

Visualization Term Project

20141087 Ryeongyang Kim

20161206 Jaewoong Lee

November 25, 2019

Contents

1	Introduction	3
2	Materials	3
2.1	Building Layout	3
2.1.1	Main Layout	3
2.1.2	Energy Zone Layout	3
2.1.3	Prox Zone Layout	3
3	Methods	3
3.1	Python Packages	3
3.1.1	Scikit-learn: Machine Learning in Python	3
3.1.2	Matplotlib	4
3.1.3	Pandas	4
3.1.4	SciPy	4
3.2	TSNE	4
4	Results	4
4.1	Question 1	4
4.1.1	General Information of prox Data	4
4.1.2	Workflow	5
4.1.3	Movement Direction and Distance	5
4.2	Question 2	7
4.2.1	General Information of Building Data	7
4.2.2	Workflow	7
4.2.3	Find Abnormality in General Data	7
4.3	Question 3	7
4.3.1	General Information of Hazium Concentration	7
4.3.2	Workflow	9
4.4	Question 4	9
4.4.1	General Information	9
4.4.2	Workflow	9
5	Discussion	9

List of Tables

1	Basic Statistics Data within Movement Distance	5
---	--	---

List of Figures

1	Main Layout of the building	3
2	Energy Zone of the Building	3
3	Prox zone of the Building	4
4	Distribution of Movement Distance	4
5	Workflow for Question 1	5
6	Movement Direction/Distance on First Floor	5
7	Movement Direction/Distance on Second Floor	6
8	Movement Direction/Distance on Third Floor	6
9	TSNE Plot for General Building Data	7
10	Workflow for Question 2	7
11	Abnormality Founded by Algorithms	8
12	Abnormality by Timeline	8
13	Hazium Data from Different Data Sources	8
14	Workflow for Question 3	8
15	Workflow for Question 4	9

1 Introduction

In this term project, we have to answer several question with virtual building data.

2 Materials

2.1 Building Layout

To analyzing movement data, we should find corresponding coordinate with zone data. To find matching coordinate, we calculate the approximate center of all zones, and consider the approximate center coordinate as representative of its zone.

2.1.1 Main Layout

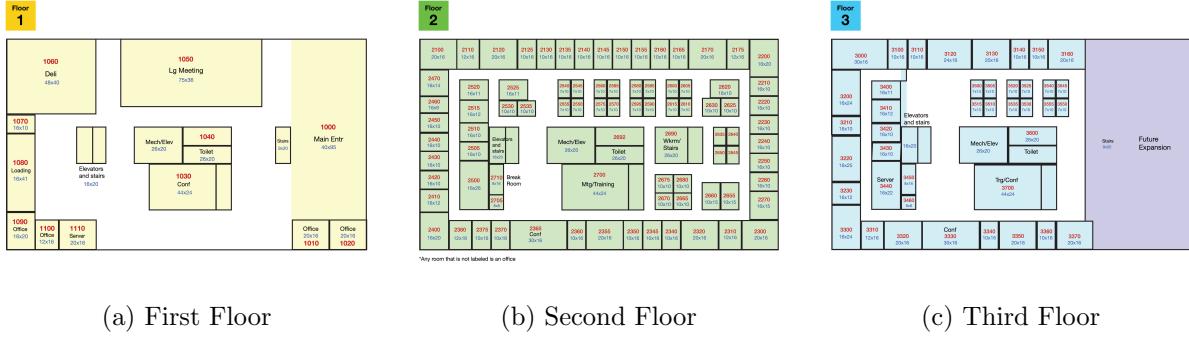


Figure 1: Main Layout of the building

The main layout of this building is as figure 1.

2.1.2 Energy Zone Layout

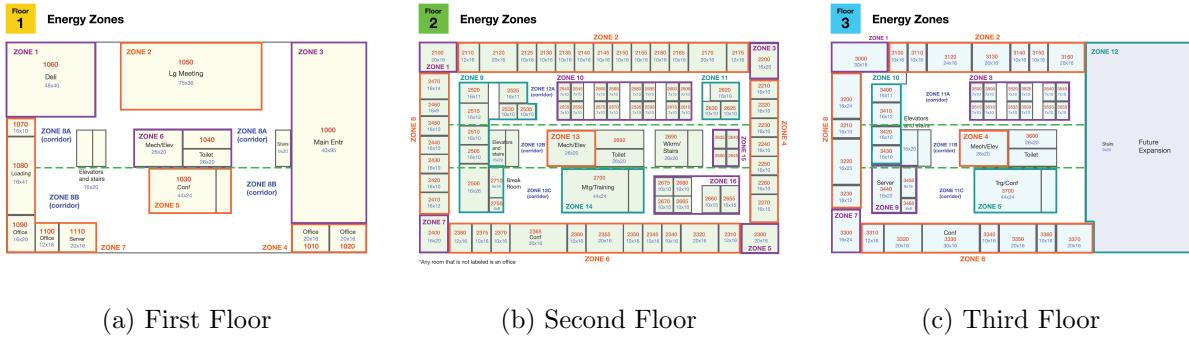


Figure 2: Energy Zone of the Building

The energy zone of this building is as figure 2.

