights for building owners to manage their indoor environments better.

1.2.3 Abnormal Location Detection using Scoring Method

In Chapter 4, we introduce a framework for selecting the most abnormal locations using a scoring method. This framework identifies environmental abnormal time points by applying an updated anomaly detection method developed by Diab and

his team [1]. Based on existing literature and including feedback from building scientists, we decided to use the median time series as a control line for our analysis. We then calculated scores based on several features: the number of anomalies, the vertical distance from the control time series, and the Dynamic Time Warping (DTW) distance [2].

We used grid search optimization and a simulation study to optimize the weights of this scoring function. It is important to note that using the same weights for each dataset is inappropriate, as different scenarios demand different weightings. Therefore, we aimed to estimate the most suitable basic time series model for the specified control time series. We simulated ten time series using the identified model and generated one abnormal time series by adding white noise to the original time series. Through the grid search algorithm [5], we optimized the weights to obtain the highest score for the abnormal time series. We applied the framework to this building temperature data and synthetic data to identify collective anomalies in time series.

1.2.4 A Novel Structure Learning Method for Bayesian Network

In Chapter 5, we present an improved anomaly detection method using a Bayesian network-based framework [8]. We used the collective anomaly detection method from our previous study to label the data and developed a Bayesian network model to predict the abnormality of specific time points.

The prediction accuracy of a Bayesian network heavily depends on its structure.

Hence, several traditional methods exist for structure learning. However, most of these methods return networks based only on the data, leading to dependencies that may not reflect real-world situations. As a methodological improvement, we developed a new structure learning method that enhances prediction accuracy by merging network structures and removing unrealistic dependencies through the application of certain constraints. Details of this algorithm are described in Chapter 5.

Moreover, the features of the environmental neighbors were also considered during the development of Bayesian network models. The approach developed in Chapter 3 has been used to derive the optimum number of neighbors. Our results demonstrate that including neighboring features increases the accuracy of abnormality detection and, thus, a more effective approach to anomaly detection in building environments.

Bibliography

[1] D. M. Diab, B. AsSadhan, H. Binsalleeh, S. Lambotharan, K. Kyriakopoulos, and I. Ghafir. Anomaly detection using dynamic time warping. pages 193–198, 2019.

[2] O. Gold and M. Sharir. Dynamic time warping and geometric edit distance: Breaking the quadratic barrier. *ACM Trans. Algorithms*, 14:1–17, 2018.

[3] H. Li, X. Wu, X Wan, and W. Lin. Time series clustering via matrix profile and community detection. *Advanced Engineering Informatics*, 54:101771, 2022.

[4] P. A. P. Moran. Notes on continuous stochastic phenomena. *Biometrika*, 37:17–23, 1950.

[5] J. K. Muhammad. Grid search optimization algorithm in python. https://stackabuse.com/grid-search-optimization-algorithm-in-python/, 2020. Accessed 24 May 2022.

[6] A. Wickramasinghe and S. Muthukumarana. Social network analysis and community detection on spread of covid-19. *Model Assisted Statistics and Applications*, 16:37 – 52, 2021.

[7] A. Wickramasinghe and S. Muthukumarana. Assessing the impact of the density and sparsity of the network on community detection using a gaussian mixture random partition graph generator. *International Journal of Information Technology*, 14:607 – 618, 2022.

[8] A. Wickramasinghe and S. Muthukumarana. Enhanced anomaly detection through a bayesian framework with a novel network merging structure learning approach. Manuscript under review, 2024.

[9] A. Wickramasinghe, S. Muthukumarana, D. Loewen, and M. Schaubroeck. Temperature clusters in commercial buildings using k-means and time series clustering. *Energy Informatics*, 5, 2022.

[10] A. Wickramasinghe, S. Muthukumarana, and M. Schaubroeck. Hotspot analysis in commercial buildings using moran's i statistic. Manuscript under review, 2024.

[11] A. Wickramasinghe, S. Muthukumarana, M. Schaubroeck, and S. N. Wanasundara. An anomaly detection method for identifying locations with abnormal behavior of temperature in school buildings. *Scientific Reports*, 13:22930, 2023.

Chapter 2

Temperature Clusters in Commercial Buildings Using K-means and Time Series Clustering

Thermostats send data about the temperature of a space (or zone) within a building to the HVAC system, which then adjusts the air temperature supplied to that space accordingly. Having fewer thermostats installed can result in an incomplete view of the building's performance. Thermostats that are improperly placed can also yield incomplete measurements, causing HVAC systems to run too often or not frequently enough. As a result, the temperature measured by the thermostats may differ significantly from the actual experience of the building's occupants. In this chapter, we identify zones with similar indoor environments using one week of data. We used K-means and time series clustering to identify the temperature clusters per building floor. Based on statistical assessments, we observed that time series clustering showed better results than k-means clustering. With this method,

building owners can optimize thermostat placement, potentially reducing the number of thermostats needed. This leads to more accurate measurements and improved control over the building's thermal performance.

2.1 Introduction

In commercial buildings, it is hard to maintain and control the environment of the building while considering thermal comfort and energy consumption. These buildings are usually equipped with intelligent HVAC (Heating, Ventilation, and Air Conditioning) systems. Thermostats send data about the temperature of a space (or zone) within a building to the HVAC system, which then adjusts the air temperature supplied to that space accordingly. Fewer thermostats installed generate a less complete picture of a building's performance, as different spaces (zones) can have different temperature needs. Thermostats that are improperly placed can also yield incomplete measurements, causing HVAC systems to run too often or not frequently enough. As a result, the temperature measured by the thermostats might be very different from what the building's occupants are experiencing.

The consequences of inefficient HVAC systems can have serious effects on a building and its occupants. People can have adverse impacts on their productivity and cognitive abilities. For example, a direct correlation has been determined that each 4 degree Fahrenheit shift away from the optimal internal temperature of 72 degrees resulted in a 2 percent decrease in productivity. Notably, the economic impact of this productivity decrease also demonstrated that regaining that 2 percent productivity increase yields a 9% increase in net revenue for a company working within that

environment. A study conducted by Allen and Macomber [1] in New York City found that schools with an internal temperature of 90 degrees Fahrenheit had a 14% higher likelihood of failing an exam compared to if the same space were controlled at 75 degrees.

Nowadays, much research has been done on the topic of building health and thermal comfort. In 2017, Lee and the team conducted a research [9] to identify the thermal comfort of a residential house in Malaysia and found that to satisfy human thermal comfort HVAC system is needed for the bedroom and living room. Martin Sarnsvosky and David Bajus [15] used k-means clustering to cluster the university building based on temperature and humidity data which were obtained by sensors. With the development of smart technologies, people tend to build smart thermostats to control HVAC systems. Smart thermostats can collect more information than traditional thermostats and use machine learning algorithms to optimize the setpoint based on both efficiency and occupant comfort. Hence, in 2012, Yun and Won [20] conducted a study to build a smart energy system to control HVAC systems based on temperature and humidity. This system was evaluated using occupants' feedback. But not all building owners are concerned with replacing their existing thermostats with smart devices, due to their high capital cost and how difficult they can be to properly install. A preferred solution would be to find ways to optimize a building's control system using its existing infrastructure, allowing thermostats to be deployed and used more efficiently. Hence, this research work is focused on giving a solution to that.

In this study, we have clustered each building floor based on temperature data which were collected from wireless sensors using k-means and time-series clustering. In the

k-means method, we clustered sensors based on the mean values of variables, while in the time series clustering method, we clustered the sensors that show similar trends over time. We also considered temperature, humidity, and pressure variables and compared the cluster results with each other. Placing a minimum of one thermostat in each identified cluster will generate more accurate measurements and may lead to better control of a building's thermal performance.

2.2 Material and Methods

2.2.1 Data Collection and Preparation

The data collection was done in a commercial building in downtown Winnipeg, MB, Canada. We selected three different floors (first, second, and fourth floors) which were considered as the problematic floors by the building owner. Sixty (60) sensors were strategically positioned throughout these floors and air data (temperature, relative humidity, and pressure) was collected at five minute intervals between February 1 and February 9, 2021. During that period, over 135,000 data points were collected to evaluate the building's performance. Weather data was collected using the Government of Canada's weather API. A sample of the sensor data set and weather data set is shown in Tables 2.1 and 2.2 respectively.

Based on the data, we could understand that not all 60 sensors started to collect data at the same time. For example, when the first sensor started to collect data at "08:00:45", the second one started at "08:00:51". Hence there were a few second differences between the data points of each sensor. It created null data points when

Table 2.1: Sample of sensor data set

Time	**Sensor Number**	**Temperature (°C)**	**Humidity (%)**	**Pressure (kPa)**
2021-02-09 08:58:43	1	26.4	6	99.19
2021-02-09 08:58:42	2	20.3	6	99.15
2021-02-09 08:58:37	3	20.4	6	99.16
2021-02-09 08:58:34	4	20.7	6	99.18

Table 2.2: Sample of weather data set

Time	**Temperature (°C)**	**Relative Humidity (%)**	**Pressure (kPa)**
2021-02-09 08:00	-24.6	61.0	99.20
2021-02-09 09:00	-23.8	58.0	99.22
2021-02-09 10:00	-23.3	52.0	99.28
2021-02-09 11:00	-22.9	52.0	99.31

we considered all time points as not all the sensors have data at each time point. To fix this, we rounded seconds into minutes, in the timestamp and considered only the hours and minutes. Also, another issue was that the weather data were collected on an hourly basis. Because of that, when merging weather data with sensor data we were losing most of the valuable data points. We interpolated the weather data to one minute intervals which helped merge the two data sets without losing any data in the sensor data set.

2.2.2 K-means Clustering

In this analysis, we filtered sensors on each floor and considered the mean temperature of each sensor to cluster the sensors. The main objective of clustering is to group similar data points (mean temperature of each sensor) together and discover the underlying pattern. To achieve this goal, k-means requires a fixed number of clusters (k). This target number k, refers to the number of centroids, which is an imaginary

or real location of the center of the cluster. Then every data point is allocated to the nearest cluster while minimizing the intracluster variation.

In the standard k-means clustering algorithm [7] total within-cluster variation is defined as the sum of squared distances of Euclidean distances between items and the corresponding centroid which is shown as:

$$W(C_k) = \sum_{x_i \epsilon C_k} (x_i - \mu_k)^2. \tag{2.1}$$

Here x_i is i^{th} data point of cluster k (C_k), and μ_k is the mean value of points in cluster k. The total within-cluster variation is defined as Eq(2.2). The total within-cluster sum of squares measures the goodness of the clustering, which increases as the sum of squares measures decreases.

$$\textit{total within cluster variation} = \sum_{k=1}^{k} W(C_k) = \sum_{k=1}^{k} \sum_{x_i \epsilon C_k} (x_i - \mu_k)^2 \tag{2.2}$$

K-means clustering has the following steps; first, specify the number of clusters (k), and second, randomly select k data points as initial centroids. The third step is assigning the remaining data points to their closest centroid. The fourth step is recomputing the cluster centroids by calculating new mean values of data points in formed clusters. Then repeat the third and fourth steps until no changes in clusters or no changes in centroids or maximum number of iterations are achieved.

In this analysis, our objective is to find the optimal number of thermostats for one floor, and to achieve that, we needed to find the optimal number of clusters. For

that, there are different methods, and we used the elbow plot method. Here we conducted clustering using different number of clusters (k) and calculated the total within the sum of squares for each k value and plotted it against k. Finally, the k value at the location of the bend (elbow joint) in the plot is considered the optimal number of clusters. For example, Figure 2.1 represents an elbow plot, and the total sum of squares distance decreases as k increases, but at k=4 there is a bend. It shows that having additional clusters will reduce the sum of squares by small values. Hence four can be considered as the optimal number of clusters.

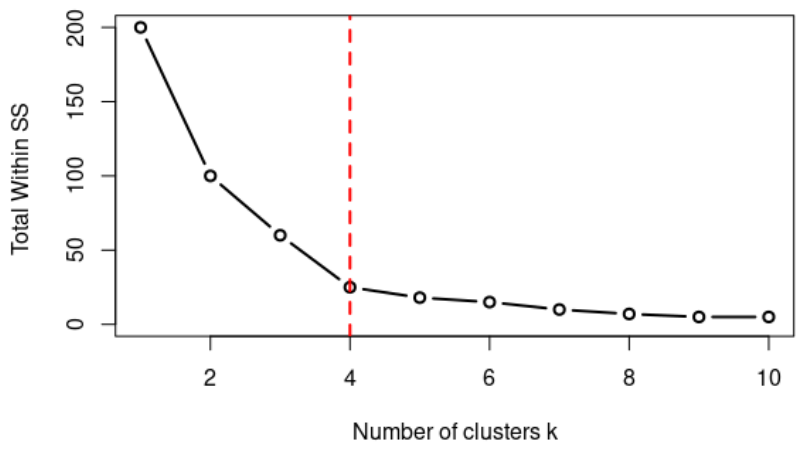


Figure 2.1: Example for Elbow plot to check the optimal number of clusters (k). The red line indicates the elbow joint and it will help to find the k value.

2.2.3 Time series Clustering

In k-means clustering, we considered the mean temperature of each sensor as data points, but by averaging data potentially valuable information is lost. In time series clustering we can overcome that issue by considering all data points and grouping

sensors with similar time series into the same cluster. Here, hierarchical clustering is used to cluster the time series based on Euclidean distance.

Hierarchical clustering produces a nested hierarchy of similar groups of objects, according to a pairwise distance matrix of the objects [5]. In time series clustering, objects are a series of numbers. In the agglomerative algorithm, clusters are initialized with each series that belongs to their own groups. The algorithm then merges the similar groups into larger clusters, based on the distance matrix. There are several types of methods and distance matrices to develop clusters. Hierarchical clustering does not require the number of clusters to generate clustering results.

The clustering method used by Santos and team [14], showed that the mean linkage method using the correlation similarity metric provides the most appropriate results when studying weather variables. This method overcomes some deficiencies of other hierarchical methods in clustering homogeneous groups. Therefore, clustering is less affected by a typical observation in cluster formation, according to Unal et al. [17]. The average distance between all pairs of objects in any two clusters can be calculated as follows:

$$D(r,s) = \frac{1}{n_r n_s} \sum_{i=1}^{n_r} \sum_{j=1}^{n_s} D(x_{ri} - x_{sj}) \tag{2.3}$$

where $D(r,s)$ is the distance between clusters r and s, and (n_r, n_s) are the number of elements in those cluster. The main objective of that study was to cluster precipitation data based on their behavior over time by incorporating their temporal variations [17].

In our study, we used the average method and correlation similarity metric based on

the work by Unal et al [17]. To compare those results, we used Ward's method and Euclidean distance, which is the most popular combination of method and similarity metric for hierarchical clustering. Ward's method creates groups while minimizing the pooled within-cluster sum of squares. The Euclidean distance between two different time series is called Q, and C can be calculated as Eq.(3). Here q_i and c_i represent i^{th} data points of time series Q and C respectively:

$$D(Q, C) = \sqrt{\sum_{i=1}^{n}(q_i - c_i)^2}. \tag{2.4}$$

Multivariate Clustering

In multivariate clustering, we try to cluster sensors, where all the features within each cluster are as similar as possible. In this study, we measured temperature, relative humidity, and pressure. Finding an appropriate method to combine those variables for the time series clustering was one of the challenges in this study.

Since time series clustering uses time series to group objects, we needed to create a time series by considering multiple features. For that, we proposed the following method to create a single time series by considering all variables. In this process, we first normalized all the series (temperature, relative humidity, and pressure) using Min-Max normalization:

$$V_{norm}(i) = \frac{v_i - Min(v)}{Max(v) - Min(v)} \tag{2.5}$$

$$new_{(i)} = \frac{\sum_{j=1}^{n}(V_{norm}(ji))}{n}. \tag{2.6}$$

By normalizing the data we brought all the variables into one scale. Then, using Eq. 2.6, we generated a new variable by combining all three variables. Let $V_{norm}(ji) = i^{th}$ observation of j^{th} variable, $n =$ number of variables, and $new_{(i)} = i^{th}$ observation of new variable. This new variable generated a time series for each sensor, and we used those time series for the clustering.

2.2.4 Similarity Score Measures

Once the clusters are identified based on different algorithms, a comparison study can be done using similarity measures. Those indices measure the similarity between cluster results with true labels.

Adjusted Rand Index (ARI)

The Rand Index computes a similarity measure between two clustering by considering all pairs of samples and counting pairs that are assigned in the same or different clusters in the predicted and true clustering. If the number of data vectors for clustering is n, then there are ${}_nC_2$ pairs. For every example pair, there are three possibilities in terms of grouping. The first possibility is that the paired examples are always placed in the same group as a result of clustering (a). The second possibility is that the paired examples are never grouped together (b). The third possibility is that the paired examples are sometimes grouped and sometimes not grouped together. The RI of two groupings is then calculated by the following formula:

$$RI = \frac{Count\ of\ Pairs\ in\ Agreement}{Total\ Number\ of\ Pairs} = \frac{a+b}{{}_nC_2}. \tag{2.7}$$

RI had one drawback; it yields a high value for pairs of random partitions of a given set of examples. To overcome this drawback, the Rand Index score is then "adjusted for chance" into the Adjusted Rand Index [8] score using the following scheme:

$$ARI = \frac{RI - Expected\ RI}{max(RI) - Expected\ RI}. \tag{2.8}$$

The adjusted Rand index is thus ensured to have a value close to 0 for random labeling independently of the number of clusters and samples and exactly 1 when the clusterings are identical (up to a permutation).

Normalized Mutual Information (NMI)

The Mutual Information (MI) is a measure of the similarity between two labels of the same data. Where $|U_i|$ is the number of the samples in cluster U_i and $|V_j|$ is the number of the samples in cluster V_j, the Mutual Information between clustering U and V is given as:

$$MI(U,V) = \sum_{i=1}^{|U|} \sum_{j=1}^{|V|} \frac{|U_i \cap V_j|}{N} log \frac{N|U_i \cap V_j|}{|U_i||V_j|}. \tag{2.9}$$

This metric is independent of the absolute values of the labels: a permutation of the class or cluster label values won't change the score value in any way. This metric is furthermore symmetric and can be useful to measure the agreement of two independent label assignment strategies on the same data set when the real ground truth is not known. Normalized Mutual Information (NMI) [13] is a normalization

of the Mutual Information (MI) score to scale the results between 0 (no mutual information) and 1 (perfect correlation).

Adjusted Mutual Information (AMI)

The baseline value of mutual information between two random clusterings tends to be larger when the two partitions have a larger number of clusters (with a fixed number of nodes). Hence Adjusted Mutual Information (AMI) [18] will be able to adjust the Mutual Information (MI) score to account for chance. The AMI between clustering U and V is given as:

$$AMI(U,V) = \frac{MI(U,V) - E\{MI(U,V)\}}{max\{H(U),H(V)\} - E\{MI(U,V)\}}. \tag{2.10}$$

This metric is independent of the absolute values of the labels: a permutation of the class or cluster label values won't change the score value in any way. Here H(U) and H(V) indicate the entropy associated with the partitioning U and V. The AMI takes a value of 1 when the two partitions are identical and 0 when the MI between two partitions equals the value expected due to chance alone.

2.2.5 Evaluating Cluster Results

Once perform the cluster analysis we have to evaluate the cluster result. In this study, we used silhouette scores to find the goodness of clustering techniques.

Silhouette Score

Silhouette score is used to observe the separation distance between the resulting clusters [12]. This measures how close each point in one cluster is to points in the neighboring clusters. To calculate the Silhouette score for each observation, the following distances need to be calculated:

1. Mean distance from the observation to all other observations in the same cluster. Let's denote it by D_{in}.

2. Mean distance from the observation to all other observations in the nearest cluster. Let's denote it by D_{out}.

After calculating the above two distances, the silhouette score, **S**, for each sample is calculated using the following formula:

$$S = \frac{(D_{out} - D_{in})}{max(D_{in}, D_{out})}. \tag{2.11}$$

The silhouette score varies from -1 to 1, where 1 means clusters are clearly distinguished and -1 means clusters are assigned in the wrong way. The value 0 means the distance between clusters is not significant.

2.2.6 Zones and Clusters

In the tested building, each floor had multiple thermostats which were connected to variable air volume (VAV) terminal units. VAV terminal units are zone-level flow control devices. Thermostats that are connected to one VAV unit are controlled by

that VAV unit and can be considered as one zone. Based on the number of VAV units, we could identify different numbers of zones. After the inspection, we recognized that there are 8, 10, and 10 zones on floor 1, floor 2, and floor 4 respectively. We considered those numbers of zones as the numbers of clusters that we want from the cluster analysis, and compared those cluster results with the actual zones. It helped us identify whether those sensors within zones collect similar data or not, and whether having one thermostat for each zone is reasonable or not.

We clustered building floors based on k-means and two different time series clustering algorithms. From each clustering method, three different results were generated using temperature, temperature + relative humidity, and temperature + relative humidity + pressure. Then, the similarity between zone labels and cluster labels is compared using the ARI, NMI, and AMI similarity matrices. These similarity measures are mainly used to do pairwise comparisons of two cluster results.

Then to determine the minimum number of thermostats, we found the optimal number of clusters using three clustering methods with different variables as discussed earlier. Silhouette score was used to evaluate those cluster results and identify the best method to cluster the building environment.

2.2.7 Enhanced Time Series Clustering using Community Detection

In this section, we introduce an enhanced time series clustering technique based on community detection from network analysis. Before explaining the details of our method, we first give some of the theoretical background that helps to understand

our approach. The following subsections will cover the basic concepts of social network analysis, including nodes and edges, edge weights, and the main community detection algorithms we used in this study: Louvain and Girvan-Newman. We will also discuss the novel time series clustering method using community detection.

Social Network Analysis

Social network analysis (SNA) [19] is a methodological approach used to study relationships and structures within a set of entities, often represented as graphs. It provides insights into how entities interact and the patterns that emerge from these interactions.

A social network graph is created using nodes and edges to represent the actors and their relationships respectively. Mathematically, a network can be represented as follows, where two actors are denoted by i and j. By generating Y_{ij} values for the entire network with n actors, we can construct an adjacency matrix, denoted as $Y \equiv \lfloor Y_{ij} \rfloor_{n \times n}$.

Here, Y_{ij} is defined as:

$$Y_{ij} = \begin{cases} 1 & \text{if there is a relationship between } i \text{ and } j. \\ 0 & \text{otherwise.} \end{cases} \tag{2.12}$$

The adjacency matrix Y represents the network, indicating the presence or absence of relationships between all pairs of actors. In this matrix, 1 indicates the presence of a relationship, while 0 indicates no relationship. This matrix will form the basis for studies working with the network's structure and act as a building block for most algorithms on community detection.

As we focus on analyzing time series data from IoT sensors in a building, we consider a time series a node in a network. The relationships between these time series are based on their similarity, which can reveal important information about indoor environmental conditions.

Mathematically, we can represent a graph G as follows:

$$G = (V, E)$$

, where V is the set of nodes and E is the set of edges. The connections can be directed or undirected, depending on whether the relationship has a specific direction (e.g., influence) or not.

Edge Weight

In network analysis, the weight of an edge indicates the strength of the relationship between two nodes. Among various time series similarity measures, including Euclidean distance and Dynamic Time Warping (DTW) [16], cross-correlation [4] gave the most effective results. Hence, for the time series clustering method, we propose using autocorrelation to measure edge weights.

Mathematically, the autocorrelation for a time series X_t can be defined as:

$$\rho(k) = \frac{Cov(X_t, X_{t-k})}{Var(X_t)}, \tag{2.13}$$

where $\rho(k)$ is the autocorrelation at lag k, Cov is the covariance, and Var is the variance. A high autocorrelation value at a specific lag indicates that the time series values at that lag are similar, implying a strong relationship between them.

When constructing our graph, we set the weight for edges between time series nodes according to the value of their pairwise autocorrelation. If two time series have high autocorrelation with similar lag values, it signifies a strong temporal relationship, and thus, a heavier edge is assigned. Conversely, if two series have a low autocorrelation, then the weight on an edge will be light, indicating that the link between these points could be weak.

Community Detection Algorithms

Community detection algorithms are used to identify groups of nodes that are more densely connected than nodes outside the group. The Louvain method [3] and the Girvan-Newman algorithm [6] are prominent algorithms that are mainly used in community detection studies.

The Louvain method is a widely used technique for community detection in large networks. It optimizes the modularity of a network, which measures how effectively a network is divided into communities. The algorithm operates in two phases: initially, it assigns each node to its own community, and then it iteratively merges communities to maximize modularity. Mathematically, the modularity Q can be expressed as:

$$Q = \frac{1}{2m} \sum_{i,j} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta(c_i, c_j), \tag{2.14}$$

where A_{ij} is the adjacency matrix, k_i and k_j are the degrees of nodes i and j, m is the total number of edges in the network, and $\delta(c_i, c_j)$ is 1 if nodes i and j are in the same community and 0 otherwise. By maximizing Q, the Louvain method

identifies densely connected subgraphs, thereby revealing the underlying community structure.

In contrast, the Girvan-Newman method takes a different approach to community detection by systematically removing edges from the original network. It calculates the betweenness centrality of each edge, which indicates the number of shortest paths that pass through that edge. The algorithm removes the edge with the highest betweenness centrality, breaking the network into smaller components. This removal continues until all edges are gone, allowing communities to be identified based on the resulting connected components. The betweenness centrality $C_B(e)$ for an edge e can be calculated as:

$$C_B(e) = \sum_{s \neq t \in V} \frac{\sigma_{st}(e)}{\sigma_{st}}, \tag{2.15}$$

where σ_{st} is the total number of shortest paths from node s to node t, and $\sigma_{st}(e)$ is the number of those paths that pass through edge e. The Girvan-Newman method effectively uncovers community structures by analyzing the relationships and dependencies between nodes as edges are systematically removed.

