Decomposition-based Data Augmentation for Time-series Building Load Data

Yang Deng, Rui Liang, Dan Wang, Ao Li and Fu Xiao

The Hong Kong Polytechnic University

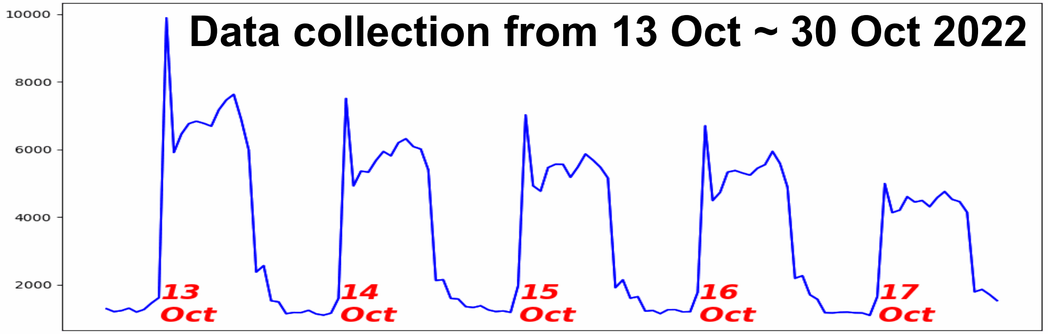
The Hong Kong Polytechnic University
香港理工大學

Department of Computing
電子計算學系

Department of
Building Environment and Energy Engineering
建築環境及能源工程學系

Background

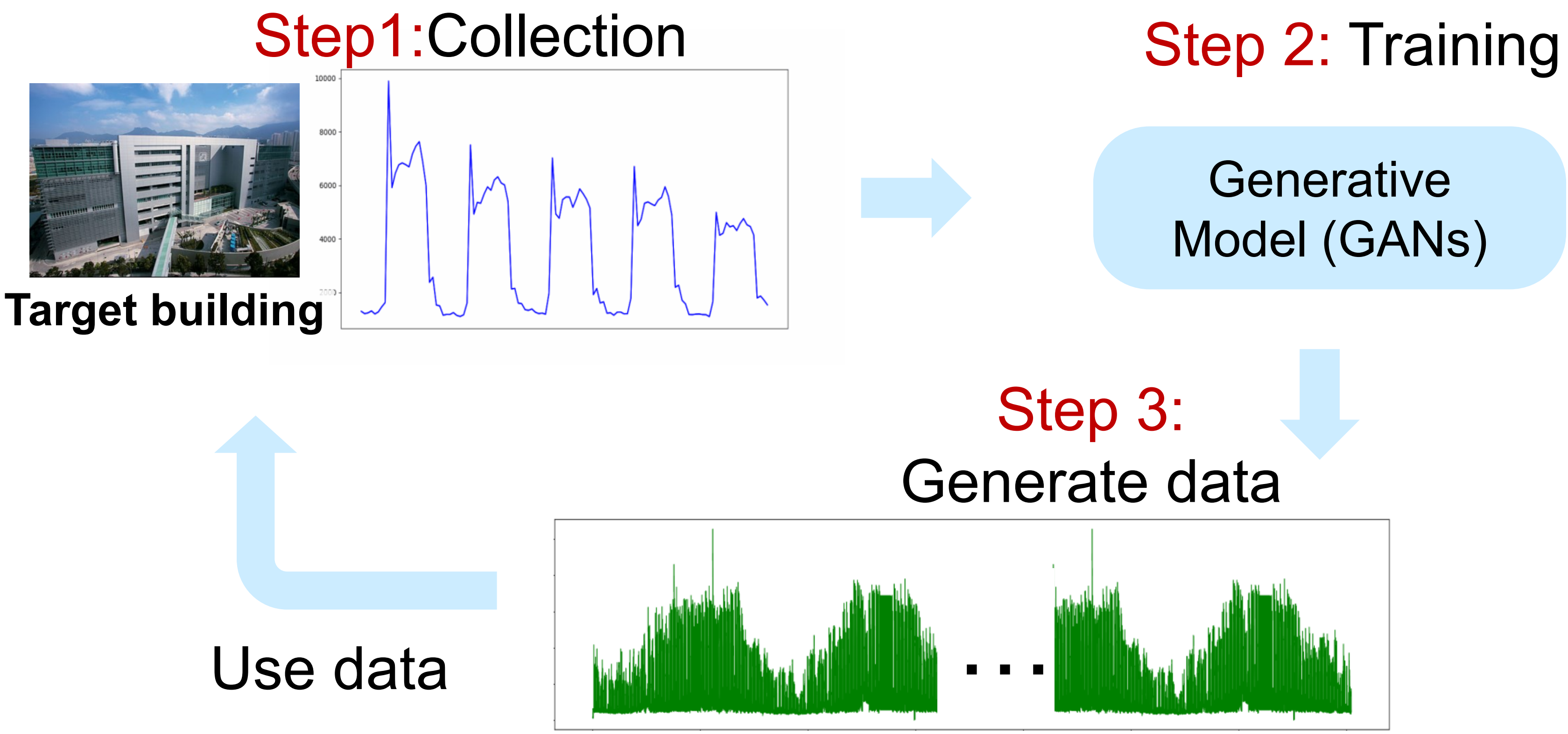
- Time-series building load data
  - The fundamental of many applications, such like load forecasting, etc.
  - Load data is collected gradually over time.



