

# Learning partial correlation graph for multivariate sensor data and detecting sensor communities in smart buildings

Xiang Xie<sup>1,\*</sup>, Manuel Herrera<sup>2</sup>, Tejal Shah<sup>3</sup>, Mohamad Kassem<sup>1</sup> and Philip James<sup>1</sup>

<sup>1</sup>*School of Engineering, Newcastle University, Newcastle upon Tyne, NE1 7RU, United Kingdom*

<sup>2</sup>*Institute for Manufacturing, University of Cambridge, Cambridge, CB3 0FS, United Kingdom*

<sup>3</sup>*School of Computing, Newcastle University, Newcastle upon Tyne, NE4 5TG, United Kingdom*

## Abstract

The storage and processing of massive time series data collected from smart buildings consume considerable computational resources. However, major information redundancy can be found in the smart building data. This paper proposed a partial correlation graph based approach to map the dependencies among sensors and detect the sensor communities in which the sensors are strongly “net” correlated. Specifically, the sparse partial correlation estimation method is used to learn the partial correlation graph. The Louvain algorithm is used to detect the communities of sensors by optimising the graph modularity. The case study demonstrates that the proposed method can identify spare sensors in the detected sensor communities and thus enhance the computational feasibility of smart building applications.

## Keywords

Smart building, computational feasibility, partial correlation graph, community detection

## 1. Introduction

In recent years, the Internet of Things (IoT) has become increasingly popular in the realm of smart buildings, leading to a more livable and sustainable indoor environment. By deploying various IoT sensors and devices, a great amount of data is generated reflecting diverse aspects of buildings’ operations [1]. The heterogeneous data contains valuable information that can be used to facilitate better-informed decision-making [2]. Leveraging machine learning and many big data analytic methods, the sensor data is transformed into information and further mined to extract knowledge. This allows machines to gain better insights and wisdom into the building systems, following the Data-Information-Knowledge-Wisdom (DIKW) pyramid [3]. However, strong spatiotemporal dependencies exist between the multivariate time series generated by multiple sensors. It is unsustainable to treat each sensor as an independent individual without considering the spatial correlation and temporal dynamics among them

---

*Proceedings LDAC2023 – 11th Linked Data in Architecture and Construction, June 15–16, 2023, Matera, Italy*

\*Corresponding author.

✉ xiang.xie@newcastle.ac.uk (X. Xie); amh226@cam.ac.uk (M. Herrera); tejal.shah@newcastle.ac.uk (T. Shah); mohamad.kassem@newcastle.ac.uk (M. Kassem); philip.james@ncl.ac.uk (P. James)

ORCID 0000-0003-4601-9519 (X. Xie); 0000-0001-9662-0017 (M. Herrera); 0000-0001-7060-4211 (T. Shah); 0000-0002-9837-3934 (M. Kassem)



© 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

[4, 5]. Because computations generate 100 megatons of CO<sub>2</sub> emissions per year, accounting for 3% global carbon footprint. For smart buildings, redundant information is computed repeatedly, rarely contributing to new knowledge while generating extra carbon for computing. For example, the window open/close sensors are redundant to a certain extent if indoor and ambient temperatures are monitored for unconditioned spaces.

Graphical models have been proven as useful tools for analysing multivariate time series [6], which specify conditional independence relationships among a collection of random variables. The vertices (nodes) in the graph represent variables, while the directed or undirected edges (links) reflect the causalities or dependencies between variables. The dynamic relationships over time among the variables help to determine and explain the causation or association mechanisms of the underlying systems. The application of graphical time series models can be seen in a wide range of areas, such as financial market analysis [7] and brain interactivity analysis [8]. In particular, for high-dimensional stationary multivariate Gaussian time series, the Gaussian graphical model, an undirected graph of partial correlation coefficients, allows for sparse modelling of the underlying association structure amongst observed variables. This is an efficient tool for modelling complex systems of myriad variables by building them using a smaller proportion of variables.