2.2.8 Sensor Network Creation Algorithm

In this study, we presented a novel approach to cluster time series data using community detection techniques on a sensor network created from temperature data. The first step in this methodology involves creating a sensor network based on the pairwise maximum cross-correlation of the time series data collected from various sensors throughout the building.

To create the connections between the sensors, we calculated the cross-correlation for each pair of sensors. Cross-correlation assesses the similarity of two-time series as a function of the time lag applied to one of them. By analyzing cross-correlation, we can identify which sensors display similar temperature patterns over time. This information is essential for creating an adjacency matrix that illustrates the network of sensors.

The adjacency matrix A is generated by setting a threshold value for each sensor pair. Finding a suitable threshold was a challenge. We tried various measures and methods, including the mean of the correlation matrix, minimizing Dynamic Time Warping (DTW) within groups, and maximizing it between groups, as well as minimizing cross-correlation within groups and maximizing it between groups. Ultimately, based on our findings, we decided to use the mean of cross-correlation as our threshold. If the maximum cross-correlation value between a pair of sensors exceeds this threshold, we consider it a strong relationship and create an edge between those sensors in the network. The algorithm 1 outlines the process for creating the adjacency matrix.

After creating the adjacency matrix, we consider each sensor as a node in a graph, with edges connecting those that are similar based on our analysis. This sensor network serves as the foundation for our community detection algorithms, which will help improve the clustering of the time series data by grouping sensors with similar patterns.

Algorithm 1: Create Adjacency Matrix for Network Generation

Input: Time series data for each sensor (T_i)
Output: Adjacency matrix (A)
Initialize an empty A
for each pair of sensors (i, j) **do**
    Compute pairwise cross-correlation for T_i and T_j
    $MAX \leftarrow \max(\text{cross-correlation}(i, j))$
    $t \leftarrow \text{mean}(\text{cross-correlation}(i, j))$
    if $MAX > t$ **then**
        $A[i, j] \leftarrow 1$
    else
        $A[i, j] \leftarrow 0$
    end if
end for
Return: The adjacency matrix A

2.3 Results

2.3.1 Descriptive Analysis

Before starting the cluster analysis we analyzed the temperature, humidity, and pressure trends of each sensor to identify any anomalies and to understand the building's current condition. For an example, Figure 2.2 shows temperature variations over time on each sensor. Based on ASHRAE (American Society of Heating, Refrigerating, and Air-Conditioning Engineers) 55 standards [2], the comfortable temperature range is between 19°C and 24°C, illustrated by the highlighted region. The majority of temperature values were measured within this comfortable range, except sensors 16 and 17, both of which were located in the same office.

Then we averaged the temperatures on each floor, per location to illustrate the temperature trends over one average day. Figure 2.3 appears to indicate temperature trends corresponding with occupancy, beginning to increase at 09:00 am and

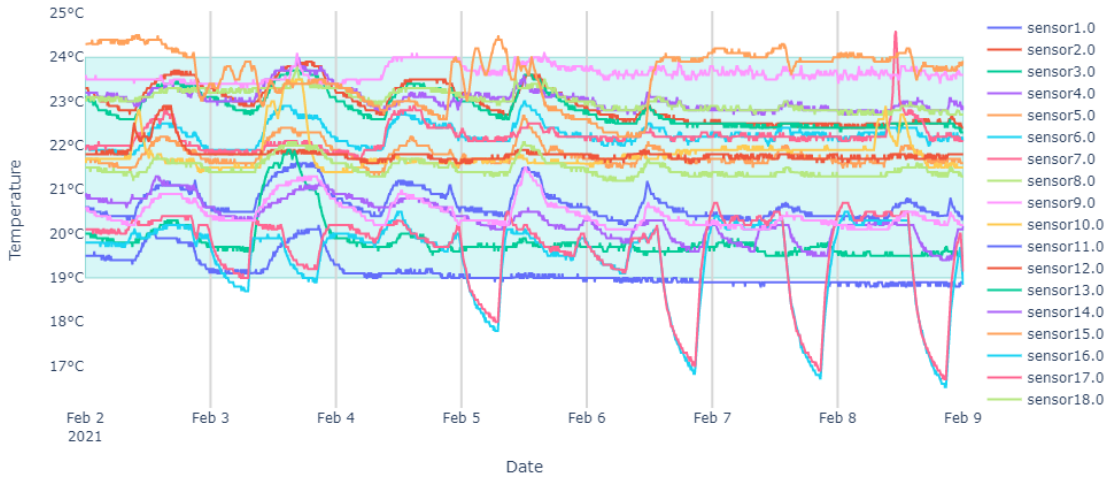


Figure 2.2: Temperature variations over the entire test period for floor 1. The green color box shows the comfortable temperature range.

decreasing in the afternoon.

Relative humidity (RH) data of all the sensors within each floor showed similar variations with time. Hence we compared floor-wise RH variations during the test period. Figure 2.4 highlights that floor 1 is typically more humid than floor 2 and floor 4. Based on the ASHRAE 55 standards, a comfortable RH range is between $20\% - 60\%$, but this building showed very low RH values.

The average air pressure per floor will depend on many factors, specifically outside weather conditions. When the weather is cold outside, the warm buoyant indoor air tends to rise to the top of the building generating increased pressure in the upper levels, and relatively low pressure in the lower levels. This is known as the stack effect [11], [10]. Based on the collected air pressure data of this building, Figure 2.5 shows the floor-wise pressure distribution, and it indicates that floor 1

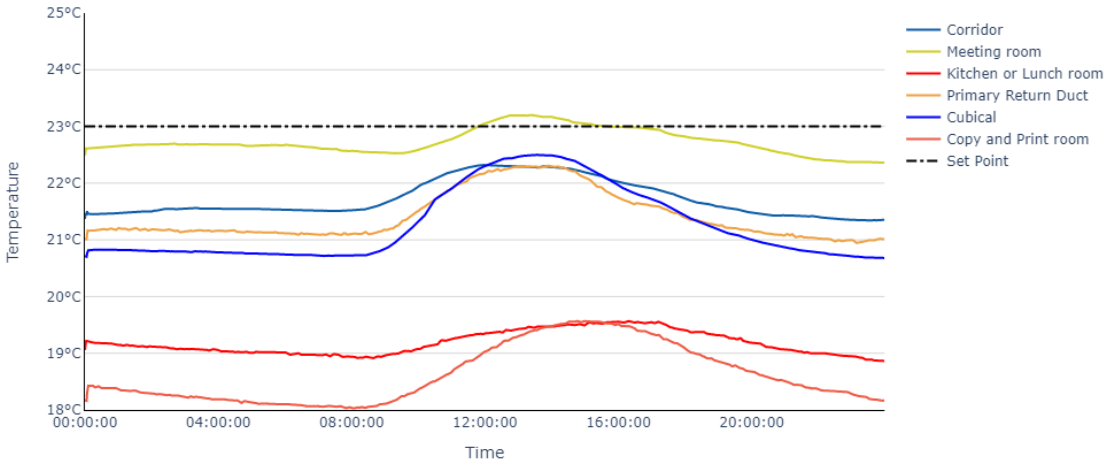


Figure 2.3: Average temperature over one average day for floor 4. Dash line shows the set point (expected Temperature) of this floor.

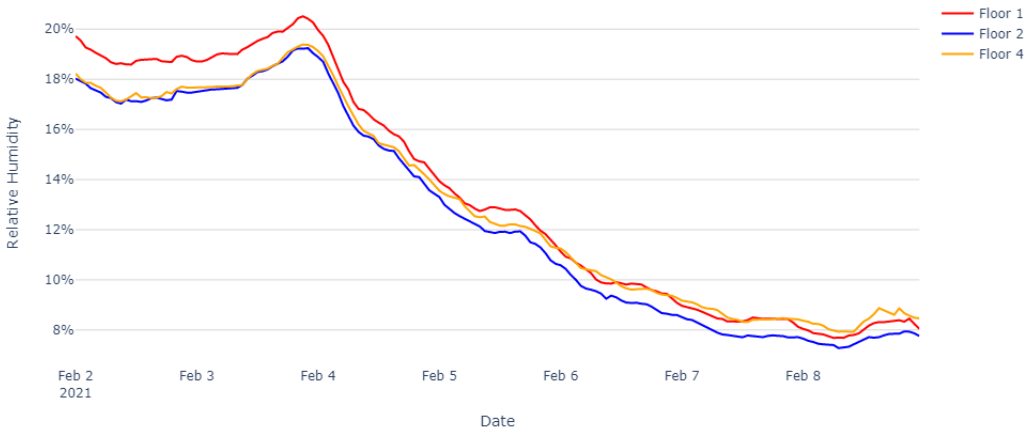


Figure 2.4: Average relative humidity per floor, over the entire test period. The building shows very low relative humidity (RH) values based on ASHRAE 55 standards.

has consistently higher pressure than floor 2 and floor 4. This may represent the reverse stack effect.

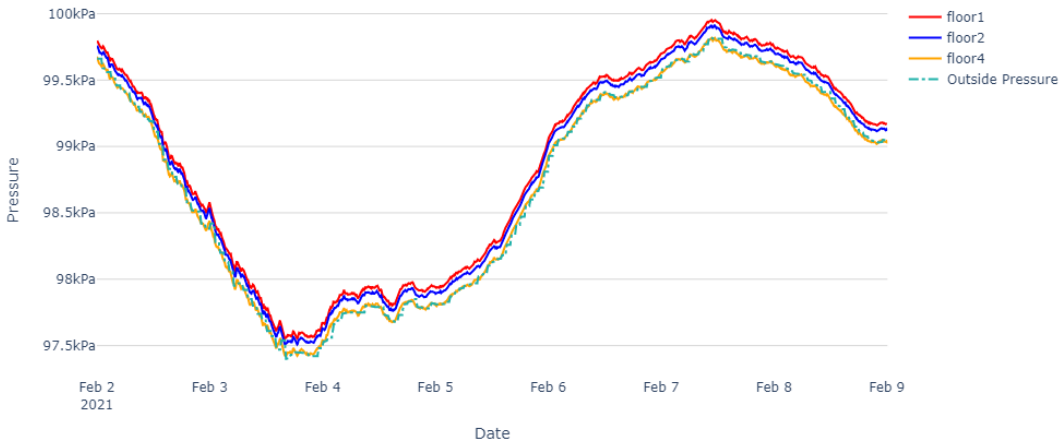


Figure 2.5: Average pressure per floor, over the entire test period with outside pressure.

2.3.2 Clustering Based on Number of Zones

As discussed in Section 2.2, we clustered each floor using three different methods by considering three different variables. We clustered sensor data from the ground floor, second floor, and fourth floor separately. For example the following Figure 2.6 - 2.8 illustrate the k-means clustering results based on temperature in three different floors. Different zones are indicated by black color borders. The legend of each floor map shows the mean temperature of each cluster from lowest to highest. Blue color nodes indicate the sensors with the lowest mean temperature, while red color nodes indicate the sensors with the highest mean temperature.

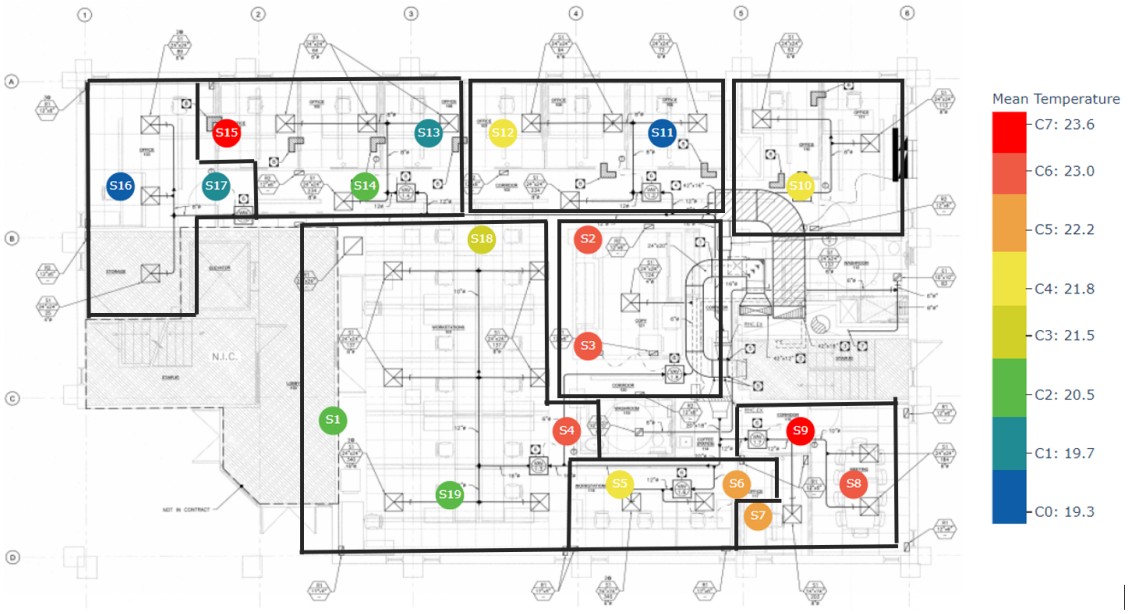


Figure 2.6: K-means cluster results with zones in floor 1. Black boxes indicate zones while different colors indicate each cluster. The legend shows the mean temperature values of each cluster

Figure 2.7: K-means cluster results with zones in floor 2. The legend shows the mean temperature values of each cluster

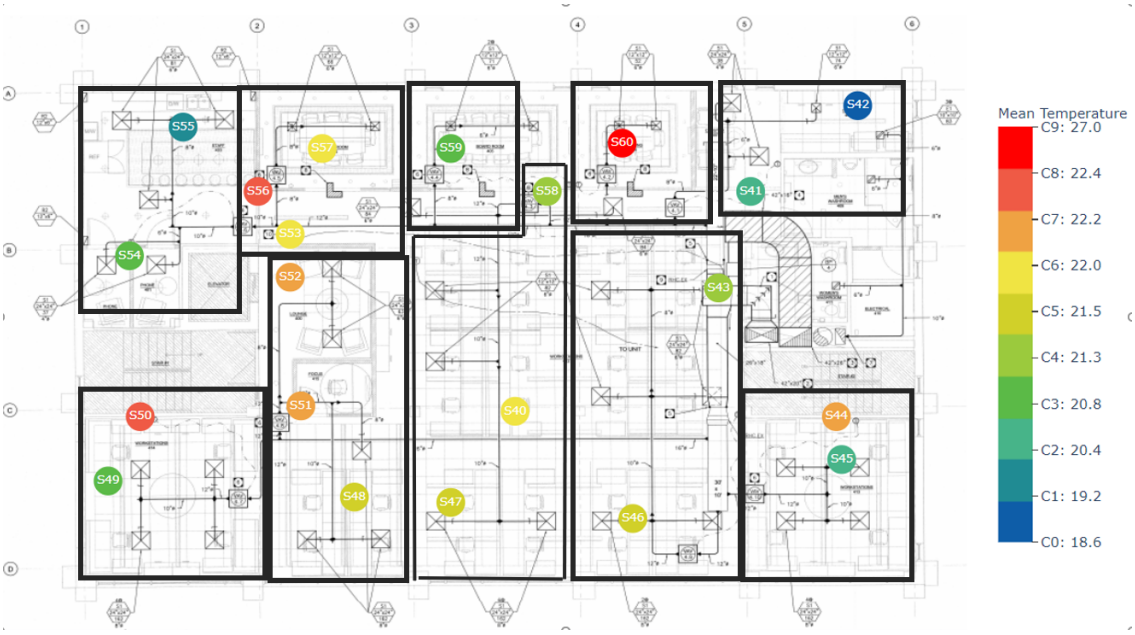


Figure 2.8: K-means cluster results with zones in floor 4. The legend shows the mean temperature values of each cluster

Then we calculated similarity matrices to measure the similarity between the zone labels and cluster labels. Table 2.3 shows the ARI, NMI, and AMI scores for each cluster results which were generated by different clustering algorithms.

2.3.3 Clustering Based on Optimal Number of Clusters

After generating clusters based on the number of zones within each floor, we considered clustering based on the optimal number of clusters. In k-means and time series clustering methods, we used elbow plot, and silhouette scores to identify the optimal number of clusters, respectively. Table 2.4 shows the optimal number of clusters and silhouette score for each clustering algorithm.

Table 2.3: Similarity scores when comparing zoning labels with cluster results. The font colors blue, red, and green indicate the maximum value of ARI, NMI, and AMI respectively.

Variables	Floor	K-means clusters			Time-series clusters (Ward, Euclidean)			Time-series clusters (Average, Correlation)		
		ARI	NMI	AMI	ARI	NMI	AMI	ARI	NMI	AMI
Temperature	Floor1	0.03	0.61	0.05	0.09	0.63	0.12	0.12	0.61	0.20
	Floor2	0.004	0.68	0.01	0.25	0.75	0.28	0.02	0.65	0.03
	Floor4	0.07	0.69	0.09	0.22	0.73	0.25	0.11	0.69	0.18
Temperature Humidity	Floor1	0.07	0.62	0.10	0.18	0.70	0.27	0.24	0.71	0.38
	Floor2	-0.08	0.64	-0.10	0.06	0.69	0.08	0.12	0.69	0.25
	Floor4	0.08	0.66	0.10	0.13	0.68	0.15	0.07	0.66	0.11
Temperature Humidity Pressure	Floor1	0.06	0.61	0.09	0.24	0.72	0.31	0.17	0.65	0.26
	Floor2	-0.09	0.63	-0.11	0.08	0.69	0.18	0.07	0.68	0.11
	Floor4	0.03	0.66	0.10	0.13	0.68	0.15	0.19	0.72	0.27

Table 2.4: Optimal number of clusters and silhouette score for each clustering method.

Variables	Floor	K-means clusters		Time-series clusters (Ward, Euclidean)		Time-series clusters (Average, Correlation)	
		Optimal clusters	Silhouette	Optimal clusters	Silhouette	Optimal clusters	Silhouette
Temperature	Floor1	3	0.66	3	0.78	5	0.71
	Floor2	4	0.61	3	0.62	3	0.68
	Floor4	4	0.58	2	0.88	3	0.61
Temperature Humidity	Floor1	4	0.46	2	0.83	2	0.41
	Floor2	3	0.40	3	0.68	4	0.60
	Floor4	4	0.55	2	0.85	2	0.64
Temperature Humidity Pressure	Floor1	6	0.27	2	0.72	2	0.41
	Floor2	4	0.44	2	0.65	2	-0.05
	Floor4	4	0.36	2	0.85	4	0.54

2.3.4 Clustering Based on Community Detection

In order to improve the current time series clustering method we applied community detection techniques from social network analysis, as described in Section 2.2.7. In Figure 2.9, we can observe three distinct patterns among those time series: one abnormal time series, several straight time series, and others exhibiting seasonal patterns. To identify which clustering method detects these clear groupings, we applied hierarchical clustering based on correlation, as well as the Louvain [3] and Girvan-Newman [6] clustering methods.

First, we generated the network based on the algorithm that we explained in

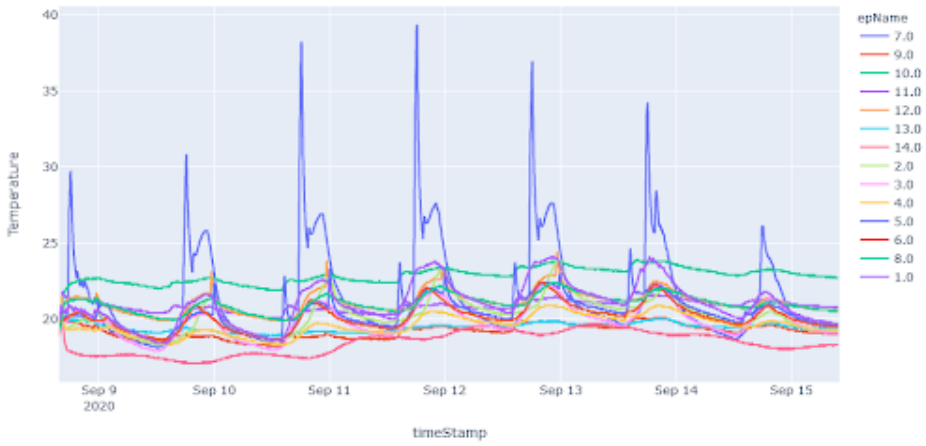


Figure 2.9: Temperature time series for floor 4 of a commercial building.

Section 2.2.8. After applying this algorithm, we obtained the graph structure shown in Figure 2.10. Next, we applied the Louvain and Girvan-Newman community detection methods to this network, and the results are presented in the same figure.

The results show that the community detection method works better than hierarchical clustering for finding clusters. Some straight time series were grouped with seasonal patterns in the time series clustering results. For example, Sensor 7, which behaves abnormally, is clearly marked as an isolated node in the network graph, making its anomaly easy to see. This method successfully separates time series with straight patterns from those with seasonal patterns. To investigate this further, we used a few other datasets and compared how well the clusters worked using the silhouette score. Based on the results in table 2.5, it is clear that the Louvain-based clustering method works better than others.

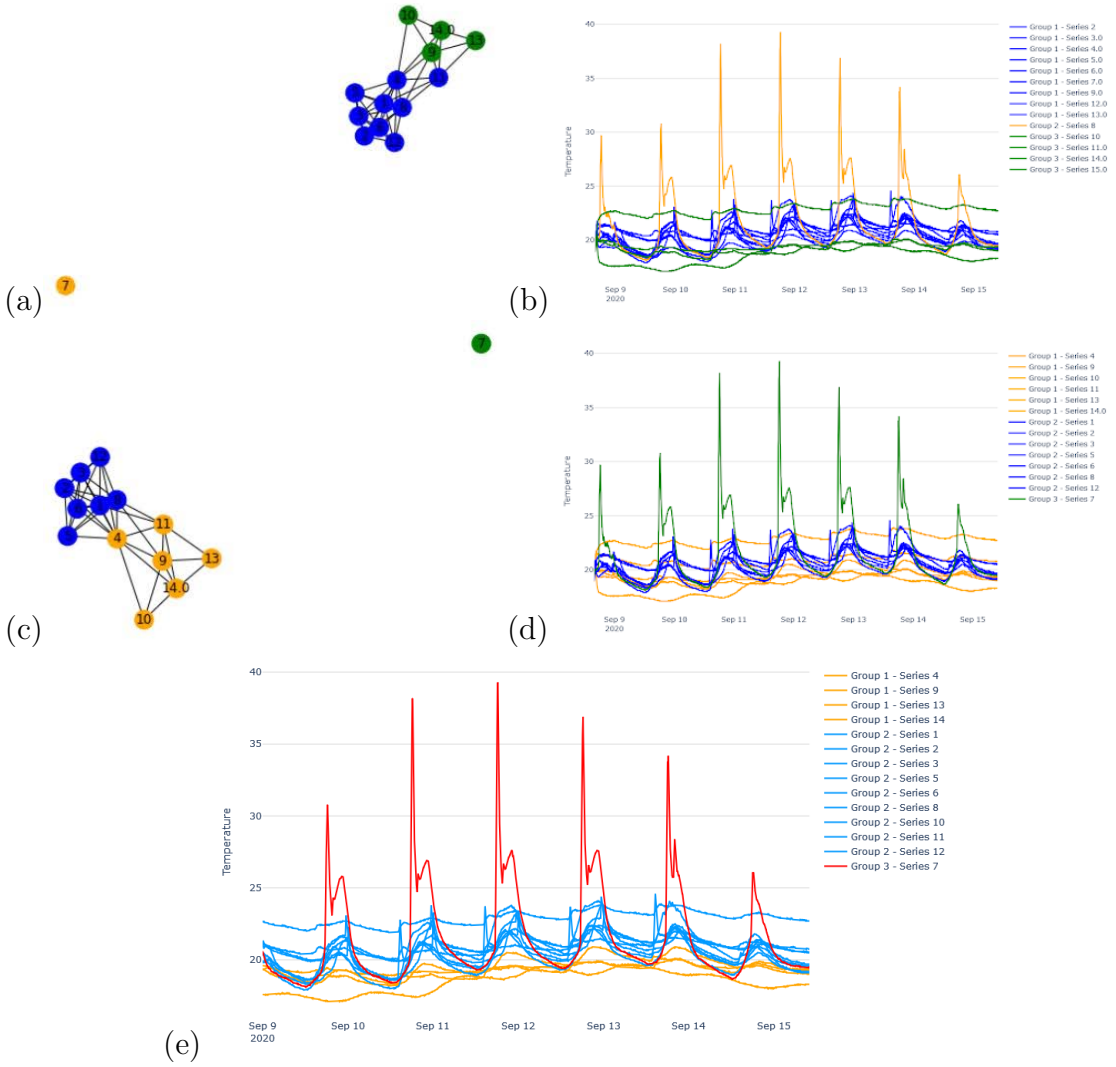


Figure 2.10: (a): Identified communities from Girvan Newman (b): Time series clusters from Girvan Newman, (c): Identified communities from Louvain (d): Time series clusters from Louvain, (e): Time series clusters from Hierarchical clustering

2.4 Discussion

Based on the results of cluster analysis with zones, it is clear that time-series cluster results have a better agreement with zones than the k-means clusters. When comparing two methods of time series clustering, the results of Ward's method and

Dataset	Clustering Method	t or k	Silhouette Score	Dataset	Clustering Method	t or k	Silhouette Score
Ground	Time series clustering (cross-correlation)	k = 2	0.5037	Floor 3	Time series clustering (cross-correlation)	k = 3	0.3610
	Girvan	t = 0.7	0.5152		Girvan	t = 0.83	0.3529
	Louvain	t = 0.7	0.5725		Louvain	t = 0.83	0.3610
Floor 4	Time series clustering (cross-correlation)	k = 3	0.5129	1715 School	Time series clustering (cross-correlation)	k = 4	0.3834
	Girvan	t = 0.83	0.5129		Girvan	t = 0.78	0.5866
	Louvain	t = 0.83	0.5659		Louvain	t = 0.78	0.3834

Table 2.5: Cluster evaluation and comparison

Euclidean distance method have higher similarity scores than the results of the average and correlation methods. However, the overall results show small similarity scores between clustering and zoning, and there can be multiple reasons to have small similarity score values. One reason is that there can be some lurking, unmeasured variables that affect the clustering, and another reason is that zoning has not been done properly.