2.1.3 Prox Zone Layout

The prox zone of this building is as figure 3.

3 Methods

3.1 Python Packages

To analyze data, we used Python programming language. Also, we adopt many Python modules as hereinafter.

3.1.1 Scikit-learn: Machine Learning in Python

Scikit-learn is a Python module integrating a wide range of state-of-the-art machine learning algorithms for medium-scale supervised and unsupervised problems [1].

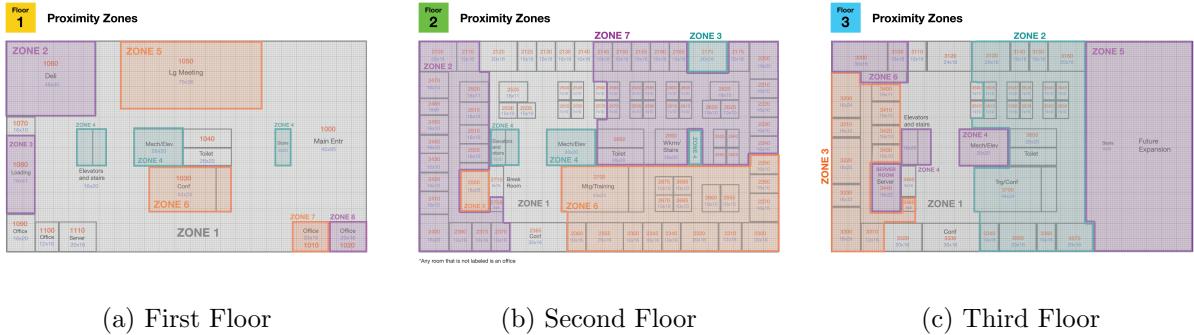


Figure 3: Prox zone of the Building

3.1.2 Matplotlib

Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms [2].

3.1.3 Pandas

Pandas is a Python library of rich data structures and tools for working with structured data sets common to statistic, finance, social sciences, and many other fields [3].

3.1.4 SciPy

SciPy is a Python-based ecosystem of open-source software for mathematics, science, and engineering [4].

3.2 TSNE

T-distributed Stochastic Neighbor Embedding (TSNE) is a machine learning algorithm for visualization high-dimensional data in a low-dimensional space [5].

4 Results

4.1 What are the typical patterns in the prox card data? What does a typical day look like for GAStech employees?

4.1.1 General Information of prox Data

First of all, we drew the distribution of movement distance as figure 4. Also, the basic statistics values, such as minimum, maximum, and average, of movement distance is in table 1.

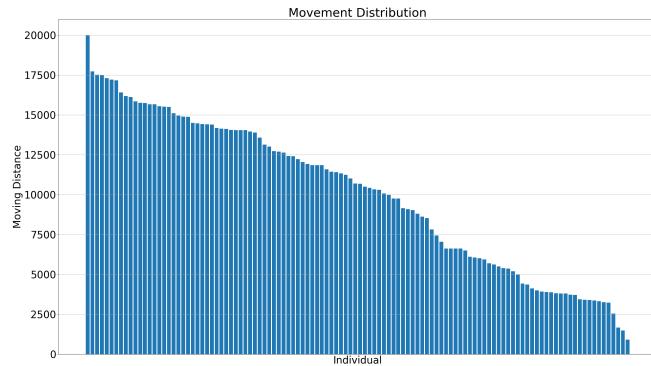


Figure 4: Distribution of Movement Distance

Table 1: Basic Statistics Data within Movement Distance

Item	Minimum	Maximum	Average	q1	Median	q3	Standard Deviation
Value	902.44	19999.38	10083.95	5642.54	10688.57	14134.16	4750.46

4.1.2 Workflow

With the general information of prox data, we have decided our workflow for question 1 as figure 5.