With the ubiquitous IoT sensors in smart buildings, buildings are modelled as systems with humans in the loop. This requires huge computation and storage resources as well as advanced software that implements innovative algorithms for analysing high-dimensional sensor data. In this paper, the Gaussian graphical model, also known as the partial correlation graph, is used to model the relationship underlying the multivariate time series data collected from smart buildings. Based on the learned partial correlation graph, the communities of sensors could be formed, in which a few sensors would represent the overall pattern of a community with similar spatiotemporal features. It leads to a computationally feasible strategy to model the diverse spatiotemporal processes for buildings in a dimension-reduced manner.

## **2. Literature review**

### **2.1. Feature selection for multivariate time series**

Data analysis is a computationally intensive process, particularly when dealing with large datasets containing a significant number of variables. In this context, these “variables” are also called “features”. The computational complexity, in terms of space or time, which measures the total amount of either time or memory taken by an algorithm to run, often increases dramatically with the feature dimensionality [9]. To avoid the so-called “curse of dimensionality”, feature extraction and feature selection techniques are proposed respectively. Feature extraction methods aim to project high-dimensional data into low-dimensional subspace. Principal Component Analysis (PCA), multidimensional scaling (MDS) and Isometric Mapping (ISOMAP) are typical feature extraction methods [10]. However, the new set of features built from the original feature set lacks clear physical interpretations. On the other hand, feature selection methods remove irrelevant and redundant dimensions from the raw dataset with minimal loss of information. By filtering out unrepresentative features, the process of feature selection helps to extract meaningful insights from the original dataset, reduce the complexity of data analysis, release

the computational burden, avoid the overfitting problems, and improve the generalisation and interpretability capacity of the corresponding machine learning approaches [11].

In general, feature selection methods can be classified into three categories, which are filter, wrapper, and embedded methods [12]. Filter methods measure and rank feature importance based on certain feature evaluation criteria quantifying their statistical characteristics, such as Pearson correlation coefficient, Fisher score [13], and mutual information [14]. Independent of the machine learning approach adopted for data analysis, the filter methods are computationally efficient by simply filtering out lowly ranked features. As the alternative, wrapper methods search for a subset of features that optimises the performance of the predefined machine learning approach. Typical search strategies include sequential search, hill-climbing search, best-first search, branch-and-bound search, and heuristic genetic algorithms. However, for very large  $d$  features within  $2^d$  search space, the wrapper methods become computationally infeasible [15]. Embedded methods provide a trade-off between filter and wrapper methods. Inheriting the merits of the wrapper and filter methods, embedded methods introduce sparse regularisation terms to the optimisation objective of a machine learning model to penalise complex models and reduce the dimensionality of the input features.

Many existing feature selection methods are based on an illusory assumption that features are independent of each other while ignoring the inherent feature structures [12]. In the case of smart buildings, the inherent feature structures mainly come from the spatial proximity and functional association of sensors inside a building. Incorporating such spatial structure can help select more important features for smart building applications. In this paper, a partial correlation graph of multivariate time series data from smart buildings is learned, where undirected edges with weights indicate the pairwise dependencies between features represented by vertices.

## 2.2. Partial correlation graph of time series

Partial correlation graphs have long been used to explicitly map the dependencies between various stationary variables in multiple domains, such as river stage forecasting [16]. By identifying the nonzero entities in the partial correlation matrix (equivalent or related to inverse covariance matrix, concentration matrix and precision matrix), the topology of the partial correlation graph can be reconstructed and the conditional dependencies among the observed variables can be elucidated [17]. The partial correlation coefficients, ranging from  $-1$  to  $1$ , encode the dependence between two variables after eliminating the influence of all the remaining variables. When determining the conditional dependency between two variables of interest  $A$  and  $B$ , the conventional Pearson correlation coefficient may give misleading results when another confounding variable  $C$  has the so-called common cause or common effect on both  $A$  and  $B$ . By computing the partial correlation coefficients, the spurious correlation can be eliminated where only the “unbiased relationship” of  $A$  and  $B$  remains.