When discussing the results of the optimal number of clusters, it is clear that we can have a smaller number of clusters with sensors that show similar performances. Hence, for this building, we could easily reduce the number of thermostats. Here, we can also see that the time-series clustering method clustered the sensors better than k-means clustering, and that Ward's method and Euclidean distance method showed better performance between the two time-series clustering methods. Clustering using only the temperature showed better results than combining humidity and pressure with temperature.

Finally, we developed a novel method to improve the results of time series clustering. A key aspect of our approach is network creation, where sensors with similar time series patterns are connected through edges. Our results show that this novel method outperforms traditional time series clustering and the Louvain-based community

detection method is better than the Girvan-Newman technique, yielding more accurate and meaningful clusters.

For future work, we plan to collect carbon dioxide (CO2) in addition to temperature measurements to evaluate the indoor air quality. Air quality tends to decrease as CO2 values increase, which has a negative impact on an occupant's health. Given that humans exhale CO2, there exists a connection between occupancy and measured CO2 values. Airflow (ventilation) exhausts the breathed air with high CO2 concentration, supplying fresh air with low CO2 concentration. Temperature and therefore airflow into a space, are controlled in part by thermostats, and by evaluating both CO2 and temperature together, we can determine effective clustering based on temperature (building efficiency) and ventilation (occupant health).

Bibliography

[1] J. G. Allen and J. D. Macomber. *Healthy buildings: how indoor spaces drive performance and productivity.* Harvard University Press, 2020.

[2] ASHRAE. *ASHRAE/ANSI Standard 55-2017 Thermal environmentalconditions for human occupancy.* American Society of Heating, Re-frigerating, and Air-Conditioning Engineers: Atlanta, GA., 2017.

[3] V. Blondel, J. L. Guillaume, et al. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics Theory and Experiment*, page P10008, 2008.

[4] P. J. Brockwell and R. A. Davis. *Introduction to Time Series and Forecasting.* Springer, New York, 3rd edition, 2016.

[5] N. Frank. *Introduction to HPC with MPI forData Science.* Springer, 2016.

[6] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.

[7] J. A. Hartigan and M. A. Wong. Algorithm as 136: A k-means clustering

algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28:100–108, 1979.

[8] H. Lawrence and A. Phipps. Comparing partitions. *Journal of Classification*, 2:193–218, 1985.

[9] Y. Y. Lee, Y. H. Lee, S. Mohammad, P. N. Shek, and C. K. Ma. Thermal characteristics of a residential house in a new township in johor bahru. *IOP Conference Series: Materials Science and Engineering*, 271:012027, 2017.

[10] S. Mijorski and S. Cammelli. Stack effect in high-rise buildings: A review. *International Journal of High-Rise Buildings*, 5:327–338, 2016.

[11] P. Miller. Stack effect: Why it was so difficult to stay warm this winter. https://www.hendersonengineers.com/insight_article/stack-effect-why-it-was-so-difficult-to-stay-warm-this-winter, 2019. Accessed 20 June 2021.

[12] J. R. Peter. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[13] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery. *Numerical Recipes: The Art of ScientificComputing (3rd ed.)*. New York: Cambridge University Press, 2007.

[14] C. A. G. Santos, R. M. Brasil Neto, R. M. da Silva, and S. G. F. Costa. Cluster analysis applied to spatiotemporal variability of monthly precipitation over

paraíba state using tropical rainfall measuring mission (trmm) data. *Remote Sensing*, 11(6), 2019.

[15] M. Sarnovsky and D. Bajus. Building environment analysis based on clustering methods from sensor data on top of the hadoop platform. In *2017 IEEE 15th International Symposium on Applied Machine Intelligence and Informatics (SAMI)*, pages 000079–000082, 2017.

[16] P. Tormene, T. Giorgino, S. Quaglini, and M. Stefanelli. Matching incomplete time series with dynamic time warping: An algorithm and an application to post-stroke rehabilitation. *Artificial Intelligence in Medicine*, 45:11–34, 2008.

[17] Y. Unal, T. Kindap, and M. Karaca. Redefining climate zones for turkey using cluster analysis. *International Journal of Climatology*, 23:1045 – 1055, 2003.

[18] N. X. Vinh, J. Epps, and J. Bailey. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *Journal of Machine Learning Research*, 11:2837–2854, 2010.

[19] S. Wasserman and K. Faust. *Social Network Analysis: Methods and Applications*. Structural Analysis in the Social Sciences. Cambridge University Press, 1994.

[20] J. Yun and K. Won. Building environment analysis based on temperature and humidity for smart energy systems. *Sensors (Basel, Switzerland)*, 12:13458–13470, 2012.

Chapter 3

Hotspot Analysis in Buildings using Moran's I Statistic

Hotspot analysis identifies areas with a higher concentration of events compared to the expected number in a given random distribution of events. Over the past few decades, hotspot analysis has been widely used in various research disciplines, including public health, crime, and environmental quality. In this chapter, we applied hotspot analysis within buildings using temperature data from wireless sensors and floor map coordinates, eliminating reliance on Geographic Information Systems (GIS) and expanding the scope of indoor environmental assessment in situations where geographical data may not be available. A key innovation of our study is Moran's I-based method to optimize the number of neighbors in the k-nearest neighbors analysis. This enhances the accuracy of hotspot identification, a critical advancement in spatial data analysis. The temporal Hotspot analysis, using the collected building data over time, reveals how factors such as vacancy and window orientation influence the creation of hot and cold spots within a building. Random forest was employed to assess the importance of these factors. In conclusion, our combined application

of Moran's I statistic and hotspot analysis effectively identifies hot and cold areas within a building, when the number of neighbors is accurately identified and the absence of outliers or abnormal sensors.

3.1 Introduction

Over the last decades, many researchers have been interested in analyzing indoor thermal data to identify building issues, to improve the energy use intensity and efficiency of buildings as well as occupant comfort levels [5, 68]. Issues with either the building's construction, or its heating, ventilation, and air conditioning (HVAC) system may lead to certain areas experiencing notably high or low thermal conditions. Identifying these conditions can be challenging, time-consuming, and costly. It would be helpful if we could use a statistical method to find these specific areas. Some researchers have focused on identifying temperature clusters within buildings to recognize areas that exhibit similar temperature patterns [63]. However, this approach falls short in distinguishing high-temperature and low-temperature clusters while integrating spatial autocorrelation. Hence, hotspot analysis [2], which is explained in Section 3.2.2, emerges as a robust solution to identify areas with statistically significant thermal concentrations compared to their surroundings.

Identification of high and low-temperature spatial clusters, known as 'Hot' and 'Cold' spots, within a building is a practical approach to enhancing energy efficiency and ensuring occupant comfort. Efficiently mixing supply air throughout large floor plans can yield a notable decrease in energy consumption, typically around 3.5 percent [52]. However, areas experiencing significant temperature variations impose

additional strain on a building's HVAC systems, compelling them to operate at heightened levels to maintain consistent temperatures. The continued prevalence of such hot and cold spots can also indicate underlying HVAC system issues. Several factors related to improper HVAC system performance can contribute to hot and cold spots within a building. Solar heat gain, electric equipment that releases heat, blocked air vents, dirty air filters, bad thermostat location and issues with ducting can be considered as some of those factors [21]. Identifying these spots makes troubleshooting easier, streamlining the diagnostic process without the need for labor-intensive manual inspections.

Hotspot analysis has been used in many research areas including crime monitoring, epidemiology, animal behavior, and environmental quality [69, 12, 39, 26, 34]. These analyses pinpoint highly clustered geolocations, based on specific features. For instance, Marson and colleagues utilized hotspot analysis to identify burglary hotspots in the Northeast Penang Island District and Kuching District [36]. Another two studies were done to identify the highly clustered locations with cervical cancer patients [35] and COVID-19 patients [53]. Some other interesting applications include locating hot spots of forest loss [24] and identifying older adults at high risk of stress [55].

Many researchers have used temperature as a feature to identify hot and cold spots. Kumari and her team have used hotspot analysis to study spatial patterns of land surface temperature in an urban city in India [27]. Another two research teams analyzed the urban heat island effect of a city in China and in Italy [67, 22]. All

these researches were done by using geolocation information (longitude and latitude) and considering the open locations [29]. However, the factors that affect the heat of an indoor space might be different from those of outdoor spaces. Therefore, in our study, we focused on the indoor environment and identifying hot and cold spots without relying on geographical data or GIS software.

By conducting a hotspot analysis, facilities managers can obtain a more precise evaluation of a building's temperature levels. This allows them to pinpoint deviations from setpoints or mean temperatures, enabling efforts to ensure a more consistent indoor environmental quality (IEQ) for all occupants without needing to rely on inaccurate or subjective data. Currently building engineers and facilities staff use thermal images to detect hotspots and issues within the building [58]. However, instead of relying on costly or inaccessible thermal imaging, building engineers and facilities staff can utilize Internet of Things (IoT) sensors to collect basic temperature data, providing a cost-effective means of identifying hot and cold spots.

In this study, we aim to address the existing knowledge gap in indoor environmental assessment methodologies by applying hotspot analysis within buildings and proposing a novel approach to enhance the accuracy of hotspot identification through Moran's I-based method. In the literature, no study has applied hotspot analysis to find hot and cold spots inside buildings. Furthermore, we investigated the factors influencing hotspots within buildings. To achieve this, we applied hotspot analysis on temperature data collected from two buildings. We introduce a new method to determine the optimal number of neighbors for the k-nearest neighbors algorithm

[14], which significantly improves the accuracy of hotspot analysis. Finally, we used Random Forest [10] to identify the key features contributing to the formation of these spots. These variables help us understand why hot and cold spots occur in the identified locations.

This chapter is structured into four main sections. Section 3.1 provides an introduction to the study, highlighting the knowledge gap in indoor environmental assessment methodologies. Section 3.2 details the methodologies, including data collection and preprocessing steps, theories underlying hotspot analysis, and optimizing the number of neighbors. In Section 3.3, we present the results of our experimental study, including the application of hotspot analysis to temperature data from commercial and school buildings in Winnipeg, Canada. Section 3.4 discusses the implications of our findings, and offers insights into the factors influencing hotspots within buildings.

3.2 Material and Methods

3.2.1 Data Collection and Preprocessing

The data collection was done in a commercial building in downtown Winnipeg. We selected the fourth floor of this building, and fourteen (14) sensors were strategically positioned within the building. Data were collected in five-minute intervals from September 9, 2020, to September 14, 2020. The sensors collected data in SI units, where the possible imperial unit equivalent has been soft converted and indicated in parenthesis. We also collected data from a school building, to show different results

of hotspot analysis based on different sensor positions. In order to apply hotspot analysis, we needed only one data point for each sensor. Hence we calculated the median temperature of each sensor to have an overall idea about the locations.

The other important data type that we wanted to collect was geographic information. This was a challenge since the sensors used for this test did not collect longitude and latitude data of their placement coordinates. Therefore, the solution that we came up with was collecting (x,y) coordinate values of the sensor placements based on a provided floor map image. Hence having a georeferenced floormap was important for this analysis. To collect (x,y) coordinates by clicking on the map, we used the 'cv2' package in python [9]. For example, Figure 3.1 shows the (x,y) coordinates of some sensors on the floor map.

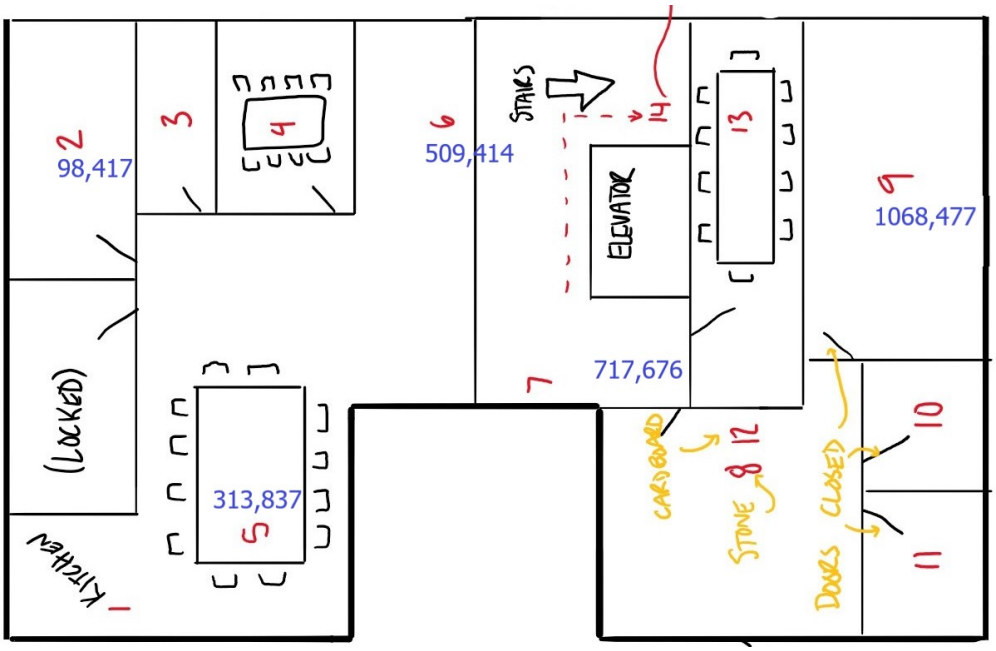


Figure 3.1: (x,y) Coordinates of some sensors on a floor map using cv2 package

However, these coordinates could not be used in their initial output format for successful hotspot analysis. To convert them into geo points we had to divide them by 100 and add decimal points to the coordinate values. After pre-processing the coordinates and collecting the temperature values of each sensor we ended up with an acceptable data frame for our analysis. Table 3.1 shows a sample of the final data set. We also applied hotspot analysis on 5-minute interval data to identify the hot and cold spots within a building at each time interval. For that, we added a column, which represents time, to Table 3.1.

Table 3.1: Sample of sensor data set

Sensor Number	**Temperature**	**x**	**y**
1	26.4°C	3.98	1.24
2	20.3°C	8.29	0.74
3	20.4°C	8.62	1.72
4	20.7°C	8.60	2.81

3.2.2 Hotspot Analysis of The Indoor Temperature of A Building

Hotspot analysis is a way to figure out if certain areas have more or fewer events compared to the surrounding areas. In this process, the first step is to assess spatial autocorrelation to understand how similar values are in nearby locations. This step helps determine if there is a spatial pattern in the data. Then we apply a statistical test to determine the significance of the observed pattern. Here we have used Anselin Global Moran's [38] I as it is the most suitable tool for understanding overall spatial autocorrelation. Finally, by utilizing Local Indicators of Spatial Association (LISA) [4], we were able to identify hotspots (high values surrounded by high values)

and coldspots (low values surrounded by low values).

Spatial Autocorrelation

Spatial autocorrelation is a fundamental concept in spatial analysis that describes the degree to which neighboring locations in a spatial dataset share similar attribute values. The principle behind spatial autocorrelation lies in the idea that things in space aren't just randomly scattered; rather, they tend to form patterns and connections with nearby entities. When analyzing spatial data, understanding spatial autocorrelation is essential as it helps identify the presence of clustering, dispersion, or randomness within the dataset. Moreover, spatial autocorrelation offers several advantages in spatial analysis and decision-making. One of its primary benefits is its ability to enhance hotspot analysis results by accounting for spatial relationships among observations. If the variable of interest is distributed randomly among all locations, there will be no spatial relationship. Spatial autocorrelation can have positive or negative values: positive values indicate that similar values are clustered together, while negative values suggest that similar values are dispersed.

Measures of spatial autocorrelation can be categorized as global and local indicators of spatial association. Global indicators give a broad understanding of spatial autocorrelation across the entire study area, while local indicators provide insights into localized patterns. The Global Moran's I [38] statistic or global spatial autocorrelation is the first step to be carried out in identifying spatial patterns of observations. The global Moran's I index value is within a range of -1.0 to +1.0. This index is defined as follows: if close to +1, it indicates perfect clustering; if close

to -1, it suggests perfect dispersion; and if equal to 0, there is no spatial correlation, indicating a randomly distributed pattern.

According to our study, the null hypothesis of global Moran's I is that temperature is randomly distributed among the sensors in our study area. Our alternative hypothesis is that the temperature is distributed in a pattern among the sensors. The moran's I statistic can be defined as follows;

$$I = \frac{n}{W} \frac{\sum_{i=1}^{n} \sum_{j=1}^{n} W_{i,j}(X_i - \bar{X})(X_j - \bar{X})}{\sum_{i=1}^{n}(X_i - \bar{X})^2}, \tag{3.1}$$

$$Z = \frac{I - E(I)}{\sqrt{V(I)}}, \tag{3.2}$$

where, W represents the spatial weights matrix and n indicates the total number of features. X_i, and X_j represent the values of the variables that were analyzed at locations i and j, respectively. The average variable value across the locations is indicated by $\bar{X}$. The Z statistic is the standardization of Moran's I.

Spatial lag is an important measure that captures the behavior of a variable in the immediate surroundings of each location. It calculates the weighted sum of the variable of interest. If our weight matrix is binary, spatial lag will give the sum of the values of neighbors, because non-neighbors have zero weight. After calculating spatial lag we can generate the Moran's I plot which visualizes the nature and strength of the spatial autocorrelation. The Moran's I plot of Figure 3.2 displays a positive relationship between both variables, which indicates a positive spatial autocorrelation.

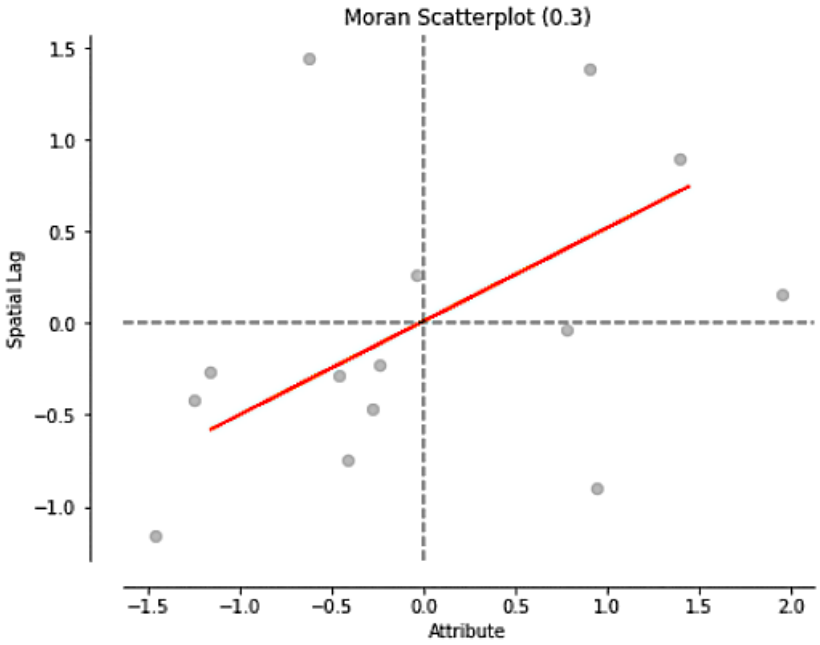


Figure 3.2: Moran's I plot: the scatter plot which displays the variable of interest against the spatial lag

In this study, the spatial distribution pattern of temperature for each sensor was investigated by the global Moran's I. The global Moran's I index value must show the clustering distribution pattern to find high- and low-concentration clusters for further analysis. However, the Global Moran's I check whether there are any spatial clusters. It does not detect the clusters (hot spots/cold spots). Therefore, local measures of spatial autocorrelation [4] which will be used to perform our hotspot analysis (Local Moran's I statistic), are necessary to identify the type of spatial correlation and test the significance of local spatial patterns.

Local Indicators of Spatial Association (LISA)

Local Indicators of Spatial Association is a spatial statistical method that provides insights into spatial patterns at the local level. LISA focuses on identifying local hotspots, and coldspots in a spatial dataset. It evaluates whether the values of a variable at a specific location are similar to or different from the values in its neighboring locations. This method helps pinpoint areas with unique spatial patterns, revealing whether specific locations exhibit high or low values surrounded by similar values.

Local Moran's I is a specific statistic within the LISA framework. Local Moran's I statistic measures local spatial autocorrelation for each feature in a dataset. The formula for Local Moran's I for the i^{th} area/sensor is as follows:

$$I_i = \frac{(X_i - \bar{X})}{S^2} \sum_{j=1}^{n} W_{i,j}(X_j - \bar{X}), \tag{3.3}$$

where $X_i, X_j, W_{i,j}, n$ and $\bar{X}$ are similar to the representations of Global Moran's I formula in equation (3.1). S^2 represents the Variance of the variable values. Here j indicates the neighbors of i^{th} area/sensor. If I_i is positive and significant, the area is part of either the HH cluster (High value surrounded by High) or the LL cluster (Low value surrounded by Low). If I_i is negative and significant, the area is surrounded by dissimilar values (High value surrounded by Low or Low value surrounded by High).

3.2.3 Identifying Neighbors of Sensors

Selecting the correct neighbors for each sensor is an important part of hotspot analysis. We used the k-nearest neighbors algorithm [14] for this. The main reason for choosing the k-nearest neighbors (k-NN) method to define neighborhoods is that it is shown to be highly effective in spatial analysis across numerous studies [43, 1]. One advantage of k-NN is that it focuses on relative proximity rather than absolute physical distance. By selecting the k closest observations, k-NN ensures that each neighborhood is formed based on the most relevant nearby data points, regardless of how far apart they are. This approach is beneficial when local relationships between points are not based on distance alone but on relative positions within their surroundings. That's what we need to focus on in hotspot analysis. In contrast, distance-based methods limit neighborhoods to points within a specific radius, which can sometimes overlook meaningful relationships between points that may be farther apart but still closely linked. Additionally, distance-based methods can result in neighborhoods with uneven sizes, which could introduce bias into the analysis, as areas with fewer neighbors could have less reliable results than those with more neighbors. With k-NN, each point has the same number of neighbors, resulting in a more consistent and reliable neighborhood structure. Overall, k-NN offers a flexible and effective way to define neighborhoods, and compared to distance-based methods, it provides a clearer understanding of spatial patterns that are not bound by arbitrary distance thresholds.

However, in order to identify the neighbors of each sensor using the k-nearest neighbors method, we have to specify the number of neighbors (k) needed for each sensor. Identifying the most appropriate k number is important as this value affects

the final hot and cold spot identification. Here we have introduced a novel method for identifying the optimal k value.

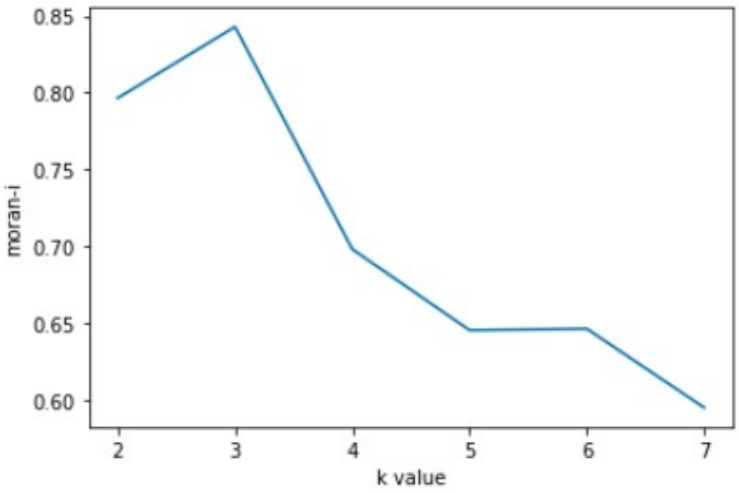


Figure 3.3: Number of neighbors (k) versus Moran's I value

In the k-means clustering algorithm, we use an elbow plot [54] to select the optimal number of clusters. From there we calculated the sum of the square distance between points in a cluster and the cluster centroid, or silhouette score. Similarly, here we used the global Moran's I statistic for each k value. Then, we selected the k value with the highest global Moran's I score. Higher Moran's I values indicate higher spatial autocorrelation and it notifies that the clusters have more solid boundaries. For example, the highest peak of the line plot in Figure 3.3 appears when k=3, which means the optimal k number should be equal to 3. We used the results from the same building dataset discussed in Section 3.2.1 to generate the Figure 3.3.

3.2.4 Feature Importance using Random Forest

Once we identify the hot/cold spots, it would be better if we could understand the reasons or features that affect creating these hot/cold spots. For example, rooms facing south may be warmer than others, while areas with windows can let in more outdoor air, thereby increasing temperature in the summer and decreasing it in the winter. Therefore, we gathered information on the features of those locations and examined how these features influence the formation of hot and cold spots. Since random forest [10] provides a robust and efficient approach for identifying the importance of features, we used that method in this study.

Random forest is a powerful and widely used machine learning algorithm for classification and regression tasks and it can be used to estimate the relative importance of input features in predicting the target variable. The importance of the feature can be measured by how much the given feature decreases the impurity. The more a feature decreases the impurity, the more important the feature is. In random forest, the impurity decrease from each feature can be averaged across trees to determine the final importance of the variable. In this study, we have considered 'Entropy' as the measure of impurity. The entropy is defined as,

$$H(S) = -\sum_{k=1}^{K} f_k log(f_k). \tag{3.4}$$

Here S is the subset of data at a particular node in the tree and f_k is the proportion of observations in S that are of class C_k, $for\ k = 1, ..., K$.