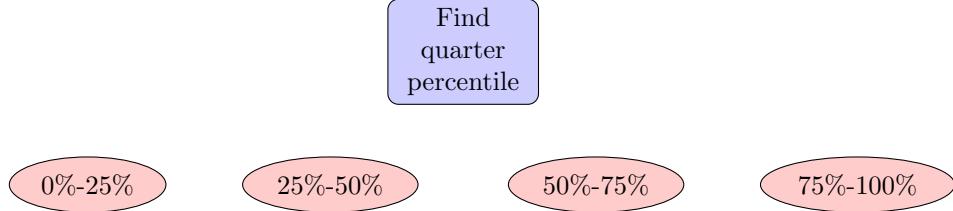


Figure 5: Workflow for Question 1

4.1.3 Movement Direction and Distance

We drew the plot about movement direction and distance with each sub-group as figures 6, 7, and 8. Note that the darkness of arrow is proportioned with number of duplicates.

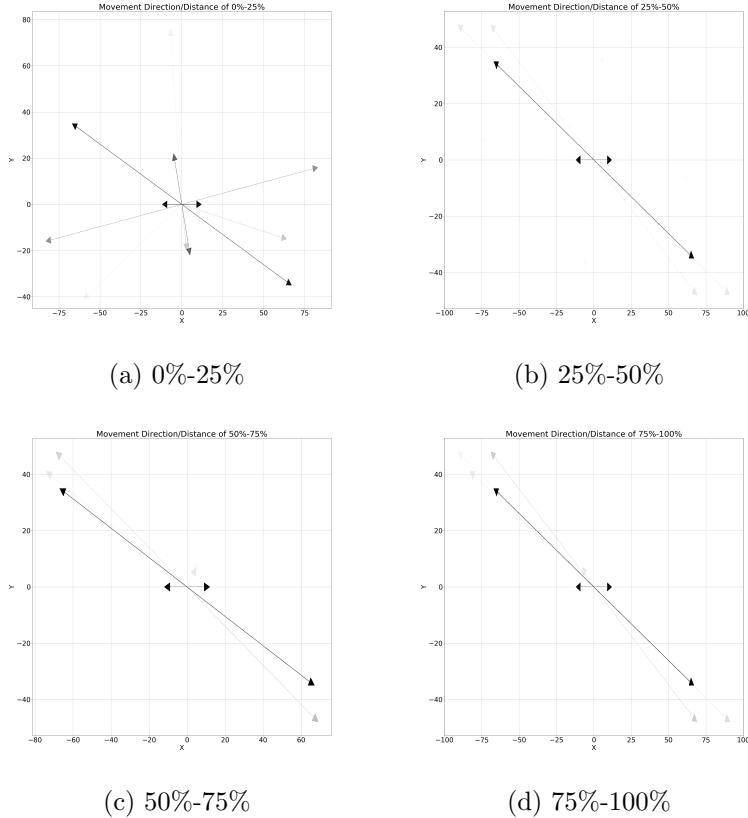
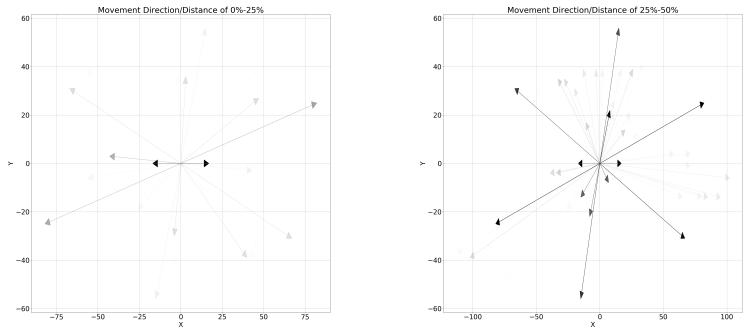


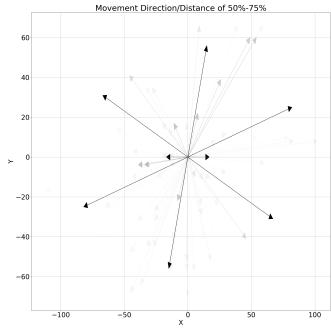
Figure 6: Movement Direction/Distance on First Floor

The movement direction and distance on the first floor is shows as figure 6. In figure 6-(b, c, d), you can see two arrows: one is left-upward arrow, the other is right-downward arrow.