Estimating partial correlation coefficients comes down to calculating the inverse of the covariance matrix. Studies have shown that partial correlations are proportional to not only the multiple linear regression coefficients but also the off-diagonal entries of the inverse covariance matrix [17]. Especially, in cases where the number of variables is greater than the number of observations, the rank of the covariance matrix is equal to the number of observations, and methods like generalised inverse or pseudo-inverse need to be adopted to tackle this ill-

posed inverse problem. However, these methods often fail to provide accurate and sparse interpretable solutions. To address this issue, regularisation techniques are explored to impose sparse constraints on the inverse covariance matrix. The overall sparsity assumption of the partial correlation matrix is reasonable and taken for granted for many real-world problems. L1-norm regularisation and elastic net regularisation, which is the combination of L1- and L2-norm regularisation, have been used to extract sparse nonzero partial correlation coefficients, with the elastic net regularisation showing stronger robustness for the estimation.

In the graphical model for the multivariate time series, the number of possible edges between vertices grows quadratically. Fortunately, there usually exists a corresponding sparse graph such that the edges directly linked to each vertice are few [18]. By taking this sparsity into account, it is possible to develop graph models with good generalisation and predictive capability from far fewer samples. This is the desired character for smart building applications, where the analysis can be conducted and the decisions can be made relying on fewer data instead of all.

### 3. Methodology

#### 3.1. Estimation of partial correlation graph

Considering a weighted undirected graph  $\mathcal{G} = \{V, \mathcal{E}\}$  with vertices indexed by  $V = \{1, 2, \dots, p\}$  and a corresponding set of undirected edges  $\mathcal{E} \subseteq \mathbb{R}^{p \times p}$ , let  $\mathbf{x}_i \in \mathbb{R}^n$  denote the time series observed from the vertice  $i$  and the edge from vertice  $j$  to vertice  $i$  has a weight  $w_{ij}$  ( $i, j \in [1, p]$ ). The weighted undirected graph becomes the partial correlation graph when the respective weights  $w_{ij}$  are assigned the partial correlation coefficients between these variables  $i$  and  $j$ . More precisely, the edge between vertice  $i$  and  $j$  exists if and only if  $\mathbf{x}_i$  and  $\mathbf{x}_j$  are conditionally independent given the remaining  $p - 2$  variables. Suppose the multivariate Gaussian time series data  $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_p]$  has positive-definite covariance matrix  $\Sigma = \mathbb{E}(\mathbf{X}^T \mathbf{X})$  and inverse covariance matrix  $\mathbf{P} = \Sigma^{-1}$ . The  $(i, j)$  element in  $\mathbf{P}$ , represented by  $P_{ij}$ , is nonzero when  $\mathbf{x}_i$  and  $\mathbf{x}_j$  observed by vertice  $i$  and  $j$  are conditionally dependent straightforward.

Acquiring the inverse of the covariance matrix becomes an ill-posed problem when the number of observations  $n$  is less than the number of variables  $p$ . In this case, the covariance matrix  $\Sigma$  is singular and non-invertible. A computationally efficient sparse partial correlation matrix estimator, Sparse PARTial Correlation Estimation (SPACE) method, is proposed in [17]. SPACE method alternates between solving for the partial correlation matrix  $\mathbf{P}$  and for the diagonal of the precision matrix  $\Theta$  with the objective function of:

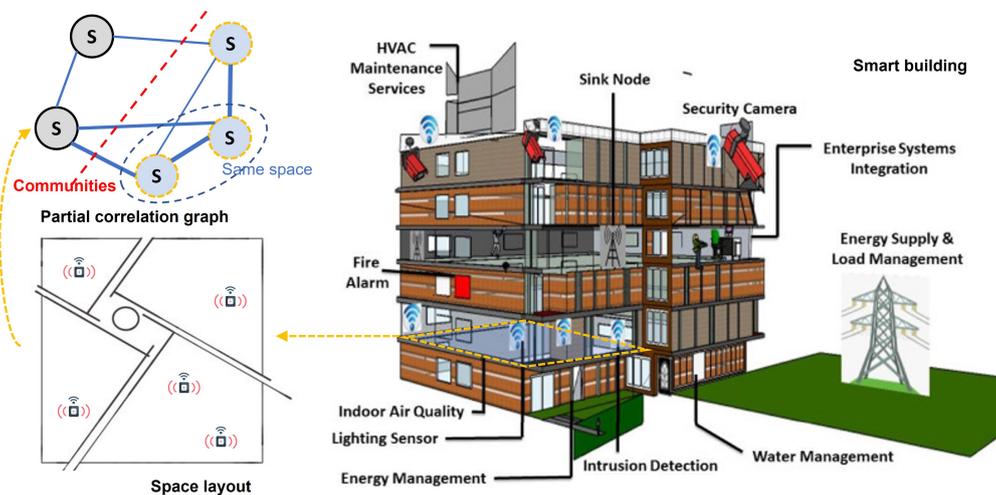
$$L(\mathbf{P}, \Theta) = \frac{1}{2} \sum_{i=1}^p \|\mathbf{x}_i - \sum_{j \neq i} P_{ij} \sqrt{\frac{\Theta_{jj}}{\Theta_{ii}}} \mathbf{x}_j\|_2^2 + \lambda \sum_{1 \leq i < j \leq p} |P_{ij}| \quad (1)$$