3.3 Results

In this section, we discussed the results of hotspot analysis on the commercial building data and school building data. For commercial building analysis, we considered two special scenarios; one with an outlier temperature value and the other one with randomly distributed temperature values over the floormap. We used these scenarios to assess the drawbacks and assumptions of hotspot analysis. Subsequently, we applied hotspot analysis at each time point and used the results to identify the features of the location that could influence the creation of hot/cold spots. We employed Random Forest and presented the results in this section.

3.3.1 Experimental Analysis

First, we evaluated the performance of hotspot analysis inside a commercial building under three different scenarios. In this experiment, we employed three distinct temperature datasets, with the first and third datasets being synthetic. The first one contains one sensor with a very high temperature (30^0C) compared to the other sensors (around 20^0C). This represents the cases where we have outliers. As the second data set, we used the original dataset of the building. It has a group of high-temperature sensors and a group of low-temperature sensors. The last data set contains sensors that have spatially randomly distributed temperatures. Figures 3.4 - 3.6 with floor maps illustrate the temperature heatmaps of those sensors with their placements.

First, we applied hotspot analysis to the sensors with an outlier sensor (sensor 3). To identify neighbors we applied the k-nearest neighbor method using the optimal k

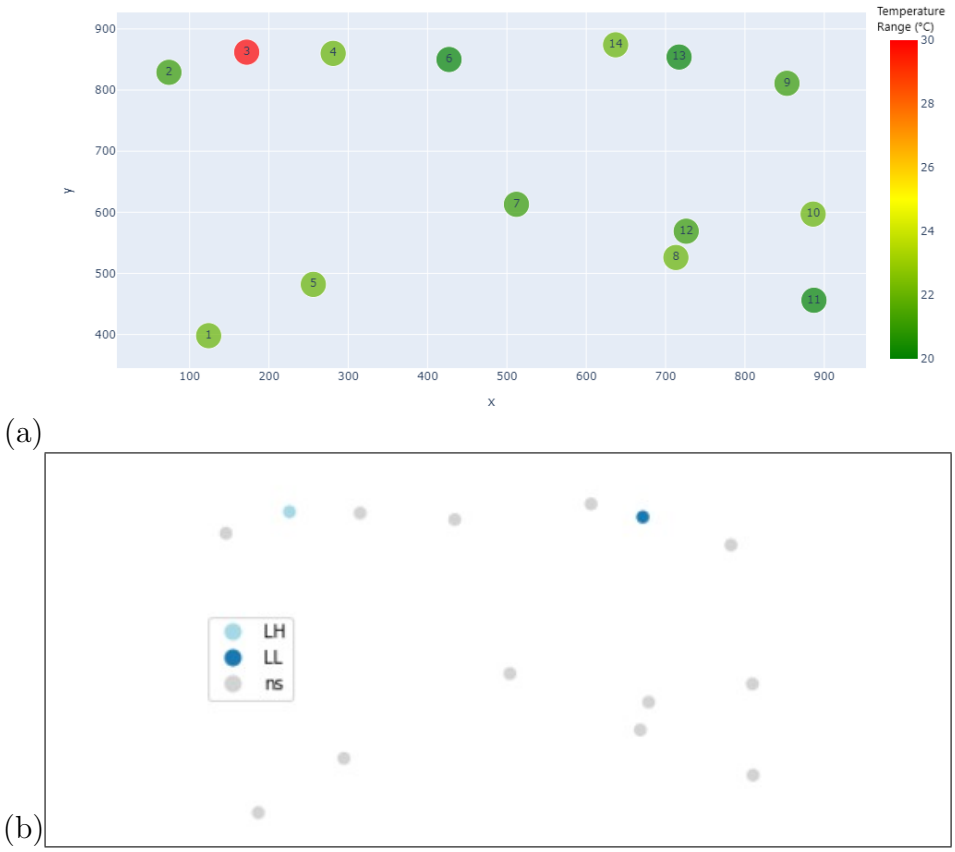


Figure 3.4: (a): Temperature heatmap of dataset 1 with an outlier. The green-to-red color range indicates the cold-to-hot temperature range. (b): Result from hotspot analysis

value, which was obtained by the novel Moran's I based method discussed in Section 2.3. But based on the result which is shown in plot (b) of Figure 3.4, sensor 3 is identified as a cold spot surrounded by hot spots. It should be a hotspot surrounded by cold spots. Hence hotspot analysis struggles to find hotspots when there are outliers.

Then we applied hotspot analysis to the original dataset. According to plot (a) of Figure 3.5, we could observe that it has sensors with groups of high (sensors 2, 3,

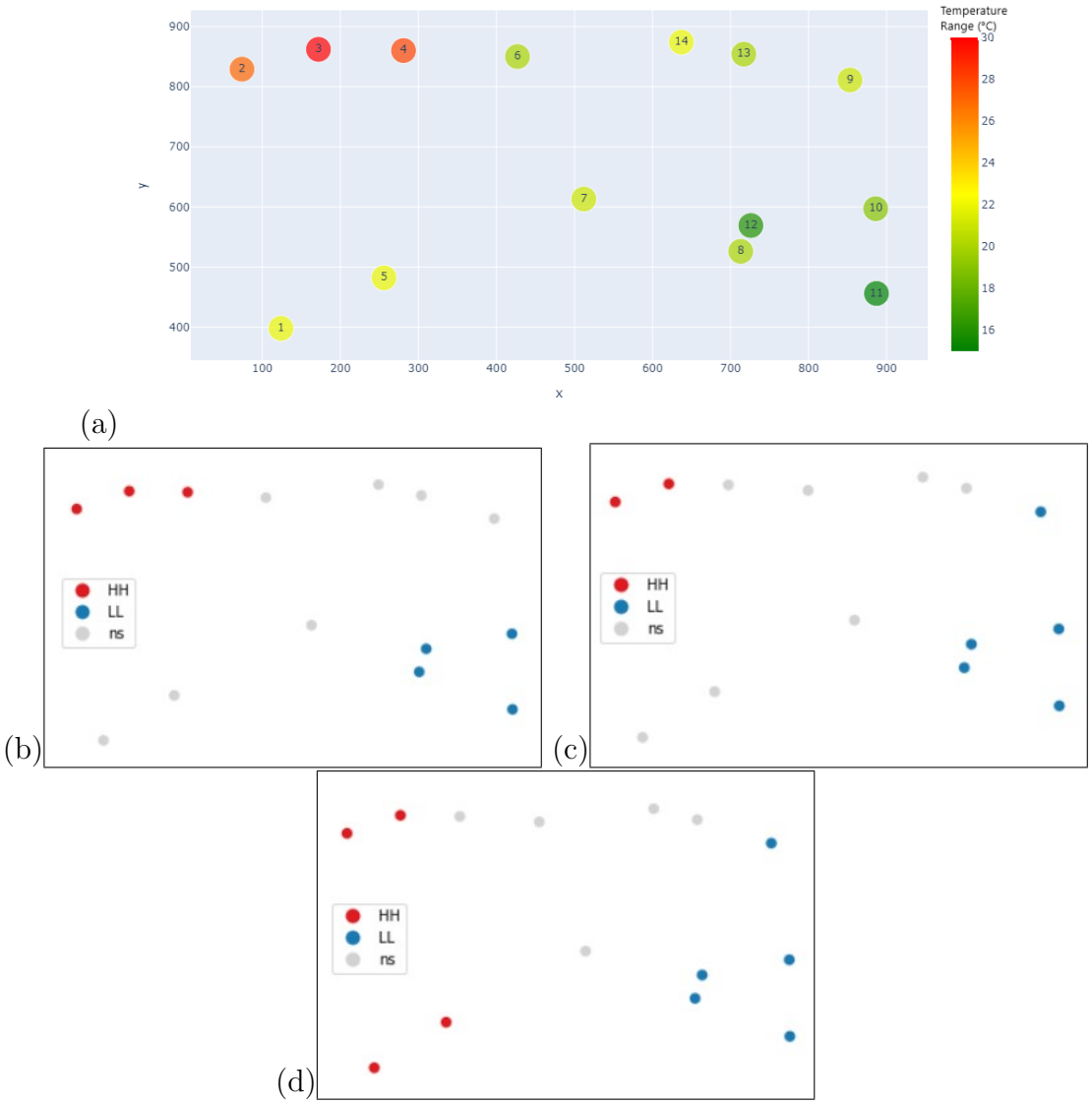


Figure 3.5: (a): Temperature heatmap of the original dataset of the commercial building. Hotspot analysis results when we consider (b): $n = 3$ (c): $n = 4$ and (d): $n = 5$.

and 4) and low values (sensors 8, 10, 11, and 12). Plot (b) to (d) shows the results of hotspot analysis for $n = 3$ to $n = 5$ respectively. From those plots, it is clear that the hotspot analysis method could more accurately identify hot and cold spots when

$n = 3$. The Moran's I-based method that we introduced also identified the optimal n as 3.

Afterward, we conducted hotspot analysis on the dataset where temperatures were randomly distributed among sensors on the floor map of Figure 3.6. The first step of hotspot analysis is checking the global Moran's I test statistic. When considering the randomly distributed sensors we observed a negative value, which indicated a random distribution. Hence, there were no significant hotspots within those sensors.

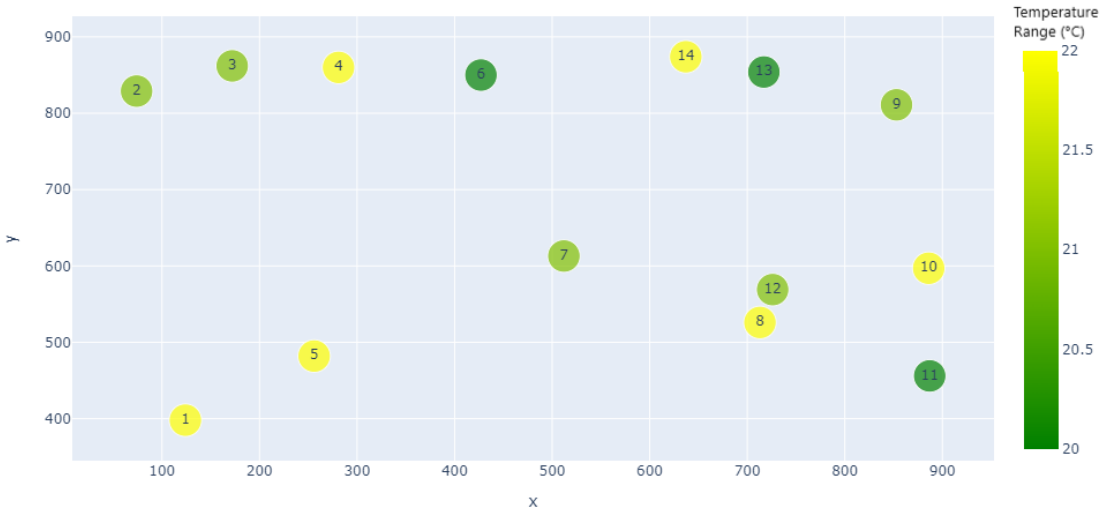


Figure 3.6: Temperature heatmap of the randomly distributed sensors

Finally, we applied the hotspot analysis to a school building. The temperature heatmap for that building is shown in Figure 3.7 and we can see that there is no group of high or low-temperature sensors in that floor map. The temperature values are also within a comfortable range. Once we checked Moran's I statistic for this floor map, we received negative values for all k values. It also confirmed that the

temperature is randomly distributed through this floor and we could not see any significantly high or low areas.

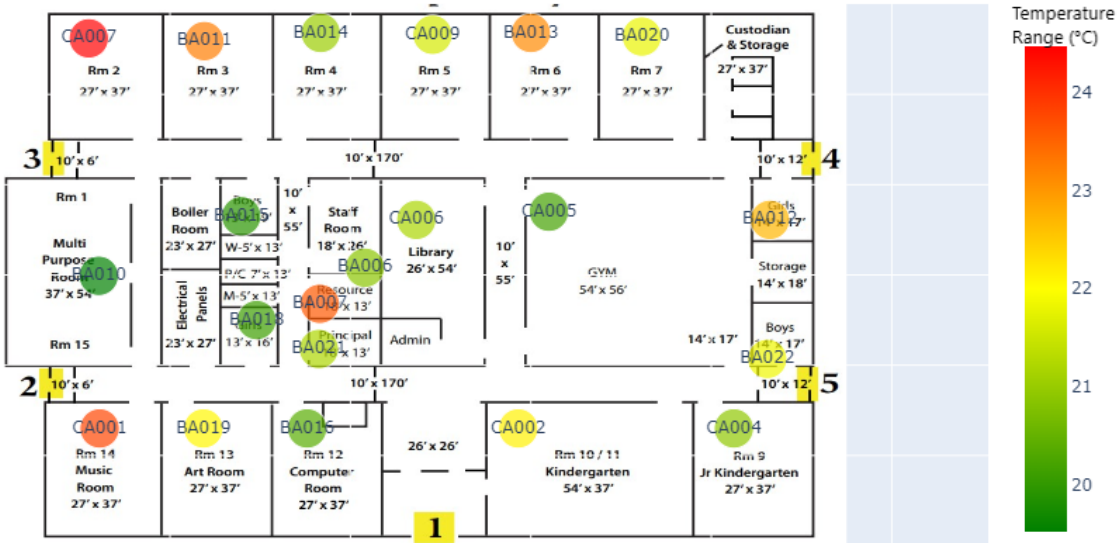


Figure 3.7: Temperature heatmap of the school building. There are no clearly identifiable hot or cold areas.

3.3.2 Exploratory Analysis using The Temporal Hotspot Results

In this section, we have done some exploratory analyses in order to identify the relationship between hot/cold spots and sensor placement and time attributes. To accomplish this we analyzed commercial building data and used the following placement attributes: the location of the sensor within its room type (office, meeting room, kitchen, reception), the position of the sensor (above desk height, desk height, on the floor, on a window sill), whether the space was regularly occupied or not, has

a window or not, and the cardinal direction of the window.

First, we calculated Moran's I statistics on each time point of the data frame. Then we filtered the time points where we can observe positive Moran's I value. On each day we divided time into three time windows; morning (0:00 to 8:00), daytime (8:00 to 16:00), and night (16:00 to 24:00). Then calculated the average Moran's I value on each time window. As Moran's I statistic is high, it suggests that sensors with similar temperature values tend to be located near each other in space. So, based on the above bar plot in Figure 3.8, during the daytime, we will able to see highly clustered areas. September 12th and 13th show lower Moran's I values compared to the other days, but those two days are weekends. On September 14th, we observed a significantly lower average Moran's I value compared to nighttime. This could mainly be because it was a Monday, following the weekend when no machines were operating and no people were present. As a result, little heat transfer occurred on Monday morning, and the outside temperature was quite low, ranging from 10°C to 15°C. However, by the end of the day, the spaces that were occupied had gained more heat, and the outside temperature ranged between 20°C and 25°C during the evening and midnight. These factors may explain the outcomes observed in the study.

Then we applied hotspot analysis on each time point and identified the time points that each sensor is categorized as a significant hot or cold spot. Through this analysis, we could identify whether a set of sensors is always categorized as hot or cold. The following Figure 3.9 shows the proportion of time that each sensor has

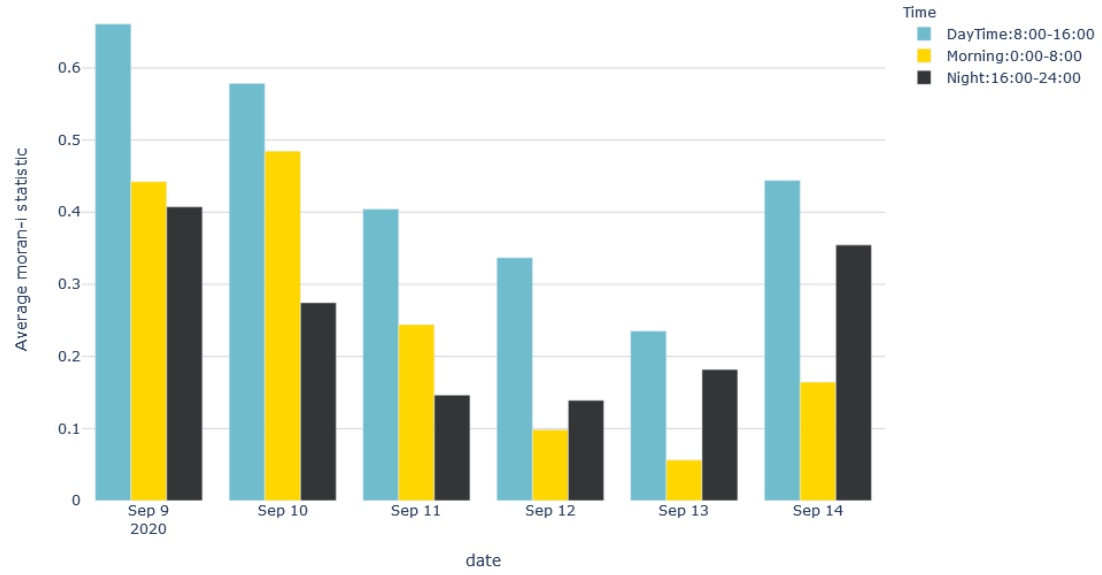


Figure 3.8: Change of average Moran's I statistic (spatial autocorrelation) with the time. Significantly higher average Moran's I value can be seen during the day time

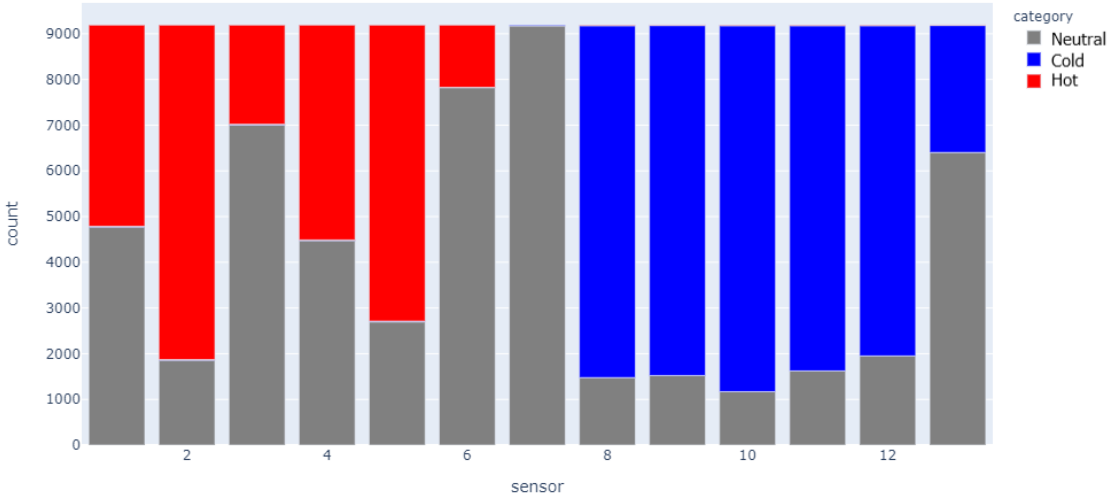


Figure 3.9: Proportion of time that each sensor has been categorized as hot, cold, or neutral

been categorized as hot, cold, or neutral.

When we look at the following bar plot in Figure 3.10, it is clear that vacant places are identified as cold spots 99% of the time, while non-vacant spaces are identified as hot spots 90% of the time. This could be because we collected data during September when the outdoor temperature is typically colder than indoor controlled temperatures which would have a greater cooling effect on vacant spaces, which do not require their IEQ to be conditioned to occupant comfort-level standards. Since building owners try to maintain comfortable temperature levels in areas that are regularly occupied, those can be categorized under hot spots.

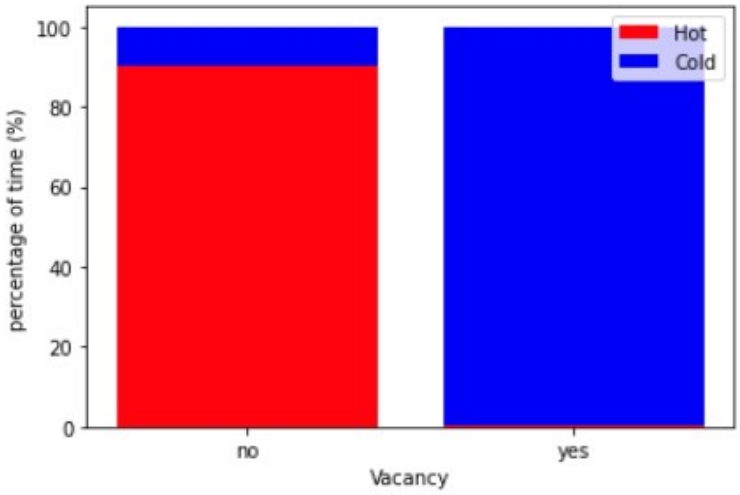
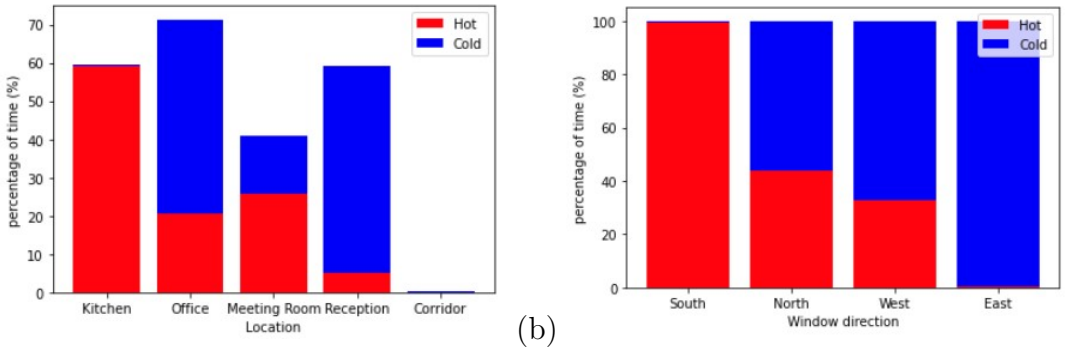


Figure 3.10: Proportion of time that each sensor has been categorized as hot, cold, based on vacancy of the place

When considering the locations, we got the result which is shown in Figure 3.11. Here, the kitchen and office locations are identified as hot spaces, while the reception area has a cooler space. Also within the test, corridors would always show neutral temperatures. When discussing the result based on the window direction, the places



(a) (b)

Figure 3.11: Proportion of time that each sensor has been categorized as hot, cold, (a) based on location, (b): based on window direction

with south-facing windows show hot spaces most of the time. Due to the solar loading effect, the presence of these hot spots can be considered an acceptable result. So based on all these results, we can conclude that hotspot analysis identifies the hot and cold spaces up to an acceptable level.

In order to identify the features that affect the hot and cold spot creation we used random forest. We used all the placement attributes we explained at the beginning of this section to select the important feature. Out of all the features, the 'Vacancy' variable is identified as the most important feature. The sensors that are placed on the floor and locations that have East or North faced windows also have a higher importance. In the following Figure 3.12, the importance value shows the reduction in impurity. The higher the reduction, the more important the feature.

However, the time of day when a building is occupied or unoccupied does not affect the identification of hot/cold spots. Hence even though, being a vacant space affects hot/cold spots, we have no evidence to say that, temperature changes happen due to the heat transmitted by occupants. A possible explanation for this is that facility managers are able to control different spaces based on their intended

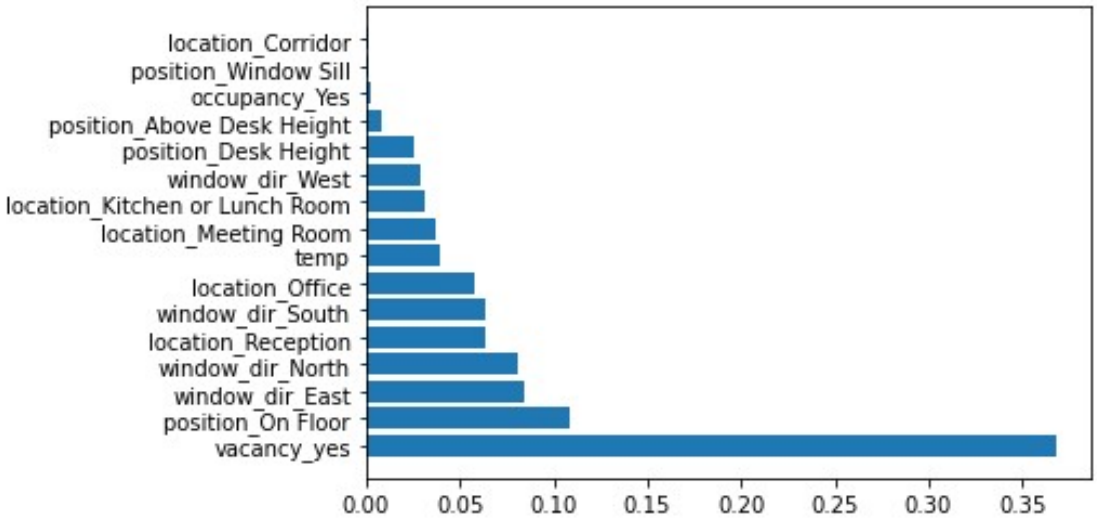


Figure 3.12: Importance of each placement feature on identifying hot and cold spots.

occupancy use, and therefore vacant spaces are intentionally conditioned to be more correlated with external temperatures to increase energy consumption.