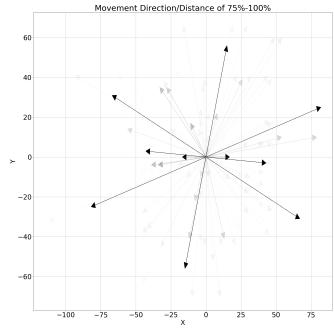


(a) 0%-25%

(b) 25%-50%

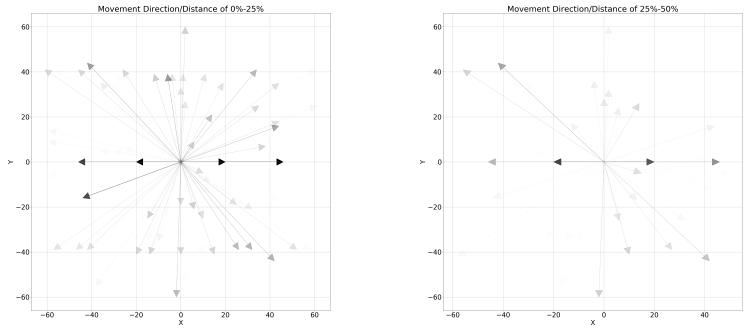


(c) 50%-75%

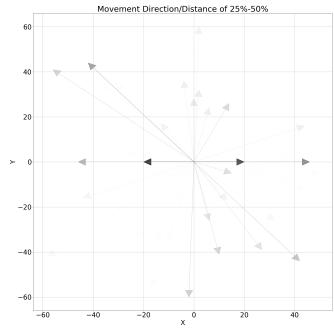


(d) 75%-100%

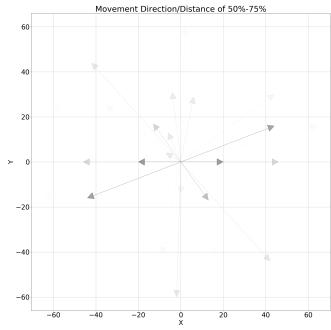
Figure 7: Movement Direction/Distance on Second Floor



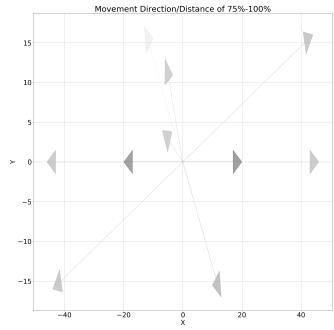
(a) 0%-25%



(b) 25%-50%



(c) 50%-75%



(d) 75%-100%

Figure 8: Movement Direction/Distance on Third Floor

4.2 Describe up to five of the most interesting patterns that appear in the building data. Describe what is notable about the pattern and explain its possible significance.

4.2.1 General Information of Building Data

First of all, we drew the distribution of the building data using TSNE technique. The TSNE plot of general building data is shows as figure 9.

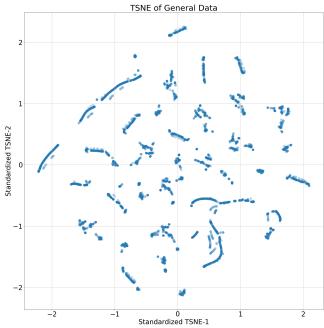


Figure 9: TSNE Plot for General Building Data

4.2.2 Workflow

Figure 10: Workflow for Question 2

4.2.3 Find Abnormality in General Data

To find patterns which appear in the building data, we should find that normality/abnormality in the building data. However, there are over 400 columns in the general building data; therefore, it is almost impossible to find abnormality column-by-column by human. Hence, we used these four algorithms which are included in scikit-learn: *EllipticEnvelope* [6], *OneClassSVM*, *IsolationForest* [7, 8], and *LocalOutlierFactor* [9].

Abnormality founded is shows as figure 11. Note that some data were considered as abnormal in multiple algorithms; however, no data were considered as abnormal in all algorithms. Moreover, with the data in figure 11, we can display the timeline of abnormality as figure 12.

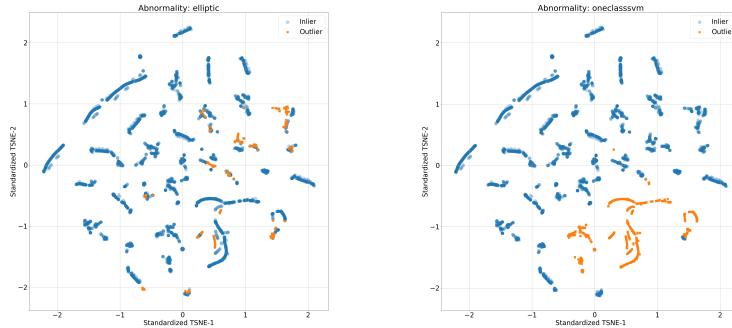
In the figure 12-(a), we can know that which algorithm consider specific time as abnormal events (yellow marked is abnormal); and, in the figure 12-(b), we can realize that how many algorithms consider specific time as abnormal events.