$$P_{ij} = -\frac{\Theta_{ij}}{\sqrt{\Theta_{ii}\Theta_{jj}}} \quad (2)$$

To solve this, a Least Absolute Shrinkage and Selection Operator (LASSO) problem is formulated. Typically, the optimisation problem converges after 2 to 3 iterations.

### 3.2. Inference of sensor communities in smart buildings

By 2026, the number of sensors deployed in smart buildings will exceed one billion, according to a study from Juniper Research. With this growth, more smart building applications will emerge to gain insights and make better-informed decisions using the vast amount of building data collected [19]. Assuming  $p$  sensors are deployed in a specific area of a building, the data accumulates with time according to the sensor sampling rate or the frequency of changes in sensor data values. However, the time series data from these  $p$  sensors are not independent of each other due to the spatial correlations. To reflect the dependencies among these sensor readings, a partial correlation graph of the multivariate time series collected from  $p$  sensors is learned, as illustrated in Figure 1. These sensors can be of the same type, such as carbon dioxide (CO<sub>2</sub>) sensors within the same space. Typically, they provide similar readings under the well-mixed assumption when the sensor sampling interval is greater than 10 minutes. Strong dependencies also exist between data generated by different types of sensors. For instance, the volatile organic compound (VOC) concentration is highly correlated with the CO<sub>2</sub> concentration because CO<sub>2</sub> serves as a common surrogate indicator for indoor air quality.



**Figure 1:** Learning of partial correlation graph for time series data from sensors deployed in smart buildings (adapted from [20]).

To identify intensively tied sensors with considerable data redundancy, the community detection approach is used to cluster dependent sensors based on the estimated partial correlation graph [21]. A community refers to a group of vertices that are closely connected to each other but less connected to the vertices outside the group. Detected communities contain groups of sensors that generate highly correlated time series. Louvain algorithm is a classical method to extract communities from networks, which optimises the defined modularity of communities indicating the density of (weighted) edges within communities with respect to edges outside communities [22]. The choice of Louvain algorithm is due to its properties of computational efficiency and scalability that make it suitable even for large-size networks. The modularity score is formulated as:

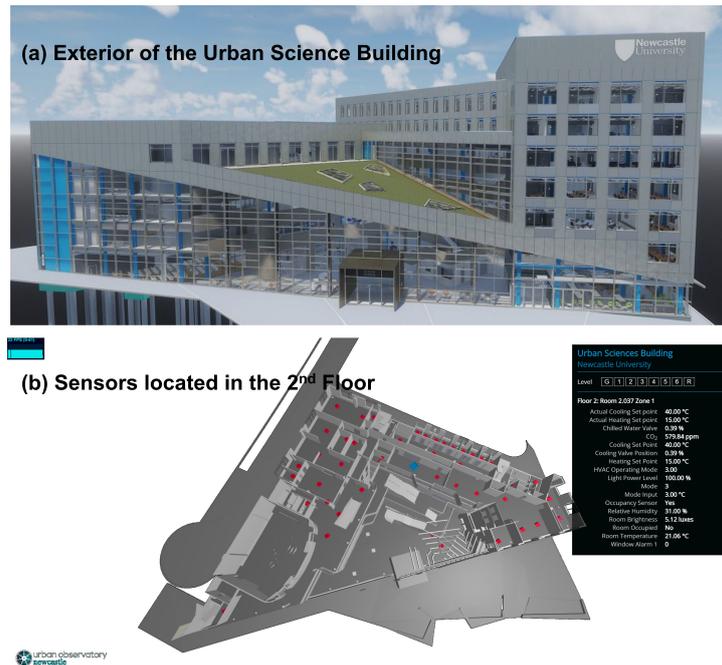
$$Q = \frac{1}{2m} \sum_{i,j} \left[ A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j) \quad (3)$$