3.4 Discussion

This study provides an important contribution to exploring the hot and cold spots of commercial buildings using hotspot analysis. While hotspot analysis has been widely utilized in various research fields, its application in building science research remains largely unexplored. Existing literature on spatial analyses of buildings has mostly focused on cluster analysis rather than hotspot analyses. However, hotspot analysis is equally important as it helps identify spatial patterns and locations where temperatures significantly differ from surrounding areas. In contrast, cluster analysis identifies regions with similar temperature patterns, helping classify areas into temperature zones or climatic regions. The originality of this study lies precisely in its application of building data collected from wireless sensors in hotspot analysis

using Moran's I statistics to identify hotspots. Unlike most existing studies that rely on GIS software and geographical information for hotspot analysis, our study employs the calculation of Moran's I statistics using the (x,y) coordinates of the floor map image.

Through the experimental study, we observed that in order to improve the accuracy of hotspot analysis, we have to consider two main factors. Firstly, it is essential to have sensors spatially clustered based on the feature of concern. When high or low-temperature sensors are dispersed across the floor, or outliers are present, hotspot analysis fails to detect these sensors. It is a drawback of hotspot analysis. Secondly, identifying the correct number of nearest neighbors for each sensor is important. To address this, we introduced a novel method using Moran's I statistics, and our experimental analysis demonstrated its effectiveness.

In the results section of this study, we have demonstrated the relationships between identified hotspots and the features of those locations. It is observed that south-facing locations tend to be hotspots, while east-facing locations are predominantly identified as cold spots. The sensors located in the middle of the building consistently indicate neutral temperatures. Solar loading could be the reason for these hotspots. Furthermore, vacant areas within the building tend to be cold spots, whereas non-vacant spaces display both cold and hot areas. This observation is reasonable. In non-vacant spaces, individuals contribute body heat, and also considering the experiment was conducted in the fall in Winnipeg, a city known for its cold climate, occupants likely used heaters. To identify the importance of these placement features we applied random forest, which revealed that having a non-vacant or vacant space

is the most important feature.

By identifying these specific areas where hot and cold spots are occurring, we can diagnose where specific issues are impacting that building's IEQ and seek to identify a potential resolution to the identified anomaly. This presents a significant improvement to current building science applications which commonly deploy either building-level data analysis which cannot identify site-specific areas of concern, or occupant surveys which rely on subjective and individual feedback. This analysis also offers facility managers a way to verify if their HVAC control settings are properly distinguishing between areas that should be conditioned to occupied and non-occupied levels.

Based on all these analyses, it is clear that hotspot analysis can be applied to temperature data within buildings with some limitations. The effect of applying this analysis can mean more accessible and site-specific data which will help to realize a more uniformly controlled temperature across a floorplan, helping to both decrease energy consumption levels and improve occupant comfort levels. Buildings do have many enclosed spaces, such as rooms with walls, doors, and windows. Even though the sensors are close to each other, there could be walls between them. Therefore, as a future development, we aim to enhance hotspot analysis by considering whether the locations are enclosed or open spaces. The heat transmission is different in closed and open spaces. Therefore, considering whether a location is an open or closed space can be important for accurately assessing temperature patterns and conducting hotspot analysis.

Bibliography

[1] M. S. Ahmed, M. N'diaye, M. K. Attouch, and S. Dabo-Niange. k-nearest neighbors prediction and classification for spatial data. *Journal of Spatial Econometrics*, 4, 2023.

[2] J. Aldstadt. *Spatial Clustering*, pages 279–300. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[3] J. G. Allen and J. D. Macomber. *Healthy buildings: how indoor spaces drive performance and productivity*. Harvard University Press, 2020.

[4] L. Anselin. Local indicators of spatial association—lisa. *Geographical Analysis*, 27(2):93–115, 1995.

[5] A. Ayanlade, O. M. Esho, K. O. Popoola, O. D. Jeje, and B. A. Orola. Thermal condition and heat exposure within buildings: Case study of a tropical city. *Case Studies in Thermal Engineering*, page 100477, 2019.

[6] A. Bajaj. Anomaly detection in time series. https://neptune.ai/blog/anomaly-detection-in-time-series#:~:text=Anomaly, 2022. Accessed 20 April 2022.

[7] P. Biam. Rtem hackaton api and data science tutorials. https://www.kaggle.com/datasets/ponybiam/onboard-api-intro?select=all_points_metadata.csv/, 2022. Accessed 03 May 2022.

[8] Y. Bing, P. D. Van, and S. Riahy. General modeling for model-based fdd on building hvac system. *Simulation Practice and Theory*, 9:387–397, 2002.

[9] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[10] L. Breiman. Random forests. *Machine Learning*, 45:5–32, 2001.

[11] P. J. Brockwell and R. A. Davis. Time series: Theory and methods (2nd ed.). *New York: Springer*, 2009.

[12] C. M. Catherine, S. Eduardo, Mc. Kaitlyn, F. Trevon, S. Emma, and N. Karin. Maternal hiv and syphilis are not syndemic in brazil: Hot spot analysis of the two epidemics. *PLOS ONE*, pages 1–10, 2021.

[13] M. C. Chuah and F. Fu. Ecg anomaly detection via time series analysis. In *Frontiers of High Performance Computing and Networking ISPA 2007 Workshops*, pages 123–135, 2007.

[14] T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, pages 21–27, 1967.

[15] N. Debanjana and P. Harry. Automated real-time anomaly detection of temperature sensors through machine-learning. In *International Journal of Sensor Networks*, page 137–152. Inderscience Publishers, Geneva 15, CHE, 2020.

[16] E. Delzendeh, S. Wu, A. Lee, and Y. Zhou. The impact of occupants' behaviours on building energy analysis: A research review. *Renewable and Sustainable Energy Reviews*, pages 1061–1071, 2017.

[17] M. D. Diab, A. Basil, B. Hamad, L. Sangarapillai, G. K. Konstantinos, and G. Ibrahim. Anomaly detection using dynamic time warping. In *2019 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)*, pages 193–198, 2019.

[18] T. Fu. A review on time series data mining. *Eng. Appl. Artif. Intell.*, page 164–181, 2011.

[19] W. A. Fuller. *Introduction to statistical time series*. John Wiley and Sons., 1976.

[20] J. C. Gower. Properties of euclidean and non-euclidean distance matrices. *Linear Algebra and its Applications*, pages 81–97, 1985.

[21] M. Gross. 9 causes of hot or cold spots in your home and how to fix them. https://kobiecomplete.com/blog/9-causes-of-hot-or-cold-spots-in-your-home-and-how-to-fix-them.html, 2019. Accessed 14 February 2023.

[22] G. Guerri, A. Crisci, A. Messeri, L. Congedo, M. Munafò, and M. Morabito. Thermal summer diurnal hot-spot analysis: The role of local urban features layers. *Remote Sensing*, 13:538, 2021.

[23] R. Haroon and S. Pushpendra. Monitor: An abnormality detection approach in buildings energy consumption. In *2018 IEEE 4th International Conference on Collaboration and Internet Computing (CIC)*, pages 16–25, 2018.

[24] N. L. Harris, E. Goldman, C. J. Gabris, S. Minnemeyer, S. Ansari, M. Lippmann, L. Bennett, M. Raad, M. Hansen, and P. Potapov. Using spatial statistics to identify emerging hot spots of forest loss. *Environmental Research Letters*, 12:024012, 2017.

[25] S. M. S. Islam, K. M. A. Islam, and M. R. A. Mullick. Drought hot spot analysis using local indicators of spatial autocorrelation: An experience from bangladesh. *Environmental Challenges*, 6:100410, 2022.

[26] M. A. Jalali, D. Ierodiaconou, H. Gorfine, J.Monk, and A. Rattray. Exploring spatiotemporal trends in commercial fishing effort of an abalone fishing zone: A gis-based hotspot model. *PLOS ONE*, 10:e0122995, 2015.

[27] M. Kumari, K. Sarma, and R. Sharma. Using moran's i and gis to study the spatial pattern of land surface temperature in relation to land use/cover around a thermal power plant in singrauli district, madhya pradesh, india. *Remote Sensing Applications: Society and Environment*, 15:100239, 2019.

[28] Y. Y. Lee, Y. H. Lee, S. Mohammad, P. N. Shek, and C. K. Ma. Thermal characteristics of a residential house in a new township in johor bahru. *IOP Conference Series: Materials Science and Engineering*, 271:012027, 2017.

[29] M. Lehnert, J. Geletič, J. Kopp, M. Brabec, M. Jurek, and J. Pánek. Comparison between mental mapping and land surface temperature in two czech cities: A

new perspective on indication of locations prone to heat stress. *Building and Environment*, 203:108090, 2021.

[30] H. Lin and T. Hong. On variations of space-heating energy use in office buildings. *Applied Energy*, 111:515–528, 2013.

[31] F. Liu, H. Lee, Y .and Jiang, J. Snowdon, and M. Bobker. Statistical modeling for anomaly detection, forecasting and root cause analysis of energy consumption for a portfolio of buildings. *Proceedings of Building Simulation 2011: 12th Conference of International Building Performance Simulation Association*, pages 2530–2537, 2011.

[32] F. T. Liu, K. M. Ting, and Z. Zhou. Isolation-based anomaly detection. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, page 3, 2012.

[33] H. F. Lopes. Ar(1) plus noise model. http://hedibert.org/wp-content/uploads/2017/06/AR1plusnoise-blockmove-singlemove-fixedparameters.html, 2017. Accessed 15 April 2022.

[34] B. Mark, J. A. Fava, C. A. Harnanan, P. Strothmann, S. Khan, and S. Miller. *Hotspots Analysis: Providing the Focus for Action*, pages 149–167. Springer Netherlands, Dordrecht, 2015.

[35] B. M. Martinez, L. Thompson, M. Cockburn, and J. Tsui. Hot spot analysis of cervical cancer among racial and ethnic minorities in los angeles county. *Cancer Epidemiology, Biomarkers & Prevention*, 31:PO–004–PO–004, 2022.

[36] T. Masron, A. Marzuki, N. F. Yaakub, M. N. Nordin, and N. Jubit. Spatial

analysis of crime hot-spot in the northeast penang island district and kuching district, malaysia. *Planning Malaysia*, 19, 2021.

[37] T. C. Mills. *Time series Techniques for Economics*. Cambridge University Press, 1990.

[38] P. A. P. Moran. Notes on continuous stochastic phenomena. *Biometrika*, 37:17–23, 1950.

[39] V. Morckel and N. Durst. Using emerging hot spot analysis to explore spatiotemporal patterns of housing vacancy in ohio metropolitan statistical areas. *Urban Affairs Review*, 59:309–328, 2023.

[40] G. Moschini, R. Houssou, J. Bovay, and S. Robert-Nicoud. Anomaly and fraud detection in credit card transactions using the arima model. *Engineering Proceedings*, 5, 2021.

[41] J. K. Muhammad. Grid search optimization algorithm in python. https://stackabuse.com/grid-search-optimization-algorithm-in-python/, 2020.

[42] F. Nielsen. *Introduction to HPC with MPI forData Science*. Springer, 2016.

[43] M. Otair. Approximate k-nearest neighbour based spatial clustering using k-d tree. *International Journal of Database Management Systems*, 5, 2013.

[44] A. Pandarasamy, D. K Harshad, G. Tanuja, C. M. Zainul, S. Amarjeet, and S. Pushpendra. Multi-user energy consumption monitoring and anomaly detection with partial context information. In *Proceedings of the 2nd ACM*

International Conference on Embedded Systems for Energy-Efficient Built Environments, page 35–44. Association for Computing Machinery, 2015.

[45] T. G. Puranik and D. Mavris. Anomaly detection in general aviation operations using energy metrics and flight data records. *Journal of Aerospace Information Systems, American Institute of Aeronautics and Astronautics*, 15:22–35, 2018.

[46] A. Ratanamahatana, J. Lin, D. Gunopulos, E. Keogh, M. Vlachos, and G. Das. Mining time series data. In *Data Mining and Knowledge Discovery Handbook (2nd ed.)*, page 1049–1077. Springer US, Boston, MA, 2010.

[47] C. Robert, C. William, and T. Irma. Stl: A seasonal-trend decomposition procedure based on loess. *Journal of Official Statistics*, page 3, 1990.

[48] I. Sakellaris, D. Saraga, C. Mandin, C. Roda, S. Fossati, Y. de Kluizenaar, P. Carrer, C. Dimitroulopoulou, V. G. Mihucz, T. Szigeti, O. Hänninen, E. De Oliveira Fernandes, J. G. Bartzis, and P. M. Bluyssen. Perceived indoor environment and occupants' comfort in european "modern" office buildings: The officair study. *International Journal of Environmental Research and Public Health*, 13:444, 2016.

[49] C. A. G. Santos, N. R. M. Brasil, R. M. da Silva, and S. G. F. Costa. Cluster analysis applied to spatiotemporal variability of monthly precipitation over paraíba state using tropical rainfall measuring mission (trmm) data. *Remote Sensing*, 11(6), 2019.

[50] D. Savage, X. Zhang, X. Yu, P. Chou, and Q. Wang. Anomaly detection in online social networks. *Social Networks*, 39:62–70, 2014.

[51] S. Schmidl, P. Wenig, and T. Papenbrock. Anomaly detection in time series: A comprehensive evaluation. *Proc. VLDB Endow.*, 15(9):1779–1797, 2022.

[52] X. Shan, N. Luo, K. Sun, T. Hong, Y. Lee, and W. Lu. Coupling cfd and building energy modelling to optimize the operation of a large open office space for occupant comfort. *Sustainable Cities and Society*, 60:102257, 2020.

[53] M. Shariati, T. Mesgari, M. Kasraee, and M. Jahangiri-rad. Spatiotemporal analysis and hotspots detection of covid-19 using geographic information system. *Journal of Environmental Health Science and Engineering*, page 1499–1507, 2020.

[54] R. L. Thorndike. Who belongs in the family? *Psychometrika*, 18(4):267–276, 1953.

[55] A. Torku, A. P.C. Chan, E. H. K. Yung, and J. Seo. Detecting stressful older adults-environment interactions to improve neighbourhood mobility: A multimodal physiological sensing, machine learning, and risk hotspot analysis-based approach. *Building and Environment*, 224:109533, 2022.

[56] P. Tormene, T. Giorgino, S. Quaglini, and M. Stefanelli. Matching incomplete time series with dynamic time warping:an algorithm and an application to post-stroke rehabilitation. *Artificial Intelligence in Medicine*, 45:11–34, 2008.

[57] T. M. Tubiak. *The Certified Six Sigma Black Belt Handbook, Second Edition*. ASQ Quality Press, 2009.

[58] K. Ueno. How to look at a house like a building scientist (part 2: Heat). https://www.greenbuildingadvisor.com/article/how-to-look-

at-a-house-like-a-building-scientist-part-2-heat, 2019. Accessed 20 March 2023.

[59] Y. Unal, T. Kindap, and M. Karaca. Redefining climate zones for turkey using cluster analysis. *International Journal of Climatology*, 23:1045 – 1055, 2003.

[60] Z. Wang, T. Parkinson, P. Li, B. Lin, and T. Hong. The squeaky wheel: Machine learning for anomaly detection in subjective thermal comfort votes. *Building and Environment*, 151:219–227, 2019.

[61] L. Wei, J. Hongyi, C. Dandan, C. Lifei, and J. Qingshan. A real-time temperature anomaly detection method for iot data. In *5th International Conference on Internet of Things, Big Data and Security*, pages 112–118, 2020.

[62] D. J. Wheeler and D. S. Chambers. *Understanding Statistical Process Control*. SPC Press, 1992.

[63] A. Wickramasinghe, S. Muthukumarana, D. Loewen, and M. Schaubroeck. Temperature clusters in commercial buildings using k-means and time series clustering. *Energy Informatics*, 5, 2022.

[64] J. Woo, P. Rajagopalan, M. Francis, and P. Garnawat. An indoor environmental quality assessment of office spaces at an urban australian university. *Building Research & Information*, 49:842–858, 2021.

[65] H. Yassine, G. Khalida, A. Abdullah, B. Faycal, and A. Abbes. Artificial intelligence based anomaly detection of energy consumption in buildings: A review, current trends and new perspectives. *Applied Energy*, 287:116601, 2021.

[66] J. YooSeok, K. TaeWook, and C. Chanjun. Anomaly analysis on indoor office spaces for facility management using deep learning methods. *Journal of Building Engineering*, page 103139, 2021.

[67] M. You, R. Lai, J. Lin, and Z. Zhu. Quantitative analysis of a spatial distribution and driving factors of the urban heat island effect: A case study of fuzhou central area, china. *International Journal of Environmental Research and Public Health*, 18, 2021.

[68] W. Yu, B. Li, R. Yao, D. Wang, and K. Li. A study of thermal comfort in residential buildings on the tibetan plateau, china. *Building and Environment*, pages 71–86, 2017.

[69] H. Zubaidi, I. Obaid, H. A. Mohammed, S. Das, and N. S. S. Al-Bdairi. Hot spot analysis of the crash locations at the roundabouts through the application of gis. *Journal of Physics: Conference Series*, page 012032, 2021.

Chapter 4

An Anomaly Detection Method for Identifying Locations with Abnormal Behavior of Temperature in School Buildings

Time series data collected using wireless sensors, such as temperature and humidity, can provide insight into a building's heating, ventilation, and air conditioning (HVAC) system. Anomalies of these sensor measurements can be used to identify locations of a building that are poorly designed or maintained. Resolving the anomalies present in these locations can improve the thermal comfort of occupants, as well as improve air quality and energy efficiency levels in that space. In this chapter, we developed a scoring method to identify sensors that show collective anomalies due to environmental issues. This leads to identifying problematic locations within commercial and institutional buildings. The Dynamic Time Warping (DTW) based anomaly detection method was applied to identify collective anomalies. Then, a score for each sensor was obtained by taking the weighted sum of the number of anomalies,

vertical distance to an anomaly point, and dynamic time-warping distance. The weights were optimized using a well-defined simulation study and applying the grid search algorithm. Finally, using a synthetic data set and the results of a case study we could evaluate the performance of our developed scoring method. In conclusion, this newly developed scoring method successfully detects collective anomalies even with data collected over one week, compared to the machine learning models which need more data to train themselves.

4.1 Introduction

Anomaly detection is one of the most popular research areas in time series data mining. A data point that does not follow the pattern of the rest of the data can be considered an anomaly or outlier. Identifying these anomaly points is important for many industries. Some applications include the detection of abnormal behavior of ECG signals in the health industry [6], credit card fraud detection in the banking industry [19], anomalous behavior in aircraft [21], identifying spammers, online fraudsters in social media [23] and many more [7].

Nowadays commercial buildings also provide an opportunity to monitor indoor environment quality with the help of Internet of Things (IoT) sensors. These devices measure the environment of the building and generate temporal data which helps building owners understand the indoor environmental quality (IEQ) and thermal comfort of the tenants. In another research work, we used these sensor data to identify locations with similar thermal environments of a building [28]. Identifying the issues of the indoor thermal environment is important for building owners to

save energy and keep tenants comfortable. To identify these issues, we can inspect IoT sensor data and detect abnormal behaviors. The presence of these abnormal behaviors might occur due to the issues with the location of those sensors.

In 2019, Wang and his team proposed an anomaly detection framework for thermal comfort in buildings [26]. It is a stochastic-based, two-step anomaly detection framework that is based on occupants' votes. An outlier is automatically flagged when a vote significantly differs from other occupants' votes. In the building industry, Fault diagnosis and detection (FDD) is an application of anomaly detection that monitors building HVAC systems to identify faults [4]. Graph-based anomaly detection methods also have been introduced by some researchers [14, 30]. Jung and the team [13] conducted research to detect anomalies in indoor office spaces by predicting values using long short-term memory (LSTM). They have used IoT sensors to collect temperature and humidity data.

The majority of existing research [15, 8, 27, 2] have considered identifying abnormal points of a single time series (intra-time series anomaly detection). But, these single abnormal points can be raised due to equipment failures within the data collection hardware itself, which may lead to a false positive or false negative. Lieu et al. [15] have used energy data, and a data point that falls outside the 95% confidence bounds is considered abnormal consumption. By dividing univariate time series into subsequences, Debanjana and Harry [8] could classify a data point as an anomaly or normal point using one-class support vector machines (OC-SVM) in 2020. In Wei et al.'s paper [27], an unsupervised temperature anomaly detection method is proposed to detect anomalies in real-time temperature time series. It sets dynamic thresholds

based on the Smoothed Z-Score Algorithm.

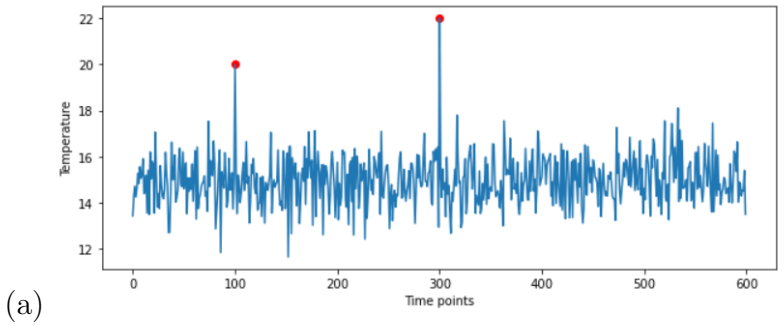
The importance of our study is that it identifies abnormal time series compared to every time series that we use in a study (inter-time series anomalies) and proposes a novel method to identify collective anomalies, which can occur due to environmental issues. Also, this method does not require a large number of data to detect anomalies and our scoring method identifies the sensors with the most abnormal data points. Ultimately the locations of these sensors can be considered as the locations that indicate a potential issue within the HVAC system or building construction. The scoring method development considered the number of anomalies, the vertical distance between the average point and an abnormal point, and the DTW distance between the average time series and the given time series. In the end, we applied the developed scoring method in a synthetic data set and a school building, located in New York, USA, and discussed the results.

4.2 Material and Methods

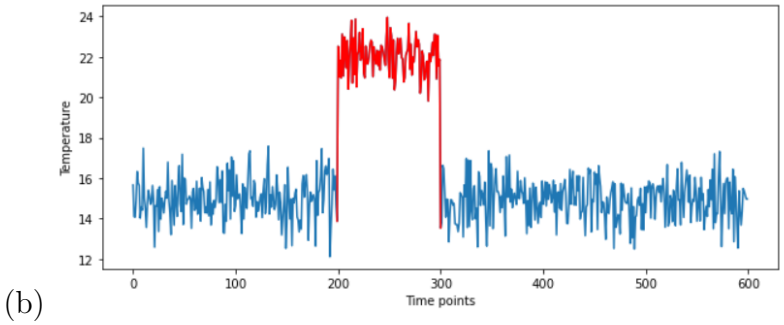
4.2.1 Anomaly Detection

An anomaly is something that deviates from what is considered standard or expected [10]. The anomaly detection problem for time series is usually formulated in a way that can identify outlier data points relative to some usual signal. Anomaly detection within the context of buildings has real-world implications. Anomalies in building IEQ data can be caused by either environmental issues or hardware issues. A collective anomaly could identify inefficient controls or HVAC systems or could point to occupant behaviors negatively impacting the energy use within that

space [29]. Contextual anomalies (as defined by a greater variation from the average within that same space) could identify failing or poorly calibrated hardware, or failure points within a building's mechanical or construction environment. Figure 4.1 shows the difference between these contextual and collective anomalies.



(a)



(b)

Figure 4.1: (a): Contextual anomalies: due to hardware issue, (b): Collective anomalies: due to environmental issue.

There are many approaches to detecting an anomaly in time-series data [24] and a few common methods are presented in Table 4.1. In order to identify environmental problematic areas, we wanted to identify collective anomalies. Hence, we applied these three anomaly detection methods and DTW based anomaly detection method, which is explained in Section 4.2.1, to a sample time series data and selected the best method.

Method	Summary
Seasonal Trend Decomposition [22]	It only uses residue data from the decomposition to identify anomalies. This algorithm calculates the deviation of residue and uses a threshold to identify anomalies.
Isolation Forest method [16]	It randomly selects a feature from the given set of features and then randomly selects a split value between the max and min values of that feature to isolate the outliers.
Forecasting method [1]	The forecasting method is based on the approach that generates a predicted value of the next point, by considering several points from the past.

Table 4.1: Anomaly detection methods used in this study were compared with the DTW-based method.

Anomaly Detection Method Based on Dynamic Time Warping

This method is based on the anomaly detection method, which was introduced by Diab et al [9]. It has a control time series and a data time series. Anomalies are identified based on the distance between points of the optimal path. The optimal path means the optimal match between the control series and the given time series. The basic steps of this algorithm are as follows:

1. Identify the optimal similarity path between two series.
2. Calculate Euclidean distances between those points in the optimal path.
3. Calculate median absolute deviation (MAD) of distances.
4. Consider the points that are 3 times MAD away from the median as outliers.
5. Three or more consecutive outliers are considered as anomalies.

The importance of this algorithm is that it considers three or more consecutive outliers as anomalies. It detects only the collective anomalies, which are caused by environmental issues. However, this algorithm identifies anomalies based on a control sequence, but in practical situations, we will not always be able to find control sequences. Also, this method detects anomalies within a single time series. But in our study, we want to identify abnormal time series compared to all the time series (inter-time series anomaly detection). Then it helps to identify abnormal sensors and locations with issues. Due to these limitations, we made some alterations to the existing method to apply it in our research study.

Instead of the control sequence, we used the median time series, which was created by calculating the median values of all the time series at each time point. Also when calculating the MAD value, we observed all the distance values we got when comparing the median time series with each time series. By calculating MAD in that way, we could consider all the time series and identify the anomalies.