4.3 Describe up to five notable anomalies or unusual events you see in the data. Prioritize those issue that are most likely to represent a danger or a serious issue for building operations.

4.3.1 General Information of Hazium Concentration

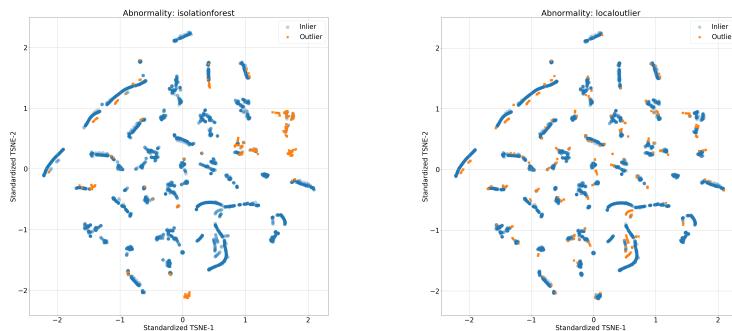
In the question 3, we need to find a danger or a serious issue for building data. Hence, we suppose that a danger will be related with Hazium concentration.

In the figure 13, we can see Hazium concentration of many sources.



(a) *EllipticEnvelope*

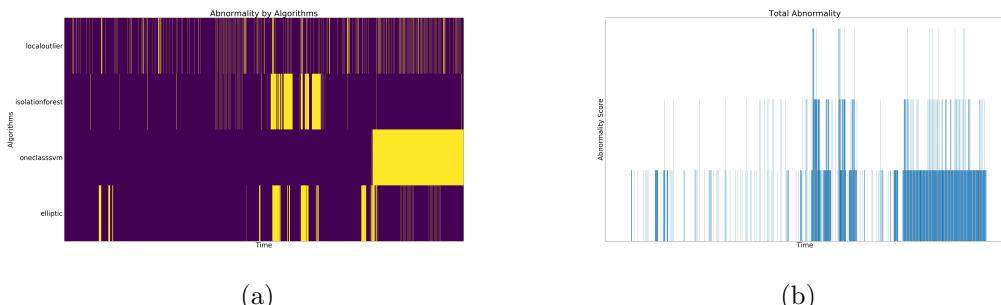
(b) *OneClassSVM*



(c) *IsolationForest*

(d) *LocalOutlierFactor*

Figure 11: Abnormality Founded by Algorithms



(a)

(b)

Figure 12: Abnormality by Timeline

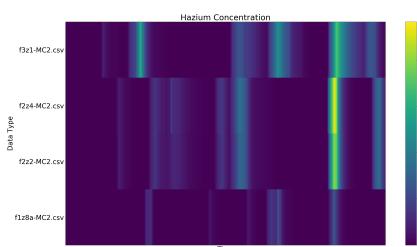


Figure 13: Hazium Data from Different Data Sources

Figure 14: Workflow for Question 3

4.3.2 Workflow

4.4 Describe up to three observed relationships between the proximity card data and building data elements. If you find a causal relationship, describe your discovered cause and effect, the evidence you found the support it, and your level of confidence in your assessment of the relationship.

4.4.1 General Information

4.4.2 Workflow

Figure 15: Workflow for Question 4

5 Discussion

References

- [1] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, *et al.*, “Scikit-learn: Machine learning in python,” *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.
- [2] J. D. Hunter, “Matplotlib: A 2d graphics environment,” *Computing in science & engineering*, vol. 9, no. 3, p. 90, 2007.
- [3] W. McKinney, “pandas: a foundational python library for data analysis and statistics,” *Python for High Performance and Scientific Computing*, vol. 14, 2011.
- [4] E. Jones, T. Oliphant, P. Peterson, *et al.*, “Scipy: Open source scientific tools for python,” 2001.
- [5] L. v. d. Maaten and G. Hinton, “Visualizing data using t-sne,” *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.
- [6] P. J. Rousseeuw and K. V. Driessen, “A fast algorithm for the minimum covariance determinant estimator,” *Technometrics*, vol. 41, no. 3, pp. 212–223, 1999.
- [7] F. T. Liu, K. M. Ting, and Z.-H. Zhou, “Isolation forest,” in *2008 Eighth IEEE International Conference on Data Mining*, pp. 413–422, IEEE, 2008.
- [8] F. T. Liu, K. M. Ting, and Z.-H. Zhou, “Isolation-based anomaly detection,” *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 6, no. 1, p. 3, 2012.
- [9] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, “Lof: identifying density-based local outliers,” in *ACM sigmod record*, vol. 29, pp. 93–104, ACM, 2000.