, where  $m$  is the sum of all edge weights in the undirected graph,  $A_{ij}$  denotes the weight of the edge between vertex  $i$  and  $j$ ,  $k_i$  and  $k_j$ , represents the sum of weights connecting vertex  $i$  and  $j$ ,  $c_i$  and  $c_j$  are the communities of the vertices, and  $\delta(\cdot)$  is the Kronecker delta function. The Louvain algorithm alternatively conducts modularity optimization and community aggregation. This process is repeated until no further increase in modularity is possible and the detected communities stabilise, leading to an optimal partition of the partial correlation graph into communities of sensors.

#### 4. Case study

To validate the proposed methodology, the urban science building (USB) of Newcastle University, a part of the Newcastle Helix site within Newcastle city centre, is used as the testbed. With over 4,000 sensors, computing technology is embedded throughout the building's structure, making it one of the most intensively monitored buildings in the UK. Figure 2 (a) shows the exterior of the urban science building and Figure 2 (b) shows the layout of its second floor and the included spaces (red cubes indicating the centre of spaces) where sensors are deployed.

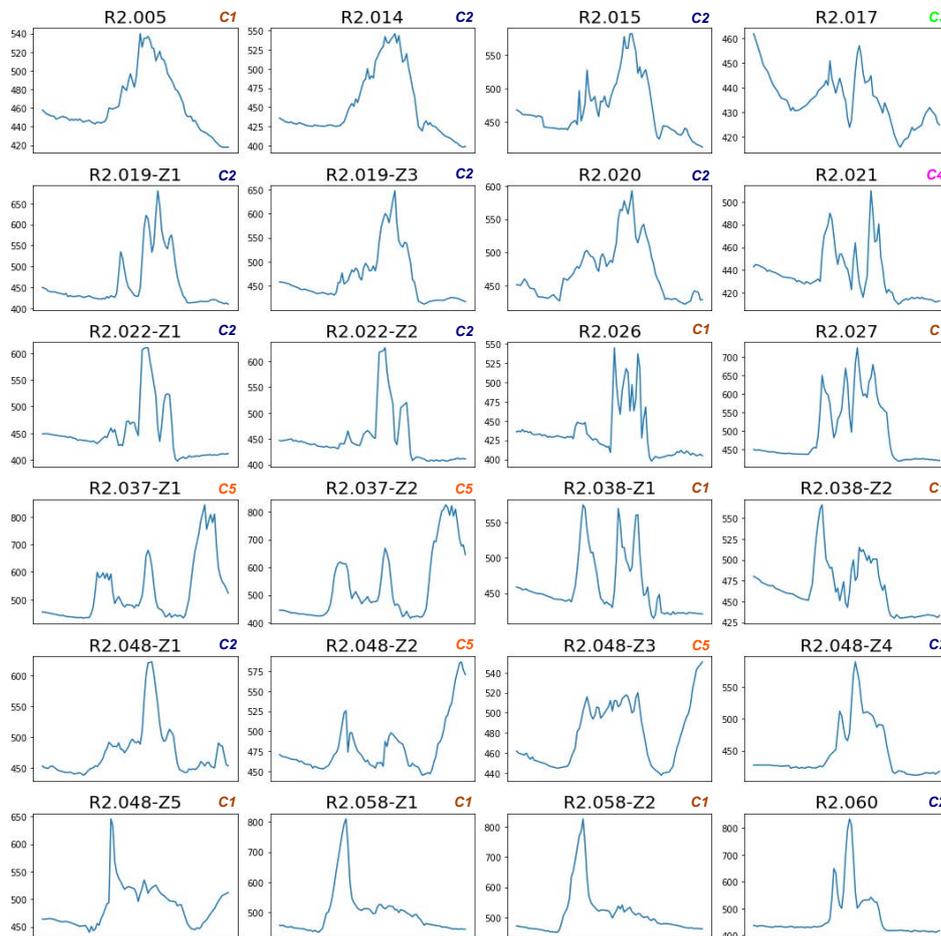
In this case study, the carbon dioxide (CO<sub>2</sub>) concentration sensors on the building's second