Dynamic Time Warping

Dynamic Time Warping (DTW) [25] measures the distance between two arrays or time series. DTW is a method that calculates an optimal match between two given sequences. This method allows us to find the distance between sequences of different lengths. Let A and B be two sequences with length L_A and L_B respectively. a_i and b_j indicate the i^{th} and j^{th} observations of A and B respectively. Then the pairwise Euclidean distances [12] can be calculated for each observation of A and B. It will yield the $L_A \times L_B$ distance matrix S. The cumulative distance matrix D is

calculated as in equation (1). The matrix D captures the total cost of alignment between the a_1, b_1 and a_{L_A}, b_{L_B}. A lower total cost shows higher similarity between the two sequences.

$$D_{i,j} = min(D_{i-1,j}, D_{i,j-1}, D_{i-1,j-1}) + S_{ij}. \tag{4.1}$$

In the above equation $i = 1, ..., L_A$, $j = 1, ..., L_B$ and $S_{ij} = d(a_i, b_j)$. The final distance between A and B is the bottom corner value of the cumulative distance matrix D, which is $D(L_A, L_B)$

Median Absolute Deviation

The Median Absolute Deviation (MAD) is a robust measure of how a set of data is spread out. The variance and standard deviation are also measures of spread, but they are more affected by extremely high or extremely low values. Also, in practical situations, it is hard to get normally distributed data. Hence, the MAD is one statistic that we can use instead. It is less affected by outliers because outliers have a smaller effect on the median than they do on the mean. MAD is defined as follows,

$$MAD = median(|X_i - median(X)|). \tag{4.2}$$

Here X_i represents i^{th} observation and according to our study X means the list distances between each data point of the optimal path, created using each time series and the control time series.

4.2.2 Abnormal Sensor Detection Method

In this section, we explain the method we developed to identify sensors that can be considered abnormal compared to a group of sensors positioned within a building. The locations of the abnormal sensors can be considered problematic areas of the building. Algorithm 2 summarizes the process of our algorithm.

Algorithm 2: Anomaly Detection Algorithm

Input:
Sensor time series data: T,
Weights: w_1, w_2, w_3
Output:
List of Anomaly scores for each sensor (ψ)
Step 1: Calculate median time series (Control time series);
Step 2:
1: **for** each time series t_i in T **do**
2: **Step 2.1:** Find number of anomalies, A_{t_i}, using DTW-based method
3: **Step 2.2:** Calculate vertical distance, V_{t_i}.
4: **Step 2.3:** Calculate DTW distance, D_{t_i}.
5: **Step 2.4:** Calculate anomaly score:

$$\psi_{t_i} = w_1 \cdot A_{t_i} + w_2 \cdot V_{t_i} + w_3 \cdot D_{t_i}.$$

6: **Step 2.5:** Add ψ_{t_i} score to the ψ
7: **end for**

We developed a scoring method to identify abnormal sensors, and identifying the main parameters that help detect differences between two-time series was important. Considering literature and observations we decided to use the following parameters to compare the difference between the control and the given time series:

1. Number of outliers (based on Section 4.2.1).

2. Median of vertical Euclidean distances between outlier points, and control time series.

3. DTW distance between time series (check the similarity of the patterns).

We calculated the above values for all the sensors, normalized using min-max normalization, and brought all values into one scale, from 0 to 1. Not all these three parameters have the same level of importance when detecting the difference between control and other time series. Hence, we considered generating a weighted score as above in algorithm 1, step 2.4, and ranking the time series based on these scores. However, since there is no historical data, on which are labeled as abnormal or not, optimizing the weights in a way that the abnormal sensors get the highest score was another challenge. Hence to overcome this, we had to develop a method to scale the weights using a simulation study.

4.2.3 Simulation Study

Since we have no information about the actual abnormal sensors, we developed a simulation study to generate a set of 'normal' time series, and an 'abnormal' by considering the control time series. The flow chart in Figure 4.2 summarizes the simulation work.

First, we had to identify the time series model of the median time series in order to generate a set of normal time series. Here we considered the auto-regressive model (AR), moving-average model (MA), Auto Regressive Moving Average (ARMA), and Auto-Regressive Integrated Moving Average (ARIMA) models which are expressed using the equations (4.3),(4.4),(4.5) and (4.6) respectively. In these equations, y_t

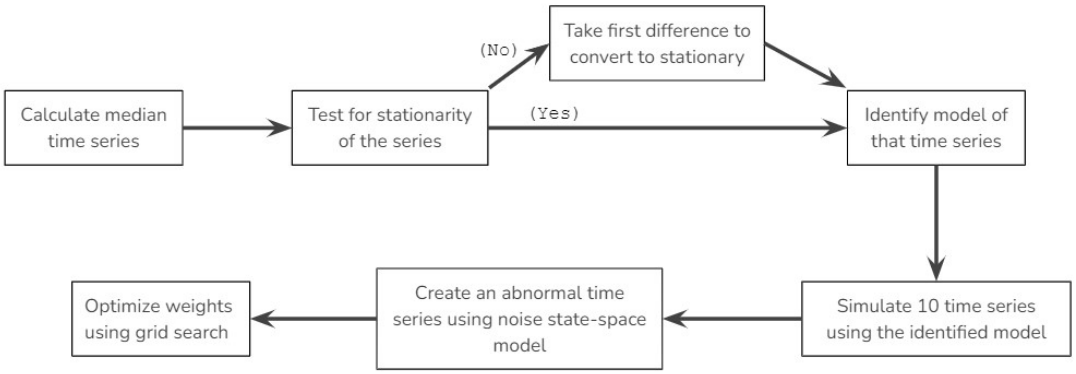


Figure 4.2: Flow chart of the simulation study

means y, the response variable, measured at time t, and β, α values represent the coefficients of the AR model and MA model respectively. ϵ_t expresses the randomness, c is the constant factor, θ_i indicates the numeric coefficient for the value associated with the i^{th} lag, and ε represents the residual. The differenced time series of the ARIMA model is indicated by $y_t{}' = y_t - y_{t-1}$.

$$y_t = \beta_0 + \beta_1 y_{t-1} + \beta_2 y_{t-2} + ... + \beta_p y_{t-p} + \epsilon_t. \tag{4.3}$$

$$y_t = c + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + ... + \theta_q \varepsilon_{t-q}. \tag{4.4}$$

$$y_t = \beta_1 y_{t-1} + \alpha_1 \varepsilon_{t-1} + \beta_2 y_{t-2} + \alpha_2 \varepsilon_{t-2} + ... + \beta_k y_{t-k} + \alpha_k \varepsilon_{t-k}. \tag{4.5}$$

$$y_t{}' = c + \beta_1 y'_{t-1} + \beta_2 y'_{t-2} + \alpha_2 \varepsilon_{t-2} + ... + \beta_p y'_{t-p} + \alpha_1 \varepsilon_{t-1} + ... + \alpha_q \varepsilon_{t-q} + \varepsilon_t. \tag{4.6}$$

The Augmented Dickey-Fuller test [11] is used to check the stationarity of the time series. If the time series is stationary, we proceeded to AR, MA, or ARMA models and selected the best model among them using the lowest AIC

(Akaike Information Criterion) and BIC (Bayesian Information Criterion) values [5]. Otherwise, we had to take the difference to convert to stationary form and consider the ARIMA model with the rest of the models. Partial Autocorrelation (PACF) plots and Autocorrelation (ACF) plots [18] were used to find the order of AR and MA respectively. The differencing order of the ARIMA model is based on the number of times we took the difference to make the series stationary. After identifying the model we simulate 10 time series from the selected model.

The next step was generating an abnormal time series which showed unusual behavior compared to the rest of the time series. For that, we applied the AR(1) plus noise state-space model [17] to generate our abnormal time series and it gave a distinguishable time series as in Figure 4.3. The AR(1) plus state space model can be explained as follows.

$$z_t = x_t + \epsilon_t. \tag{4.7}$$

$$x_t = \phi x_{t-1} + c + w_t. \tag{4.8}$$

Here equation (4.7) is the observation equation, and equation (4.8) is the state equation. x_t denotes the state at time t, the transition matrix is $[\phi]$, the observation matrix is [1], and the transition offset is c. The observation and transition noises, $WN(0, \sigma^2)$ are indicated by ϵ_t and w_t respectively. The expectation of state is $E[X_t] = \frac{c}{1-\phi}$.

Now we have 11 time series: the first 10 time series are normal and the 11^{th} series is abnormal. Using these 11 time series, we generated a median time series and calculated anomaly count, vertical distance, and DTW distance. Then we used the grid search optimization method [20] to optimize the weights in a way that

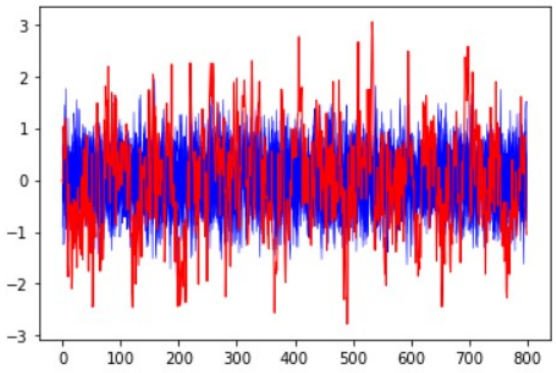


Figure 4.3: Simulated time series using state space model; blue indicates normal time series and red indicates abnormal time series.

the abnormal sensor has the highest score. The following algorithm 3 explains the weight-optimizing process.

Algorithm 3: Grid search optimization to optimize weights

Input:

Data frame with 3 parameters: number of anomalies (A),
vertical distance (V),
DTW distance (D),
for each simulated time series $t_i, i = 1, 2, ..., 11$ (t_{11} is the abnormal time series).

Output:

Optimum weights w_1, w_2, w_3

$diff = 0$;

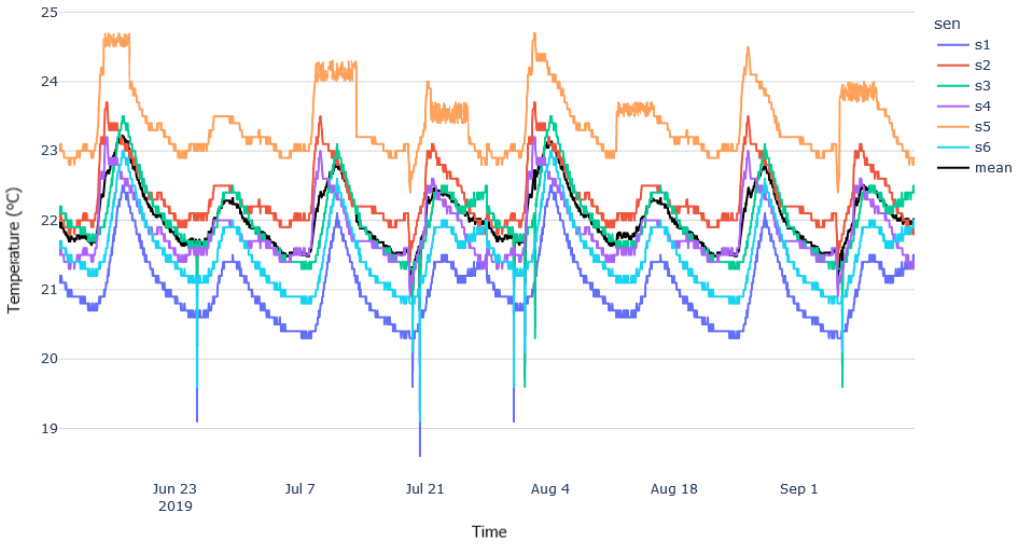
1: **while** $w_1 + w_2 + w_3 = 100$ **do**
2: Calculate the anomaly score for each time series.
$\psi_i = w1 * A_i + w2 * V_i + w3 * D_i$.
3: Store all the scores in $\Psi = [\psi_1, \psi_2, ..., \psi_{11}]$
4: **if** $is.max(\psi_{11}) = TRUE$ & $(\psi_{11} - min(\Psi)) > diff$ **then**
5: Update $diff = \psi_{11} - min(\Psi)$
6: Update optimum weights $= [w_1, w_2, w_3]$
7: **end if**
8: **end while**

4.2.4 Data Collection and Preparation

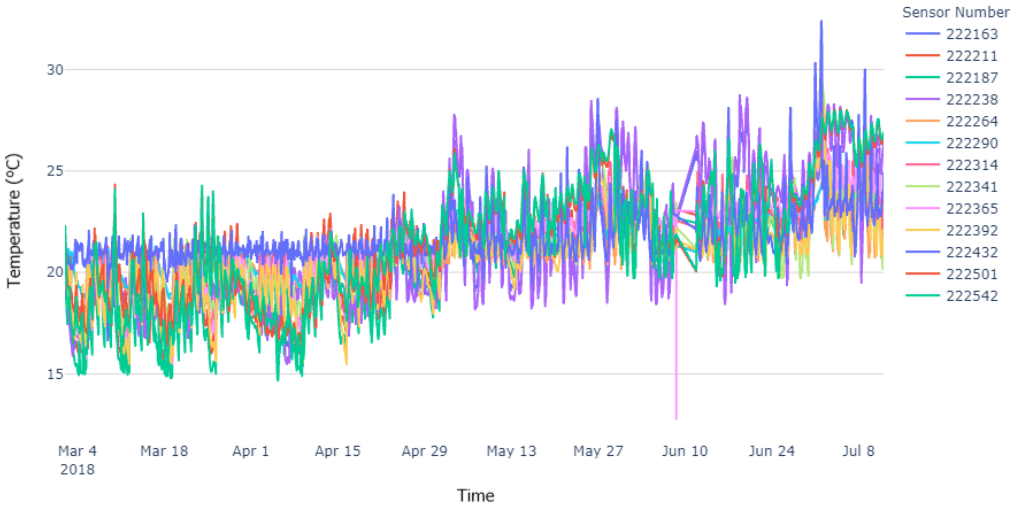
In order to test our algorithm, we used synthetic data which contains both collective and contextual anomalies, and data from a school building. For the synthetic dataset, we used an existing time series dataset and recreated similar time series with contextual anomalies and one different time series with collective anomalies. This synthetic dataset will help us to evaluate whether our method is capable of detecting the most abnormal time series due to collective anomalies.

The school building is located in New York City, USA. New York school building's data [3] were collected in 2018 from March to July (summer season), and 11 sensors were placed inside the building to collect data. The following two plots in Figure 4.4 show the variations and trends of the temperature time series of the synthetic data and the school building data.

Based on Figure 4.4, in synthetic data, we can clearly see some collective outliers in the 'orange' color time series. Other time series are similar to mean time series, but they show some contextual (single) outliers. Since our algorithm is designed to identify anomalies due to environmental issues, it should recognize the 'orange' color time series as the most abnormal one. In the New York building's data, we observe many abnormal points and sensor data with different patterns and higher variation. However, our method should be able to identify abnormal sensors in both scenarios, where the variation is high and low.



(a)



(b)

Figure 4.4: Temperature variation with time plots (a): Synthetic data, (b): New York School building

4.3 Results

4.3.1 Current Anomaly Detection Methods vs DTW Based Anomaly Detection Method

Popular anomaly detection methods mainly identify abnormal points, not an abnormal time window. In this study, we tested the anomaly detection ability of some of these popular methods: STL decomposition, Isolation Forest, Forecasting method, and DTW method. Plots in Figure 4.5 shows a sample temperature time series from a building and how each different method identifies anomalies of that time series. Temperature sharply decreases from 23^{o}C to 21^{o}C and increases back to 23^{o}C within a short period of time. Facility managers would like to identify these type of time windows, which shows indoor environmental abnormal behaviors.

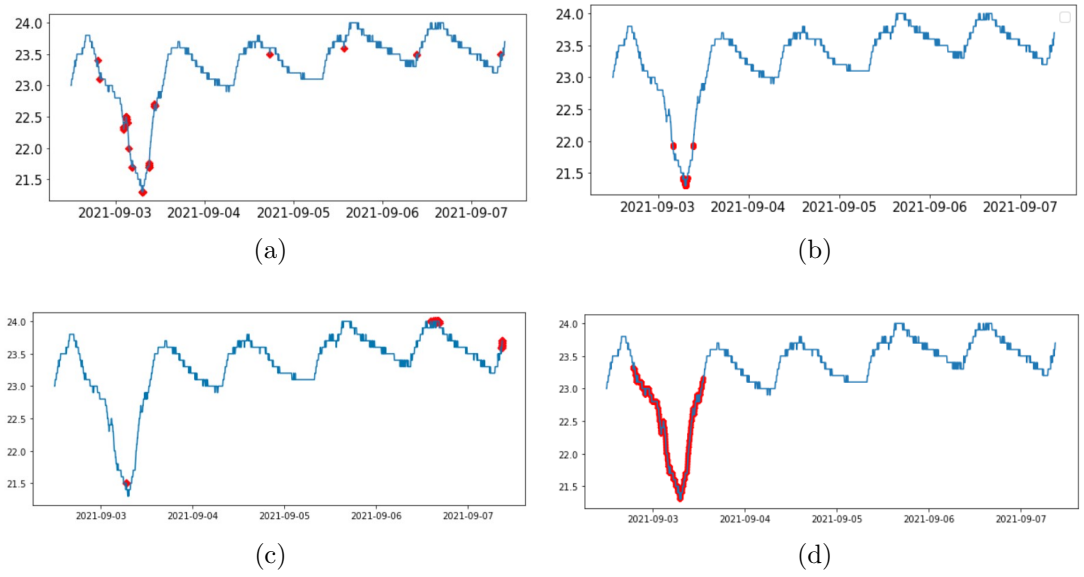


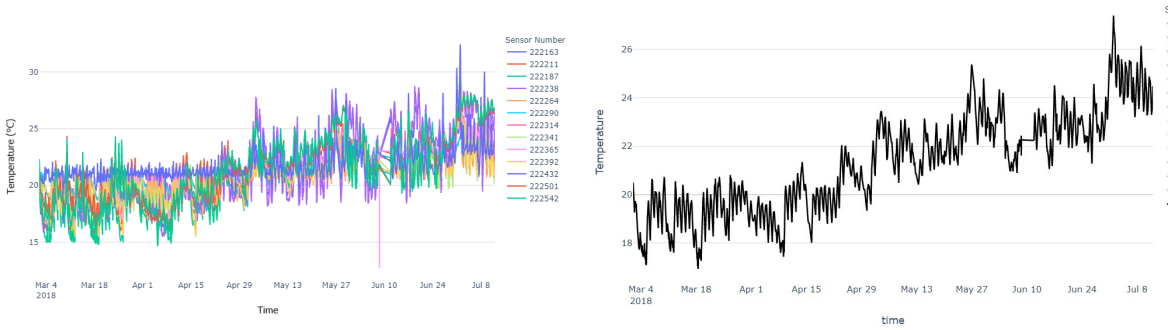
Figure 4.5: Results of anomaly detection methods (a): STL decomposition method, (b): Isolation Forest method, (c): Forecasting method, (d): DTW-based method. Anomaly points are shown in red color points.

Based on the results in Figure 4.5, we could see that only the DTW-based method identifies collective anomalies in a perfect way that we want to detect building anomalies. Hence we proceeded with the DTW-based method described previously. This method helped to identify abnormal time windows, not only the abnormal points, providing more applicable results in order to identify abnormal time series in buildings.

4.3.2 Abnormal Sensors Detection

New York School Building

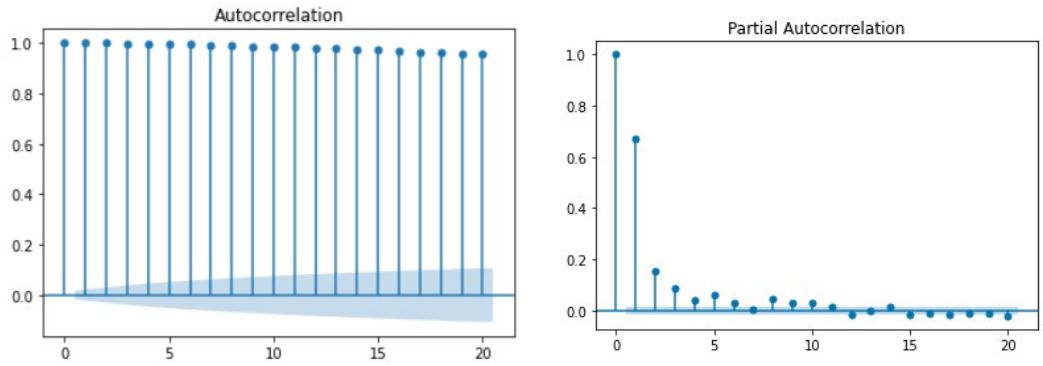
In this section, we will discuss the results of the abnormal sensor detection process of the New York building. As we explained in the methodology section, first we generated the median time series and considered it as the control time series. Figure 4.6 shows temperature vs time plots which contain all the sensors and the median time series of those sensor data.



(a) (b)

Figure 4.6: Time series plots of New York building (a): Containing time series of all the sensors and (b): Median time series of those sensors.

Then we needed to identify the time series model that best fits the median time series. First, we checked the stationarity of the time series using the Augmented Dickey-Fuller test. Based on the results the time series was a non-stationary time series ($p = 0.0835$; failed to reject the null hypothesis). Then we had to choose the ARIMA model instead of the AR, MA, and ARMA models. After the 1st order differencing, the p-value drops beyond the 0.05 threshold. Hence we could consider the order of differencing as 1. To fit the models, we had to identify the order of AR and MA models and the following PCAF and ACF plots in Figure 4.7 helped us to identify those orders.



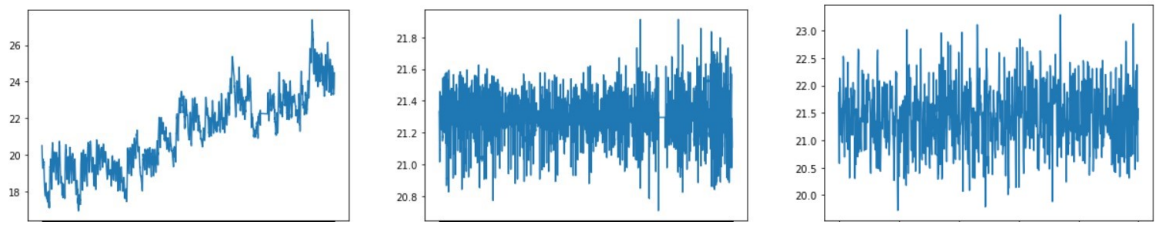
(a) (b)

Figure 4.7: (a): Auto-correlation Function (ACF) plot, (b): Partial Auto-correlation Function (PACF) plot.

Based on the PACF plot, it shows high partial autocorrelation values at $lag = 1$, but significantly drops at $lag = 2$. Hence we assigned 1 as the order of the AR model. In the ACF plot, we observe significantly high autocorrelations at all lags. Therefore, we selected 1 as the order of the MA model, since it is better to consider a simpler model. Using the orders of AR, and MA processes and order of differencing we fitted the ARIMA model, ARIMA(1,1,1). Then we used the estimated parameters of the

ARIMA model to simulate the time series. The fitted ARIMA model is:

$$y'_t = 21.3274 + 0.8239 * y'_{t-1} - 0.2899 * \varepsilon_{t-1}. \tag{4.9}$$



(a) (b) (c)

Figure 4.8: Plot (a): Original time series, (b): De-trended time series, (c): Simulated time series.

The original median time series had an increasing trend and using the differencing method we could detrend and make it stationary to fit the time series model. Once we detrended the time series we could compare the de-trended time series with the simulated time series. Figure 4.8 shows the original median, de-trended, and a simulated time series.

After that, we generated an abnormal time series, and using the grid search, we explained in algorithm 3, we could optimize the weights of the score function (ψ). The final score function of the i^{th} time series of this building, using the weighted sum of the number of anomalies (A_i), vertical distance (V_i), and DTW distance (D_i), is as follows:

$$\psi_i = (70 * A_i) + (29 * V_i) + (1 * D_i) \tag{4.10}$$

The final score values and ranks of each sensor, based on the anomaly count, vertical distance, and DTW distance compared to the real median time series,

Sensor	Count	V_dist	DTW	Score	Rank
222163	0.6616	0.7501	0.5073	68.57	11
222211	0.1514	0.4352	0.3214	23.54	3
222187	0.3736	0.8155	0.6288	50.43	9
222238	1.0000	1.0000	1.0000	100.00	13
222264	0.4758	0.1464	0.3763	37.93	8
222290	0.3626	0.0000	0.1644	25.55	4
222314	0.0832	0.0387	0.0798	7.03	2
222341	0.1963	0.4584	0.2526	27.28	5
222365	0.0000	0.1308	0.000	3.79	1
222392	0.3423	0.2622	0.3287	31.89	7
222432	0.2168	0.5017	0.4455	30.17	6
222501	0.5323	0.7906	0.57.58	60.77	10
222542	0.7143	0.8163	0.7007	74.37	12

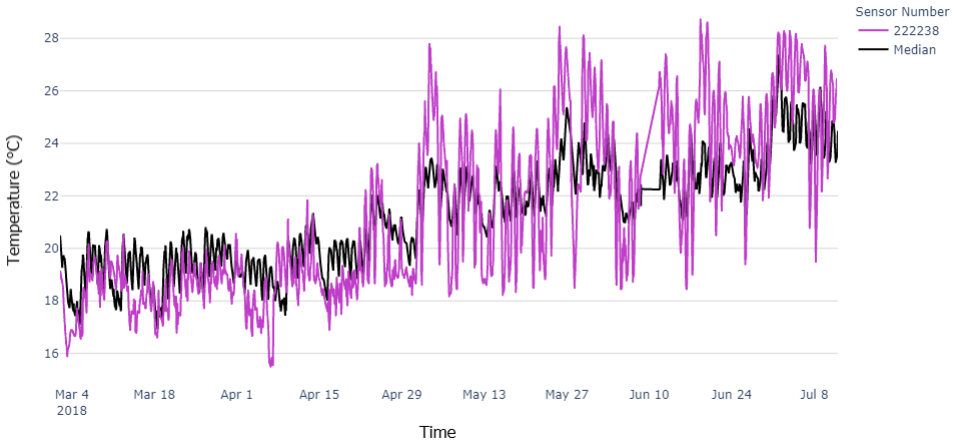
Table 4.2: Result table of New York building: The most abnormal sensor is highlighted in red, and the least abnormal sensor is highlighted in green.

are shown in the following Table 4.2. Based on the table 'sensor 222238' can be considered as the most abnormal sensor while 'sensor 222365' is the least abnormal sensor. It can be clearly seen that 'sensor 222365' follows the median time series though it has one outlier point in the middle of the time series. Figure 4.9 shows the most abnormal and least abnormal sensors compared to the median time series.