**Figure 2:** Urban science building (a), the layout of its 2nd floor and spaces where sensors are deployed (b).

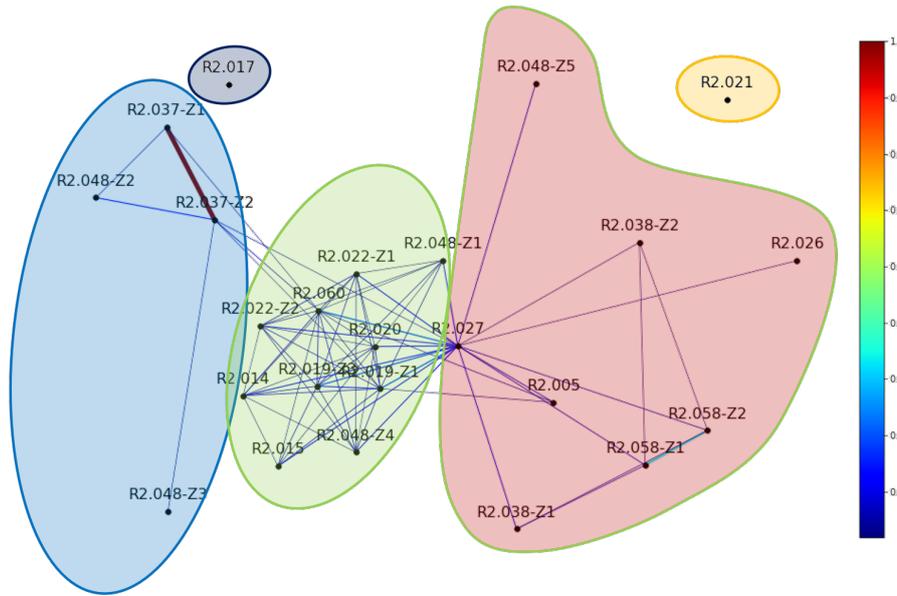
floor are used to demonstrate the information redundancy residing in the collected sensor data. Figure 3 presents the CO<sub>2</sub> concentrations measured by 24 sensors, in the unit of ppm (parts per million). The 24-hour data was collected on Feb 15, 2023, which is a Wednesday. The sensor data is cleaned and preprocessed to impute the missing sensor values and resampled to 15-minute intervals. The sensors are labelled with the respective room names they are located in (i.e., 'R'+floor\_number+'.'+room\_number). Note that, more than one sensor may be deployed in one open space, in which the sensors inside are specified by the additional zone number (i.e., 'R'+floor\_number+'.'+room\_number+'-Z'+zone\_number).

The partial correlation graph is learned based on the collected daily CO<sub>2</sub> data. Figure 4 visualises the partial correlation graph of the sensor time series, in which the edges indicate the dependencies between the CO<sub>2</sub> concentrations measured at proximate locations. The stronger the dependency is, the thicker the corresponding edge is. The highest partial correlation coefficient emerges between the CO<sub>2</sub> concentrations measured in R2.037-Z1 and R2.037-Z2, followed by the CO<sub>2</sub> concentrations measured in R2.058-Z1 and R2.058-Z2. The Louvain



**Figure 3:** Carbon dioxide concentrations measured by 24 sensors on the 2nd floor of the USB.

algorithm is used to detect the communities of sensors, the results of which are given in Table 1. Besides, the community numbers are shown in the top right corner of each time series in Figure 3. The detected 5 communities make sense to a certain extent. For example, rooms R2.048 and R2.037 are open areas next to each other and therefore share similar CO<sub>2</sub> concentration readings. Same for the room R2.060 and the neighbouring zone R2.048-Z4. An interesting phenomenon can be observed. Rooms like R.027 and R.060 are clustered into different groups although showing a correlation in-between. This is because they are clustered based on their proximity to the centroid of that cluster, rather than the "distance" between them.