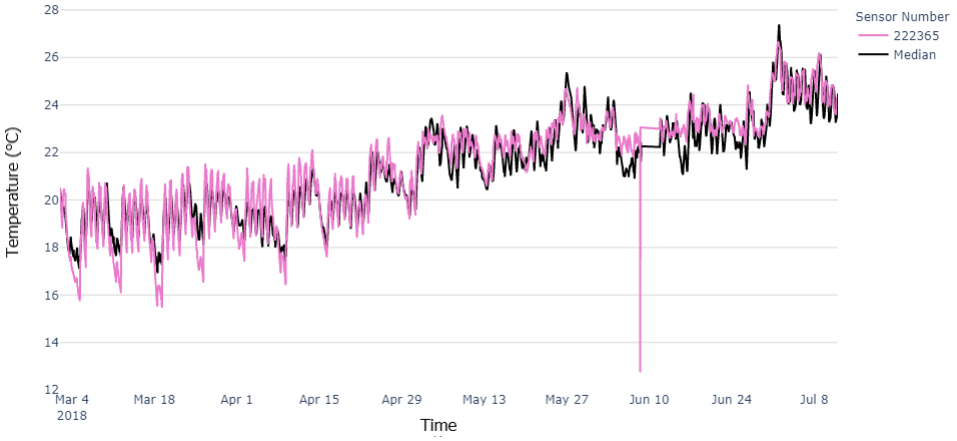
Synthetic Data Set

This section explains the results of anomaly detection of synthetic data. As with the New York analysis, first, we had to take the median time series by considering the time series of all sensors. The time series of each sensor and median time series are shown in Figure 4.10.

We checked the stationarity of the median time series which was found to be non-stationary (p-value = 0.3658). The difference was taken in order to make the median time series stationary. Then we end up with the ARIMA (2,1,2) model as



(a)



(b)

Figure 4.9: Time series plots of New York building (a): The most abnormal sensor. (b): The least abnormal sensor.

the best-fitted model. The fitted ARIMA model is,

$$y'_t = 22.0499 + 0.1174 * y'_{t-1} + 0.8050 * y'_{t-2} - 0.0427 * \varepsilon_{t-1} - 0.8837 * \varepsilon_{t-2}. \quad (4.11)$$

Then we generated an abnormal time series and 10-time series using the selected

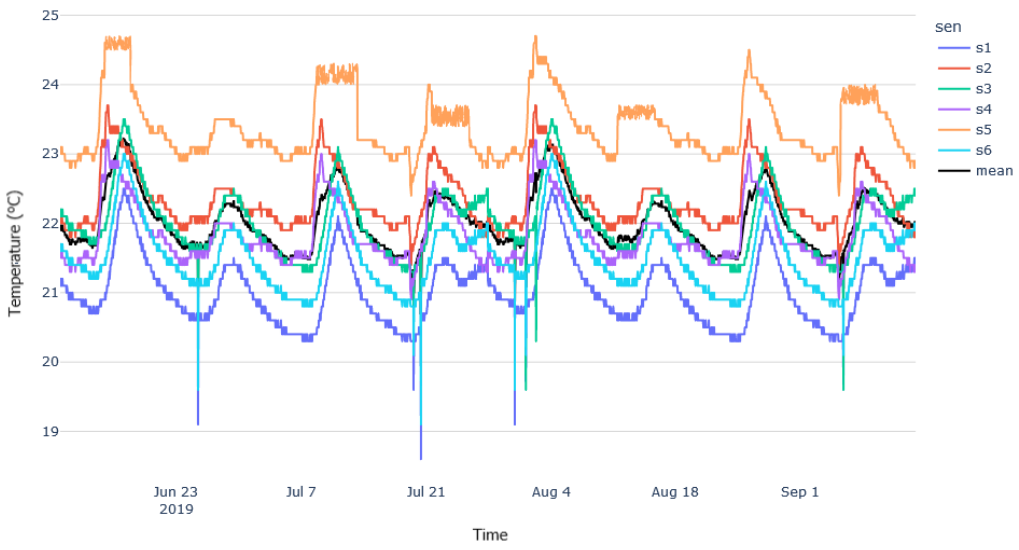


Figure 4.10: Time series of sensors in the synthetic data set.

model. Using those time series and grid search methods we found the weights for the score function of each i^{th} time series as follows,

$$\psi_i = (31 * A_i) + (68 * V_i) + (1 * D_i). \tag{4.12}$$

After that, we calculated the actual anomaly count, mean vertical distance from an anomaly to the respective median time series points, and DTW distance for each sensor. Scores were found for each sensor using the score function which is summarized in Table 4.3.

Based on the scores, the 'sensor S5' can be considered as the most abnormal sensor while the 'sensor s3' is the least abnormal sensor. Figure 4.11 shows the difference between these two time series plots. The most abnormal sensor clearly stands out from the median time series.

sensor	count	v_dist	DTW	score	rank
S1	0.594	0.594	0.593	59.44	4
S2	0.0000	0.0000	0.137	0.137	3
S3	0.00	0.00	0.00	0.00	1
S4	0.00	0.00	0.046	0.046	2
S5	1.00	0.949	1.00	96.591	6
S6	0.0000	1.0000	0.1406	68.25	5

Table 4.3: Results table of Manitoba school building: The most abnormal sensor is highlighted in red, and the least abnormal sensor is highlighted in green.

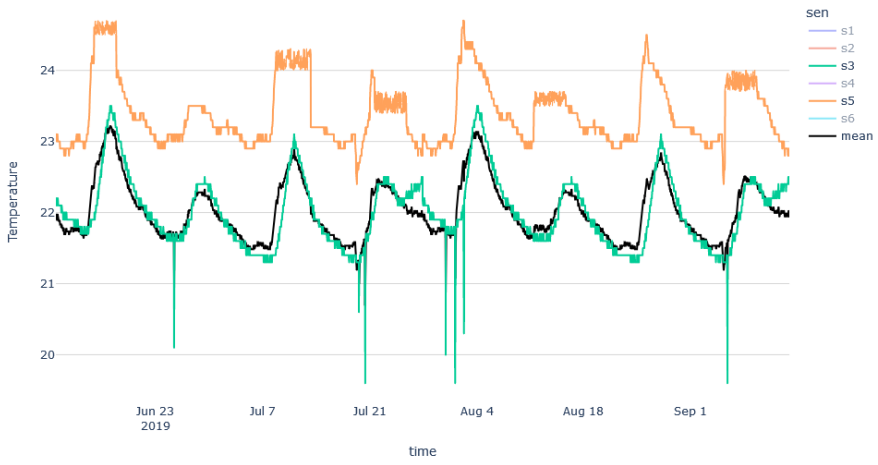


Figure 4.11: The most abnormal time series (orange), and the least abnormal time series (green), based on the median time series (black).

4.4 Discussion

This study was conducted in order to identify the locations with abnormal behaviors of indoor temperature, and we selected two school buildings to test our method. Based on the results of two test cases, it is clearly shown that our algorithm using the weighted scoring system is capable of identifying sensors with abnormal behaviors. Sensors with low scores show a better alignment with the median time series while the sensors with higher scores show significant changes in the pattern of time series

when compared with the median series.

Identifying abnormal performances of buildings is important for many parties in different ways. It helps building occupants improve their own comfort, while owners can save money by reducing energy waste and utility costs. Hence anomaly detection in buildings is an interesting research work, which is attracted by many building owners, tenants, and researchers.

Where our methodology is novel is in analyzing building conditions with that non-permanent hardware, allowing for a greater breadth and depth of data collection in a facility than existing methods. Our ability to detect anomalies is able to detect not only environmental anomalies with more specificity than traditional building monitoring systems, but we are also able to compare our results against a building's existing control system and thermostats. In this way, we can also identify whether existing thermostats and other sensors are sending accurate, properly calibrated data to the rest of the HVAC system, or if there are any false positive or negative results being shared within the system.

This anomaly detection process in buildings can help identify areas of concern that are difficult to find observationally or require a significant and costly energy audit to diagnose. These anomalies can be tied back to issues with a building's mechanical system, spaces that are not conditioned to their real occupancy use, inefficient control systems, inefficient building envelope, insulation or window issues, and beyond. Anomaly detection could also help identify thermostats or other data collection equipment that are poorly performing or miscalibrated and may be

transmitting false data that has a negative impact on that space.

However, an additional benefit of this methodology is to be able to test a building on multiple occasions to verify long-term trends and any degradation or loss of indoor environmental quality over time. A pattern of increasing frequency or severity in anomaly detection can be a key indicator of failing or degrading building conditions. Preventative analytics and fault detection can help to identify the presence of these issues before they reach a failure point, which will allow facilities managers to proactively identify and resolve known issues.

Though we used the median time series as the control time series, building owners and operators can change it to a known control time series, if available within the building. For example, if the building is on a temperature setpoint schedule, instead of the median time series, they can easily change this model to identify sensors that do not follow the setpoint schedule.

Not all anomalies are ignorable, and sometimes we do not want to consider some series as abnormal series, though we have some degree of abnormality. Hence, for future work, we are planning to improve our algorithm in a way that can identify whether the anomaly is acceptable or ignorable automatically. In this study, we only considered univariate anomalies, but in the future, we would like to develop a multivariate abnormal sensor detection algorithm as it can help to identify the performance issues of the building, quicker and more accurately than a human could.

Bibliography

[1] A. Bajaj. Anomaly detection in time series. https://neptune.ai/blog/anomaly-detection-in-time-series#:~:text=Anomaly. Accessed 20 April 2022.

[2] Z. Benkő, T. Bábel, and Z. Somogyvári. Model-free detection of unique events in time series. *Scientific Reports*, 12:1–17, 2022.

[3] P. Biam. Rtem hackaton api and data science tutorials. https://www.kaggle.com/datasets/ponybiam/onboard-api-intro?select=all_points_metadata.csv/, 2022. Accessed 03 May 2022.

[4] Y. Bing, D. H. C. Van Paassen, and S. Riahy. General modeling for model-based fdd on building hvac system. *Simulation Practice and Theory*, 9:387–397, 2002.

[5] P. J. Brockwell and R. A. Davis. *Time Series: Theory and Methods (2nd ed.)*. New York: Springer, 2009.

[6] M. C. Chuah and F. Fu. Ecg anomaly detection via time series analysis. In *Frontiers of High Performance Computing and Networking ISPA 2007 Workshops*, pages 123–135, 2007.

[7] J. da Silva Arantes, M. da Silva Arantes, H. B. Fröhlich, L. Siret, and R. Bonnard. A novel unsupervised method for anomaly detection in time series based on statistical features for industrial predictive maintenance. *International Journal of Data Science and Analytics*, 12:383–404, 2021.

[8] N. Debanjana and P. Harry. Automated real-time anomaly detection of temperature sensors through machine learning. In *International Journal of Sensor Networks*, volume 34, page 137–152. Interscience Publishers, 2020.

[9] M. D. Diab, A. Basil, B. Hamad, L. Sangarapillai, G. K. Konstantinos, and G. Ibrahim. Anomaly detection using dynamic time warping. In *2019 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)*, pages 193–198, 2019.

[10] R. Foorthuis. On the nature and types of anomalies: a review of deviations in data. *International Journal of Data Science and Analytics*, 12:297–331, 2021.

[11] W. A. Fuller. *Introduction to statistical time series*. John Wiley and Sons., 1976.

[12] J. C. Gower. Properties of euclidean and non-euclidean distance matrices. *Linear Algebra and its Applications*, 67:81–97, 1985.

[13] Y. Jung, T. Kang, and C. Chun. Anomaly analysis on indoor office spaces for facility management using deep learning methods. *Journal of Building Engineering*, 43:103139, 2021.

[14] G. Li and J. J. Jung. Entropy-based dynamic graph embedding for anomaly detection on multiple climate time series. *Scientific Reports*, 11:1–10, 2021.

[15] F. Liu, Y. Lee, H. Jiang, J. Snowdon, and M. Bobker. Statistical modeling for anomaly detection, forecasting and root cause analysis of energy consumption for a portfolio of buildings. *Proceedings of Building Simulation 2011: 12th Conference of International Building Performance Simulation Association*, pages 2530–2537, 2011.

[16] F. T. Liu, K. M. Ting, and Z. Zhou. Isolation-based anomaly detection. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 6:1–39, 2012.

[17] H. F. Lopes. Ar(1) plus noise model, 2017. Accessed: 15 April 2022.

[18] T. C. Mills. *Time series Techniques for Economics*. Cambridge University Press, 1990.

[19] G. Moschini, R. Houssou, J. Bovay, and S. Robert-Nicoud. Anomaly and fraud detection in credit card transactions using the arima model. *Engineering Proceedings*, 5, 2021.

[20] J. K. Muhammad. Grid search optimization algorithm in python. https://stackabuse.com/grid-search-optimization-algorithm-in-python/, 2020. Accessed 20 May 2022.

[21] T. G. Puranik and D. Mavris. Anomaly detection in general aviation operations using energy metrics and flight data records. *Journal of Aerospace Information Systems, American Institute of Aeronautics and Astronautics*, 15:22–35, 2018.

[22] C. Robert, C. William, and T. Irma. Stl: A seasonal-trend decomposition procedure based on loess. *Journal of Official Statistics*, 6:3, 1990.

[23] D. Savage, X. Zhang, X. Yu, P. Chou, and Q. Wang. Anomaly detection in online social networks. *Social Networks*, 39:62–70, 2014.

[24] S. Schmidl, P. Wenig, and T. Papenbrock. Anomaly detection in time series: A comprehensive evaluation. *Proc. VLDB Endow.*, 15:1779–1797, 2022.

[25] P. Tormene, T. Giorgino, S. Quaglini, and M. Stefanelli. Matching incomplete time series with dynamic time warping: An algorithm and an application to post-stroke rehabilitation. *Artificial Intelligence in Medicine*, 45:11–34, 2008.

[26] Z. Wang, T. Parkinson, P. Li, B. Lin, and T. Hong. The squeaky wheel: Machine learning for anomaly detection in subjective thermal comfort votes. *Building and Environment*, 151:219–227, 2019.

[27] L. Wei, J. Hongyi, C. Dandan, C. Lifei, and J. Qingshan. A real-time temperature anomaly detection method for iot data. In *5th International Conference on Internet of Things, Big Data and Security*, pages 112–118, 2020.

[28] A. Wickramasinghe, S. Muthukumarana, D. Loewen, and M. Schaubroeck. Temperature clusters in commercial buildings using k-means and time series clustering. *Energy Informatics*, 5, 2022.

[29] H. Yassine, G. Khalida, A. Abdullah, B. Faycal, and A. Abbes. Artificial intelligence based anomaly detection of energy consumption in buildings: A review, current trends and new perspectives. *Applied Energy*, 287:116601, 2021.

[30] M. Zheng, S. Domanskyi, C. Piermarocchi, and G. I. Mias. Visibility graph based temporal community detection with applications in biological time series. *Scientific Reports*, 11:1–12, 2021.

Chapter 5

Enhanced Anomaly Detection through a Bayesian Framework with a Novel Network Merging Structure Learning Approach

Structure learning aims to uncover the underlying dependencies or relationships between variables in a network. Traditional methods rely on statistical inference to reveal causal relationships or conditional dependencies. However, there is a growing need for innovative approaches that enhance network merging and improve both the accuracy and scalability of structure learning. In this study, we propose a novel approach that merges network structures to develop an improved framework, considering the strength of conditional dependencies between nodes. We evaluate this approach using environmental quality data collected from a commercial building to predict the likelihood of abnormal conditions. By integrating environmental data from neighboring locations, we enhance the accuracy of anomaly detection. To construct the base Bayesian networks, we use structure learning methods and

incorporate three scoring techniques: the Bayesian Information Criterion (BIC), the K2 score, and the Bayesian Dirichlet score. Our novel method effectively removes unrealistic dependencies using some domain knowledge that traditional structure learning methods fail to address. Finally, the comparative analysis demonstrated that this Bayesian network model, with our structure learning method, demonstrates better performance in anomaly detection compared to classical structure learning approaches. This can be improved for real-time anomaly detection in environmental quality monitoring systems.

5.1 Introduction

Bayesian Networks (BNs) are powerful probabilistic models that represent the conditional dependencies among a set of variables through a directed acyclic graph [11]. Each node in the network corresponds to a variable, and the edges denote probabilistic dependencies between these variables [16]. BNs are widely used for modeling uncertainty, reasoning with incomplete data, and making predictions across various domains.

Structure learning is an important step in BN model development [12]. It involves determining the optimal network structure that accurately reflects the dependencies among variables. The quality of structure learning is crucial because the structure determines how well the network captures the underlying relationships within the data. Accurate structure learning ensures that the probabilistic dependencies are modeled correctly, leading to more reliable inferences and predictions. Conversely, an incorrect network structure can result in misleading conclusions, inaccurate pre-

dictions, and ineffective decision-making. Therefore, developing robust and efficient methods for structure learning is essential to fully realize the potential of Bayesian Networks.

The primary methods for structure learning are constraint-based, score-based, and hybrid approaches. We will discuss these methods in detail in the methods section. Choosing the appropriate method for a given dataset can be challenging. In this study, we introduce a novel approach that merges multiple network structures to create an optimized final network structure, enhancing overall modeling. Del Sagrado and Moral [4] proposed a method for combining Bayesian networks based on the union and intersection of independencies. Later, Hu et al. [10] introduced a qualitative method for combining networks, focusing on the properties of conditional probability. Feng et al. [6] developed a scoring function to assess node quality, using these scores to guide the merging process. More recently, Vaniš and the team [20] proposed a scoring method that incorporates both domain knowledge and algorithmic insights for merging network structures.

Our proposed algorithm differs from these previous methods in several ways. While earlier approaches focus on qualitative methods or scoring-based algorithms to preserve structural and parametric characteristics, our algorithm prioritizes improving accuracy while maintaining structural integrity. This is a score-based and framework-based method. This aims to integrate dependencies identified by classical techniques, aiming to achieve higher prediction accuracy.

In the context of anomaly detection, Bayesian Networks offer a robust framework for identifying unusual patterns or outliers in data. When used with labeled data for

classification, BNs can effectively detect anomalies by learning from historical data with labeled normal and abnormal conditions. Previous studies have demonstrated the use of Bayesian Networks in anomaly detection across various fields [13, 5, 18].

In this study, we focus on applying Bayesian Networks for anomaly detection using labeled data, aiming to enhance accuracy by optimizing the network's representation of variable dependencies. This approach not only improves the detection of anomalies but also provides a valuable tool for classification tasks. The labeling of anomalies is an important part and we use our previously developed method for detecting environmental anomalies to label data [22]. The environmental features of the current and neighboring locations are used as independent variables in the Bayesian model. In order to identify the optimal number of nearest neighbors we used the novel method that we introduced using Moran's I statistics [21].

Our work highlights the critical role of precise structure learning in maximizing the effectiveness of Bayesian Networks, particularly in applications involving anomaly detection and classification.

5.2 Material and Methods

In this section, we provide the theoretical background necessary for understanding our approach and introduce our structure learning method in detail.

5.2.1 Bayesian Network

A causal network consists of a set of variables and a set of directed links (or edges) between these variables. Mathematically, the structure is called a directed acyclic

graph $(DAG), G = (V, E)$, , where V is a set of random variables $\mathbf{X} = (X_1, X_2, ..., X_n)$, also known as nodes and E is a set of edges that represent conditional dependencies between nodes. If there is a link from X_1 to X_2, we say that X_2 is a child of X_1, and X_1 is a parent of X_2. For each node X_i the corresponding probability distribution can be written as $P(X_i) = P(X_i|Pa(X_i))$, where $Pa(X_i)$ denotes the set of parents of X_i in G. Using the chain rule for Bayesian networks, we can write the joint probability distribution over all variables as follows;

$$P(X_1, X_2, ..., X_n) = \prod P(X_i|Pa(X_i)). \tag{5.1}$$

Conditional Probability Tables (CPTs) quantify the strength of probabilistic dependencies between variables [12]. CPTs provide the "weights" of these edges by specifying how the probability of the child node changes based on different configurations of its parent nodes.

5.2.2 Scoring Functions

Scoring functions play a crucial role in evaluating and selecting the optimal network structure. They provide a measure for comparing different DAGs and guide the search for the network structure that best represents the underlying probabilistic dependencies.

K2 Score

The K2 score was proposed by Cooper and Herskovits [3] and it measures the goodness of fit of a Bayesian network structure to the data while penalizing for

model complexity. It takes into account the likelihood of the data given the network structure and also incorporates prior knowledge in the form of equivalent sample sizes for the parameters. The equation for the K2 score is represented as follows:

$$g_{\mathrm{K2}}(G:D) = \log(p(G)) + \sum_{i=1}^{n}\left[\sum_{j=1}^{q_i}\left[\log\frac{(r_i-1)!}{(N_{ij}+r_i-1)!} + \sum_{k=1}^{r_i}\log(N_{ijk}!)\right]\right] \quad (5.2)$$

where $p(G)$ represents the prior probability of the network structure G. The second term involves summing over all variables X_i, where q_i denotes the number of states for each variable. For each state j of X_i, it calculates the likelihood of observing the data given the number of parent configurations r_i and adds the factorial term of the counts N_{ijk} for each configuration. This formula combines prior knowledge with observed data to score the network structure's fit to the data.

Bayesian Dirichlet Score

The Bayesian Dirichlet Equivalent Uniform (BDeu) score is also designed to assess the goodness of fit of a Bayesian network structure to the data as a generalization of the K2 [9]. The BDeu score is particularly useful when dealing with datasets with limited samples. It incorporates prior knowledge in the form of equivalent sample sizes for the parameters, which helps in mitigating the effects of data sparsity. The BDeu score is computed as follows:

$$g_{\mathrm{BD}}(G:D) = \log(p(G)) + \sum_{i=1}^{n}\left[\sum_{j=1}^{q_i}\left[\log\left(\frac{\Gamma(\eta_{ij})}{\Gamma(\eta_{ij}+N_{ij})}\right) + \sum_{k=1}^{r_i}\log\left(\frac{\Gamma(N_{ijk}+\eta_{ijk})}{\Gamma(\eta_{ijk})}\right)\right]\right] \quad (5.3)$$

where the values η_{ijk} are the hyperparameters for the Dirichlet prior distributions of the parameters given the network structure, and $\eta_{ij} = \sum_{k=1}^{r_i} \eta_{ijk}$. However the specification of the hyperparameters η_{ijk} is quite difficult. Hence Buntine [2] proposed a new score which is called BDeu considering the additional assumption of likelihood equivalence. The score depends on one parameter, the equivalent sample size η, and is expressed as follows:

$$g_{\text{BDeu}}(G:D) = \log(p(G)) + \sum_{i=1}^{n} \left[\sum_{j=1}^{q_i} \left[\log \left(\frac{\Gamma(\frac{\eta}{q_i})}{\Gamma(\frac{\eta}{q_i} + N_{ij})} \right) + \sum_{k=1}^{r_i} \log \left(\frac{\Gamma(N_{ijk} + \frac{\eta}{r_i q_i})}{\Gamma(\frac{\eta}{r_i q_i})} \right) \right] \right]. \tag{5.4}$$

Bayesian Information Criterion (BIC)

BIC is another scoring function used to evaluate Bayesian network structures [19]. It balances the fit of a model against its complexity. The BIC for a Bayesian Network G given data D is calculated as:

$$BIC(G:D) = -2 \ * log(p(D|G)) + k \ * log \ (N) \tag{5.5}$$

where $-2 * log(p(D|G))$ is the negative log-likelihood of the data given the network G, k is the number of parameters in the model, and N is the number of observations. The BIC penalizes models with more parameters to prevent overfitting. It helps in choosing simpler models that still fit the data well.

5.2.3 Structure Learning Methods

Score Based Method

Score-based methods aim to find the network structure that maximizes a given scoring function [8]. Formally, given a complete training dataset D of instances for variables X_n, the goal is to identify a DAG, G^* that maximizes a given scoring function. This can be expressed as:

$$G^* = \arg\max_{G \in \mathcal{G}_n} g(G : D) \tag{5.6}$$

where $\mathcal{G}_n$ represents the space of all possible DAGs for the variables X_n, and $g(G : D)$ is the scoring function that evaluates how well the network structure G fits the data D. In this study, we used *ExhaustiveSearch* function in *pgmpy* package in Python [1].

Hill - Climbing Method

Hill-climbing is a score-based method that starts with an initial network and iteratively makes small changes to improve the score [17]. The process involves:

- Start with an initial network structure G_0.
- Generate neighboring structures by adding, deleting, or reversing edges.
- Calculate the score for each neighbor.
- Move to the neighbor with the highest score if it improves the current score.

The algorithm stops when no further improvements can be found, leading to a locally optimal solution. Mathematically, if G is the current structure, the algorithm considers neighboring structures G' and selects:

$$G_{new} = \arg \max_{G' \in Neighbors(G)} g(G' : D). \tag{5.7}$$

Constraint-Based Method

Constraint-based methods focus on identifying the dependencies among variables directly from the data. They use statistical tests such as the chi-squared test to determine whether conditional independence holds. The general approach involves:

- Identify independencies in the data set using hypothesis tests.
- Construct DAG (pattern) according to identified independencies

For example, if X_i and X_j are conditionally independent given X_k, there will be no edge between X_i and X_j in the final network structure:

$$X_i \perp X_j \mid X_k \implies \text{no edge between } X_i \text{ and } X_j. \tag{5.8}$$

Independencies in the data can be identified using chi-squared conditional independence tests. Then DAG patterns can be constructed using PC function in *pgmpy* package.