**Figure 4:** Partial correlation graph between the CO<sub>2</sub> concentrations measured by 24 sensors.

**Table 1**  
Communities of CO<sub>2</sub> sensors based on the learned partial correlation graph

Community number	Included sensors
1	R2.005, R2.026, R2.027, R2.038-Z1, R2.038-Z2, R2.048-Z5, R2.058-Z1, R2.058-Z2
2	R2.014, R2.015, R2.019-Z1, R2.019-Z3, R2.020, R2.022-Z1, R2.022-Z2, R2.048-Z1, R2.048-Z4, R2.060
3/4	R2.017/R2.021
5	R2.048-Z2, R2.048-Z3, R2.037-Z1, R2.037-Z2

The detected communities of sensors can help to identify the spare sensors deployed in near spaces. In practice, the deployment of IoT sensors in smart buildings largely follows intuition. In the case of USB, at least one CO<sub>2</sub> sensor is deployed in each occupied space, and for large open spaces, one CO<sub>2</sub> sensor is responsible for monitoring an individual zone. The learned partial correlation graph and the detected communities of sensors indicate that some sensors

located in proximate locations basically provide the same piece of information. In such cases, virtual sensors can be defined by fusing multivariate time series data from duplicated sensors, which reduces the size of sensor data to be stored and processed.

## 5. Discussion

To deal with the massive time series data from smart buildings, a partial correlation graph-based methodology is proposed in this paper. The partial correlation graph reflects the conditional independence relationships among the multivariate time series data generated by multiple sensors distributed in different spaces. However, the conditional independence relationship does not necessarily correspond to the causality. This is why some distanced sensors can still appear in the same community. In the case of USB, some sensors deployed near the corridors emerge in the same community, probably because the CO<sub>2</sub> concentrations are affected by similar occupants' activities. To bring the spatial information into the equation, the spatial adjacency graph, which describes the spatial relationships among sensors, can be converted from the semantic models using ontology such as the Building Topology Ontology (BOT) [23]. By integrating the partial correlation graph and the spatial adjacency graph, the detected communities would be more physically interpretable. Furthermore, because the dependencies between sensor data rely on human activities as well, a sliding window approach will be applied to detect the changes in the sensor communities over time. The dynamic sensor communities with weekday/weekend, season, and year patterns are expected.

Admittedly, the building semantic model can be enriched using the spatiotemporal features extracted from the sensor data. But we reserve the opinion that the semantic model is not the best repository for such spatiotemporal-wise knowledge considering the uncertainties and more importantly dynamics. The learned graph in the case study only reflects the spatiotemporal pattern of a specific day, and we expect weekly, monthly, seasonal or annual changes in the acquired patterns. The semantic model, as it is, is not suitable for this type of dynamic information, unless the objects and relations can be timestamped. Alternatively, these periodic spatiotemporal patterns can be encoded in machine learning models. Further studies are needed to tackle this challenge.

## 6. Conclusion

The emergence of the smart building concept brings great opportunities and challenges. One of the main challenges is the massive data generated every minute and every second. For the multivariate time series data from smart buildings, the partial correlation graph is learned using the sparse partial correlation estimation method, which maps the dependencies among the sensors in the smart building. Leveraging the learned partial correlation graph, the sensors are clustered into different communities, in which the sensors are strongly tied. The urban science building of Newcastle University is used as the case study. The results demonstrate that the proposed methodology can uniquely identify spare sensors in a detected community of sensors that barely provide extra information to smart building applications. It leads to a computationally feasible approach to reduce the volume of sensor data with minimum information loss.