Hybrid Methods

Hybrid methods combine elements of both score-based and constraint-based approaches [7]. These methods first use constraint-based techniques to reduce the search space by identifying potential edges and then apply score-based techniques to optimize the network structure. The typical process includes:

- Identify possible edges based on conditional independence tests.
- Apply a scoring function to select the best structure from the reduced set of candidate networks.

This combination leverages the strengths of both approaches, ensuring both structural correctness and optimal scoring.

5.2.4 The Node Score

The score of the node is important for our merged network structure. The importance of a node is based on the strength of its relationship with parent nodes, which we measure using its predictive ability. There are various metrics to measure the prediction performance of classification models, such as accuracy, precision, recall, and the F1 score [15]. We have decided to use the F1 score because it offers a balanced measure of a model's performance, especially in cases with imbalanced classes or when both false positives and false negatives are significant.

In our network merging method, we saved the parents of each node based on each network structure generated using classic structure learning methods. Then the F1 score is calculated by predicting the child node variable using the parent variables,

with the training dataset. For example, in Figure 1, three different network structures will have distinct sets of parent nodes for node A. As a result, each structure will yield different F1 scores for the prediction of node A. If network structures are defined as G_1, G_2, and G_3 the respective F1 scores can be denote as $F1_{G1}(A), F1_{G2}(A), F1_{G3}(A)$. A higher F1 score indicates better prediction accuracy. We then select the parent nodes from the network structure that produces the highest F1 score as the most suitable parents for node A.

$$Pa(A)^* = Pa_{Gi^*}(A);\ where\ i^* = \arg \max_{i \in \{1,2,\ldots,n\}} F1_{Gi}(A)$$

These selected edges are then added to the edge list of the final network. Referring to Figure 5.1, if A in $G1$ yields the highest F1 score, then B and C will be the parent nodes of A in the final network.

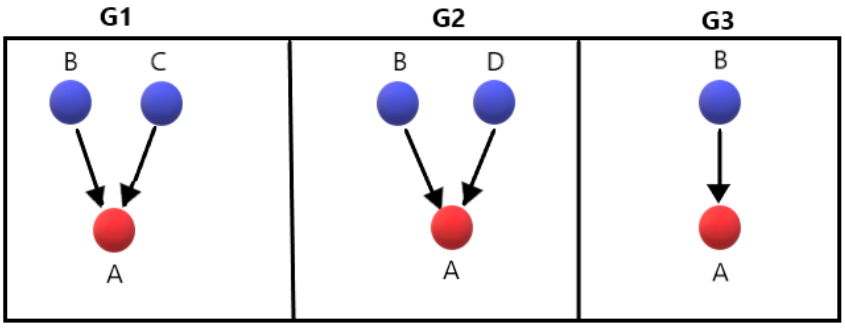


Figure 5.1: G1, G2, and G3 show the parents of node A based on different network structures.

F1 score

The F1 score is a performance metric that balances precision and recall in classification models, especially useful in imbalanced datasets. It is the harmonic mean of precision (P) and recall (R), where precision is defined as $P = \frac{TP}{TP+FP}$ and recall is $R = \frac{TP}{TP+FN}$. Here, TP represents true positives, FP denotes false positives, and FN indicates false negatives. The F1 score is calculated using the formula:

$$F1 = 2 \times \frac{P \times R}{P + R}. \tag{5.9}$$

This metric provides a single value that captures both the accuracy of positive predictions and the ability to identify all relevant instances, making it particularly useful when the class distribution is skewed or when both precision and recall are critical.

5.2.5 Novel Merging Method for Structure Learning

In this section, we will explore the Merging Algorithm for combining multiple Bayesian networks into a single network to enhance prediction accuracy. The algorithm follows these main steps.

- Generate Bayesian networks based on existing structure learning methods; $G_{list} = \{G_1, G_2, ..., G_k\}$.
- Calculate the score of each node based on its parent nodes in each network structure and identify the best parent variables.

- Finally, generate the network structure, or final edge list, based on the selected parents of each node.

When finalizing the edge list, it is crucial to remember that Bayesian Networks cannot include bidirectional edges or cycles. Therefore, the direction of each edge was determined based on the node's score. A higher score indicates a stronger dependency between the parent and child nodes, reflecting their relative importance in the network. For two nodes X_i and X_j, the edge direction is determined by:

$$\text{Direction}(X_i \to X_j) = \arg \max_{d \in \{\text{parent,child}\}} \text{Score}_{ij}(d). \tag{5.10}$$

Remove Spurious Dependencies

In Bayesian networks, dependencies where one variable's influence on another is unrealistic or nonsensical are often referred to as "spurious dependencies". These are relationships that may appear statistically significant but lack a meaningful or causal basis.

During structure learning, it is possible to encounter spurious dependencies between some variables. Our merging algorithm addresses this issue by removing such dependencies when constructing the Bayesian network. This can help ensure that the model only includes realistic and meaningful relationships.

Let $E(G)$ be the set of edges in G and $\hat{G}$ is a learned network structure based on D. In our algorithm, we maintain a set S of known spurious dependencies $S \subseteq E(\hat{G})$. When constructing the final Bayesian network G_{final}, we ensure that only edges not

in S are included. This is expressed as:

$$E(G_{final}) = E(\hat{G}) \backslash S \tag{5.11}$$

where $E(G_{final})$ denotes the set of edges in the final network. By applying this method, we incorporate domain knowledge into the structure learning process.

Merging Algorithm

The detailed algorithm for merging multiple network structures to generate the final network structure is presented in algorithm 4. This algorithm combines all the methods discussed in Section 5.2.3 on creating Bayesian network structures.

Algorithm 4: Network Merging Algorithm

Input:
Variables (nodes) $\mathcal{V}$, set of spurious dependencies S,
Generated Bayesian Networks $\{G_1, G_2, \ldots, G_k\}$
Output:
Final Bayesian Network edge list G_{final}

1: **for** each node X_i in $\mathcal{V}$ **do**
2: Initialize `parents_list` as empty.
3: Initialize `score_list` as empty.
4: **for** each network G_m in $\{G_1, G_2, \ldots, G_k\}$ **do**
5: Find parents $\text{Pa}_{G_m}(X_i)$ of node X_i in network G_m.
6: `parents_list` $\leftarrow \{\text{Pa}_m(X_i)\}$
7: Compute the score $S^m_{X_i}$ for node X_i in network G_m based on parents.
8: `score_list` $\leftarrow \{S^m_{X_i}\}$
9: **end for**
10: Add parents of node X_i with highest S_{X_i} to the `edge_list`.
11: Remove edges in S list from `edge_list`, $S \notin E(G)$
12: **end for**
13: Sort $\mathcal{V}$ based on S_{X_i}.
14: Sort `edge_list` based on sorted $\mathcal{V}$.
15: Add edges from `edge_list` to G_{final}.

5.2.6 Bayesian Framework for Anomaly Prediction

As discussed in the introduction, we develop a classification model to predict the abnormality of a location. The first step is to label each observation as 'normal' or 'abnormal'. To do this, we use the method applied in our previous work to detect environmental outliers [22]. This method identifies collective anomalies by considering the overall thermal condition of a building floor.

Now we can evaluate our classification model, as we identified the ground truth. Next, we need to find the neighbors of the current location and use their environmental feature data to enhance the prediction of the current location's abnormality. To identify the optimal number of nearest neighbors, we use the novel method proposed in one of our previous studies [21]. This method combines the nearest neighbors algorithm with Moran's I statistic, a measure of spatial autocorrelation [14].

Next, we need to discretize the continuous variables and preprocess the data. We can then generate multiple Bayesian networks using the classic structure learning methods discussed in Section 5.2.3. These networks are then merged to obtain the final merged network structure using the method proposed in Section 5.2.5. The flowchart in Figure 5.2 summarizes the anomaly detection framework.

5.3 Results

In this section, we present the results of applying Bayesian network analysis to anomaly prediction. We compare the outcomes of classical methods with those obtained from our proposed network merging structure learning method. We evaluated

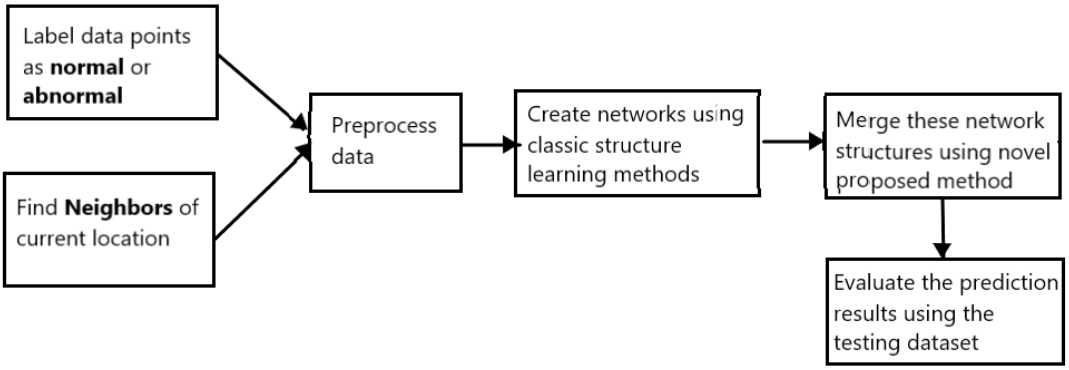


Figure 5.2: Bayesian network-based anomaly detection framework

our approach using a commercial building dataset, which was tested under three different training-to-testing ratios. This setup allowed us to determine whether a larger amount of training data provides a significant advantage for accurate prediction.

5.3.1 Case Study - Commercial Building in Downtown Winnipeg

To test our algorithm, we used data from a commercial building in downtown Winnipeg. The data, collected at 5-minute intervals in November 2021, included 11 sensors placed throughout the building to gather environmental information. These sensors were positioned in various locations and collected data over time. We then applied our anomaly detection method to label each data point and categorize the continuous variables. A sample of the final dataset is shown in Table 5.1.

TimeStamp	Humidity	Temperature	CO2	weekday	ptype
2021-11-29 18:05:00	Medium	Medium	Ideal	Monday	normal
2021-11-29 18:10:00	Medium	Medium	Ideal	Monday	normal
2021-11-29 18:15:00	Medium	Medium	High	Monday	normal
2021-11-29 18:20:00	High	Medium	High	Monday	outlier
2021-11-29 18:25:00	High	Medium	High	Monday	outlier

Table 5.1: Sample dataset

Network Generation Using Classic Methods

As explained in Section 5.2.5, we first generated network structures using the classic methods discussed in Section 5.2.3. Figure 5.3, shows the network structures generated by the score-based method. The three networks exhibit slight differences in structure and dependencies due to variations in scoring functions.

Based on the classic structure learning methods, we observe some dependencies that do not make sense. For example, the weekday variable should not depend on temperature or humidity. Therefore, we will consider the merged network structure.

Network Generation Using Merging Methods

In our novel method, we assessed the node score for each variable using the approach described in Section 5.2.4. Sometimes, some nodes may have the same parent nodes across all networks. In such cases, we considered only the unique parent lists. The following table 5.2 displays the node scores for each variable in our study. Some nodes have only one or two scores because they have only one or two unique parent lists.

Here, we remove nodes with lower node scores, as lower scores indicate weaker prediction ability. For example, we remove the 'weekday' variable if its node score is

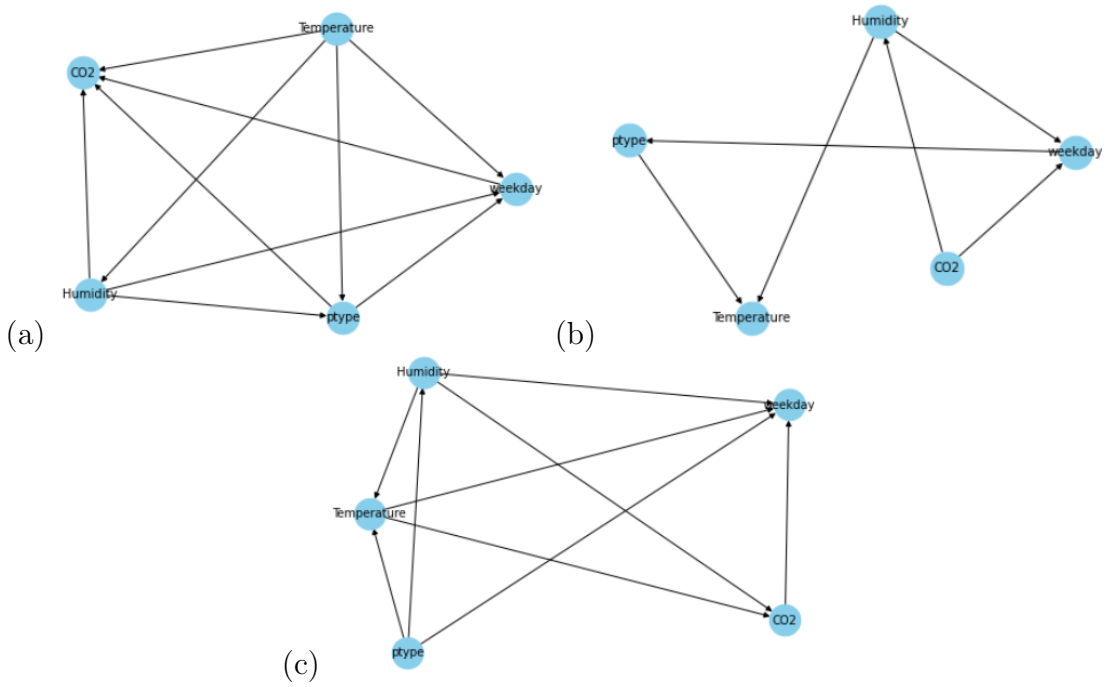


Figure 5.3: Network structures from the score-based method (a) k2 score, (b): BIC, and (c): Bayesian Dirichlet score.

Node	**Node Scores**		
Temperature	0.9804		
Humidity	0.5172	0.6273	0.4043
Weekday	0.2322	0.3003	0.4420
CO2	0.7395	0.6649	
Ptype	0.8096	0.9523	

Table 5.2: Node scores for each node

less than 0.5. The merged network structure generated from our novel method is shown in Figure 5.4. It does not contain any unrealistic relationships.

Finally, we evaluated and compared the prediction results of classical structure learning methods and our novel merging network method using Accuracy and F1 scores. The evaluation was conducted on randomly selected training and testing datasets with different proportions from the original dataset. The results of our

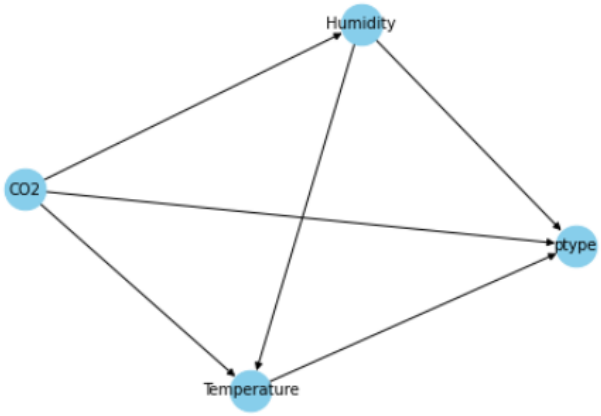


Figure 5.4: Network structure from the network merging method

experiment are shown in the Table 5.3.

Training Testing Ratio	Evaluation	Hybrid Learning			Hill Climbing			Merged Network Structure	Merged Network with Neighbors
		K2	BIC	Dirichlet	K2	BIC	Dirichlet		
0.7 | 0.3	Accuracy	0.7799	0.8822	0.8282	0.9305	0.8822	0.9228	0.9305	0.9384
	F1 score	0.7575	0.8752	0.8241	0.9300	0.8752	0.9216	0.9300	0.9642
0.8 | 0.2	Accuracy	0.7832	0.896	0.8555	0.9306	0.896	0.9306	0.9306	0.9556
	F1 score	0.7608	0.8953	0.8502	0.9306	0.8953	0.9306	0.9306	0.9757
0.9 | 0.1	Accuracy	0.8035	0.9041	0.8266	0.9249	0.9017	0.9306	0.8439	0.8801
	F1 score	0.7844	0.9041	0.8203	0.9249	0.9041	0.9306	0.8374	0.8726

Table 5.3: Summary of evaluation results: Red font represents the highest F1 scores, while blue font indicates the F1 scores of the merged network with neighbors if they are higher than the other F1 scores.

Based on the table, it is evident that the merged network structure outperforms the classic method discussed in Section 5.2.3 in terms of prediction accuracy. The constraint-based method produced a network with cycles or loops, making it unsuitable for Bayesian network model construction. When the training set is increased to 90%, the network structure using hill climbing and the Dirichlet score provides better prediction accuracy than the novel method. The last column presents the evaluation results incorporating environmental features (Temperature,

Humidity, CO2) from the 3 nearest neighboring locations. Including information from neighboring locations has improved prediction accuracy.

5.4 Conclusion

In this study, we explored the performance of Bayesian network analysis for anomaly prediction, comparing classical structure learning methods with our proposed network merging structure learning method. Our evaluation used data from a commercial building in downtown Winnipeg, where we assessed the performance of different network structures in capturing dependencies and improving prediction accuracy.

The results demonstrate that the merged network structure outperforms the classical methods in terms of prediction accuracy. This improvement is due to the merged network method's capability to more accurately capture complex dependencies and interactions among variables. Classical methods often produce networks with unrealistic relationships, which are not logically plausible. In this novel merging method, the algorithm removes these unrealistic dependencies.

By employing node scoring to filter out variables with lower prediction ability, our approach ensures that only the most relevant variables are included in the final network structure. This step helps avoid overfitting and simplifies the model, which contributes to its improved performance.

In conclusion, our research demonstrates that the novel merging network structure learning method offers a substantial improvement over classical structure learning techniques for Bayesian network development. This method, when applied to anomaly detection, demonstrates improved performance. By effectively integrating

environmental features from neighboring locations and using node scoring to refine the network structure, our method improves prediction accuracy and generates more realistic network models.

The findings from this study emphasize the value of incorporating advanced methods for network structure learning and the importance of data quality in model performance. Future research could focus on further refining the merging network method and applying it to predict the temporal behavior of anomalies using Dynamic Bayesian Networks.

Bibliography

[1] A. Ankan and A. Panda. pgmpy: Probabilistic graphical models using python. In *Proceedings of the Python in Science Conference*, SciPy. SciPy, 2015.

[2] W. Buntine. Theory refinement on bayesian networks. In *Proceedings of the Seventh Conference on Uncertainty in Artificial Intelligence*, page 52–60, 1991.

[3] G. F. Cooper and E. Herskovits. A bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 9:309–347, 1992.

[4] J. Del Sagrado and S. Moral. Qualitative combination of bayesian networks. *International Journal of Intelligent Systems*, 18:237–249, 2003.

[5] N. Ding, H. Gao, H. Bu, and H. Ma. Radm:real-time anomaly detection in multivariate time series based on bayesian network. In *2018 IEEE International Conference on Smart Internet of Things (SmartIoT)*, pages 129–134, 2018.

[6] G. Feng, J. Zhang, and S. Shaoyi Liao. A novel method for combining bayesian networks, theoretical analysis, and its applications. *Pattern Recognition*, 47:2057–2069, 2014.

[7] N. Friedman and D. Koller. Being bayesian about network structure: A bayesian

approach to structure discovery in bayesian networks. *Machine Learning*, 50:95–125, 2003.

[8] D. Heckerman. *A Tutorial on Learning with Bayesian Networks*, pages 33–82. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[9] D. Heckerman, A. Mamdani, and M. P. Wellman. Real-world applications of bayesian networks. *Communications of the ACM*, 38:24–26, 1995.

[10] L. Hu, L. Wang, and L. Dong. Quantitative combination of different bayesian networks. *Procedia Engineering*, 15:3526–3530, 2011.

[11] F. V. Jensen. *Bayesian networks and decision graphs*. Springer, 2001.

[12] D. L. Koller and N. Friedman. *Probabilistic graphical models: principles and techniques*. MIT Press, 2009.

[13] S. Mascaro, A. E. Nicholson, and K. B. Korb. Anomaly detection in vessel tracks using bayesian networks. *International Journal of Approximate Reasoning*, 55:84–98, 2014.

[14] P. A. P. Moran. Notes on continuous stochastic phenomena. *Biometrika*, 37:17–23, 1950.

[15] K. P. Murphy. *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012.

[16] J. Pearl, M. Glymour, and N. P. Jewell. *Causal inference in statistics: a primer*. John Wiley Sons, Inc, 2016.

[17] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Pearson, 3rd edition, 2010.

[18] M. Safaei, A. S. Ismail, H. Chizari, M. Driss, W. Boulila, S. Asadi, and M. Safaei. Standalone noise and anomaly detection in wireless sensor networks: A novel time-series and adaptive bayesian-network-based approach. *Software: Practice and Experience*, 50:428–446, 2020.

[19] G. Schwarz. Estimating the Dimension of a Model. *The Annals of Statistics*, 6(2):461 – 464, 1978.

[20] M. Vaniš, Z. Lokaj, and M. Šrotýř. A novel algorithm for merging bayesian networks. *Symmetry*, 15, 2023.

[21] A. Wickramasinghe, S. Muthukumarana, and M. Schaubroeck. Hotspot analysis in commercial buildings using moran's i statistic. Manuscript under review, 2024.

[22] A. Wickramasinghe, S. Muthukumarana, M. Schaubroeck, and S. N. Wanasundara. An anomaly detection method for identifying locations with abnormal behavior of temperature in school buildings. *Scientific Reports*, 13:22930, 2023.

Chapter 6

Conclusion

6.1 Summary of Main Contributions

This thesis has explored innovative approaches to enhance the indoor environment of commercial buildings through advanced data analysis techniques. Each chapter has contributed to a comprehensive understanding of how to detect and address environmental issues, ultimately supporting building owners in creating healthier, more energy-efficient spaces for their tenants.

In Chapter 2, we focused on optimizing thermostat placement by clustering building floors based on variations in indoor environmental conditions. Our findings indicated that time series clustering outperformed traditional methods, providing more accurate groupings. We then introduced a novel method that uses community detection from social network analysis to enhance the time series clustering process. Through the case studies, we demonstrated that this approach significantly improved the identification of temporal patterns, allowing for better groupings based on environmental condition variations.

In Chapter 3, we used hotspot analysis to identify areas in the building that experience excessive heat or cold. We proposed a method to optimize the number of neighbors considered in the hotspot detection process, and determining the correct number of neighbors significantly improved the detection results. Additionally, we explored the reasons for the emergence of these hotspots by analyzing location features and utilizing random forest algorithms to assess feature importance. Our results highlighted that the vacancy of space and the orientation of windows were significant factors contributing to temperature extremes. One limitation identified in the hotspot analysis was that it could not detect hot and cold spots when single outliers were present.

Identifying locations with abnormal environmental behaviors allows building engineers to focus on specific areas and resolve issues. In Chapter 4, we turned our attention to detecting abnormal locations using a scoring method. This novel approach specifically identified collective anomalies resulting from environmental factors. We developed a weighted scoring function to quantify the abnormality of each location, tuning the weights through a grid search algorithm combined with a simulation study. This method allowed us to effectively highlight environmental issues that could impact indoor comfort and energy efficiency.

The final chapter, Chapter 5, presented an enhanced anomaly detection framework using a Bayesian approach. We used the anomaly detection method from Chapter 4 to identify abnormalities at each observation. Then, we developed a Bayesian network to predict the likelihood of abnormal conditions based on various environmental factors. A key innovation in this chapter was the introduction of a novel structure learning method, which improved the prediction accuracy of the Bayesian

network model. By including environmental features from neighboring locations, we further enhanced the model's performance. This improvement demonstrated that neighboring locations have a noticeable effect on the abnormality of the current location.

The overall findings of this research highlight the importance of data-driven approaches in managing the indoor environment of commercial buildings. By using IoT sensor data and advanced analytical techniques, building owners can make informed decisions that greatly improve tenant comfort and energy efficiency. The methodologies proposed in this thesis not only address current challenges but open up new opportunities for future research in this building science field.

In conclusion, this work contributes to the growing field of building science by offering innovative solutions to common environmental issues. The findings not only highlight the effectiveness of using advanced clustering, Hotspot analysis, anomaly detection, and predictive modeling techniques but also demonstrate their practical applications in real-world scenarios. As the demand for sustainable and comfortable indoor environments continues to rise, the approaches developed in this thesis offer valuable tools for building owners and engineers striving to enhance the quality of indoor spaces.

6.2 Future Works

As we look ahead, several key areas present promising directions for further research and improvement in managing indoor environments. In time series clustering, we can enhance our methodology by integrating time series similarity measures and

mean temperature values when forming node connections. This approach will allow us to consider the temporal patterns and the overall temperature of the time series during the clustering process. It will address the limitation of our current method, which sometimes groups time series with similar patterns despite differences in their overall temperature levels.

Next, for hotspot analysis, it's important to recognize that buildings have physical barriers such as walls and doors. When selecting neighboring locations, we must account for these separations. Some neighbors may be in adjacent rooms separated by walls, while others may be in the same room but across the space from a door. Our current analysis did not consider these factors, and addressing them in future work could lead to more accurate hotspot detection and a better understanding of temperature distribution within buildings.

Finally, in our work in anomaly detection, we used a Bayesian static network, which offers a solid foundation. However, transitioning to a Bayesian dynamic network would allow us to predict the likelihood of future abnormal behaviors. This predictive capability would be invaluable for building owners, enabling them to take timely actions that could prevent issues from escalating and ensure a comfortable and efficient indoor environment.

By exploring these avenues, we can further improve our knowledge, understanding, and application of data-driven methods in building science, ultimately contributing to healthier and more energy-efficient indoor spaces.