## References

- [1] A. P. Plageras, K. E. Psannis, C. Stergiou, H. Wang, B. B. Gupta, Efficient IoT-based sensor big data collection–processing and analysis in smart buildings, *Future Generation Computer Systems* 82 (2018) 349–357.
- [2] B. Qolomany, A. Al-Fuqaha, A. Gupta, D. Benhaddou, S. Alwajidi, J. Qadir, A. C. Fong, Leveraging machine learning and big data for smart buildings: A comprehensive survey, *IEEE Access* 7 (2019) 90316–90356.
- [3] M. Ardolino, M. Rapaccini, N. Saccani, P. Gaiardelli, G. Crespi, C. Ruggeri, The role of digital technologies for the service transformation of industrial companies, *International Journal of Production Research* 56 (2018) 2116–2132.
- [4] Royal Society, Digital technology and the planet: Harnessing computing to achieve net zero, 2020.
- [5] L. Lannelongue, J. Grealey, M. Inouye, Green algorithms: quantifying the carbon footprint of computation, *Advanced science* 8 (2021) 2100707.
- [6] J. K. Tugnait, On sparse high-dimensional graphical model learning for dependent time series, *Signal Processing* 197 (2022) 108539.
- [7] T. Millington, M. Niranjana, Partial correlation financial networks, *Applied Network Science* 5 (2020) 1–19.
- [8] A. Zhang, J. Fang, F. Liang, V. D. Calhoun, Y.-P. Wang, Aberrant brain connectivity in schizophrenia detected via a fast gaussian graphical model, *IEEE journal of biomedical and health informatics* 23 (2018) 1479–1489.
- [9] O. Y. Al-Jarrah, P. D. Yoo, S. Muhaidat, G. K. Karagiannidis, K. Taha, Efficient machine learning for big data: A review, *Big Data Research* 2 (2015) 87–93.
- [10] P. Ray, S. S. Reddy, T. Banerjee, Various dimension reduction techniques for high dimensional data analysis: a review, *Artificial Intelligence Review* 54 (2021) 3473–3515.
- [11] B. Chizi, O. Maimon, Dimension reduction and feature selection, *Data mining and knowledge discovery handbook* (2010) 83–100.
- [12] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, H. Liu, Feature selection: A data perspective, *ACM computing surveys (CSUR)* 50 (2017) 1–45.
- [13] L. Sun, T. Wang, W. Ding, J. Xu, Y. Lin, Feature selection using fisher score and multilabel neighborhood rough sets for multilabel classification, *Information Sciences* 578 (2021) 887–912.
- [14] M. Bennisar, Y. Hicks, R. Setchi, Feature selection using joint mutual information maximization, *Expert Systems with Applications* 42 (2015) 8520–8532.
- [15] M. Ghaemi, M.-R. Feizi-Derakhshi, Feature selection using forest optimization algorithm, *Pattern Recognition* 60 (2016) 121–129.
- [16] S. R. Venna, S. Katragadda, V. Raghavan, R. Gottumukkala, River stage forecasting using enhanced partial correlation graph, *Water Resources Management* 35 (2021) 4111–4126.
- [17] J. Peng, P. Wang, N. Zhou, J. Zhu, Partial correlation estimation by joint sparse regression models, *Journal of the American Statistical Association* 104 (2009) 735–746.
- [18] A. Venkitaraman, D. Zachariah, Learning sparse graphs for prediction of multivariate data processes, *IEEE Signal Processing Letters* 26 (2019) 495–499.
- [19] X. Xie, Q. Lu, M. Herrera, Q. Yu, A. K. Parlikad, J. M. Schooling, Does historical data still

- count? exploring the applicability of smart building applications in the post-pandemic period, *Sustainable Cities and Society* 69 (2021) 102804.
- [20] M. U. Younus, S. ul Islam, I. Ali, S. Khan, M. K. Khan, A survey on software defined networking enabled smart buildings: Architecture, challenges and use cases, *Journal of Network and Computer Applications* 137 (2019) 62–77.
- [21] B. S. Khan, M. A. Niazi, Network community detection: A review and visual survey, *arXiv preprint arXiv:1708.00977* (2017).
- [22] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, E. Lefebvre, Fast unfolding of communities in large networks, *Journal of statistical mechanics: theory and experiment* 2008 (2008) P10008.
- [23] M. H. Rasmussen, M. Lefrançois, G. F. Schneider, P. Pauwels, BOT: The building topology ontology of the w3c linked building data group, *Semantic Web* 12 (2021) 143–161.