- **Building data Augmentation**
  - ML-based applications require a large amounts of data for training/testing ML model.
  - Data augmentation (DA) scheme is necessary.

Existing Data Augmentation scheme

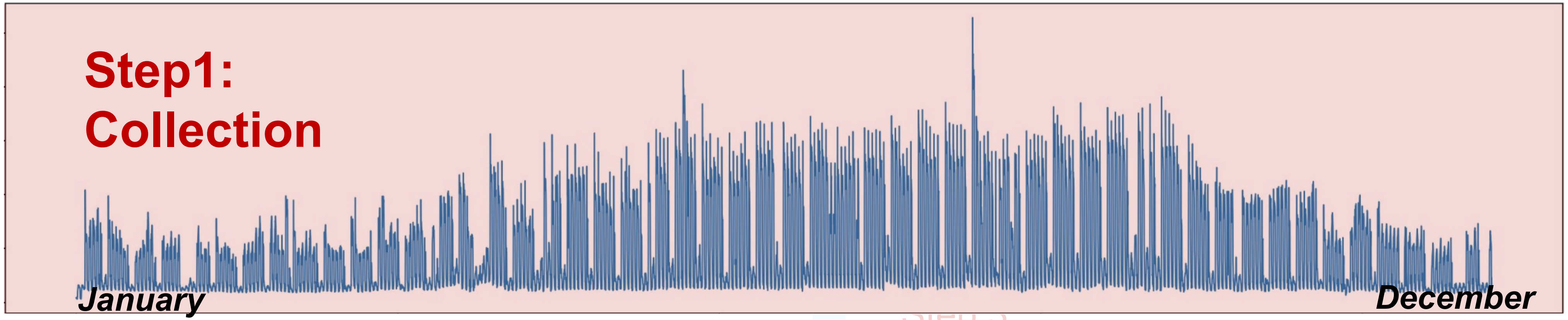
- Most are based on **generative models, i.e., GANs.**



Existing Data Augmentation scheme

A long period, usually year-level

Target building



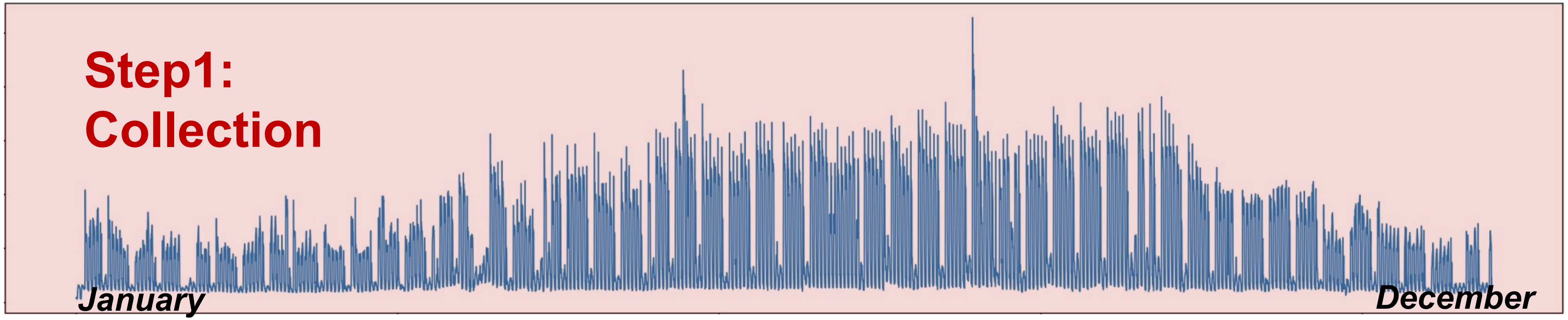
Step 3:
generation

Step 4:
Support
applications

Existing Data Augmentation scheme

A long period, usually year-level

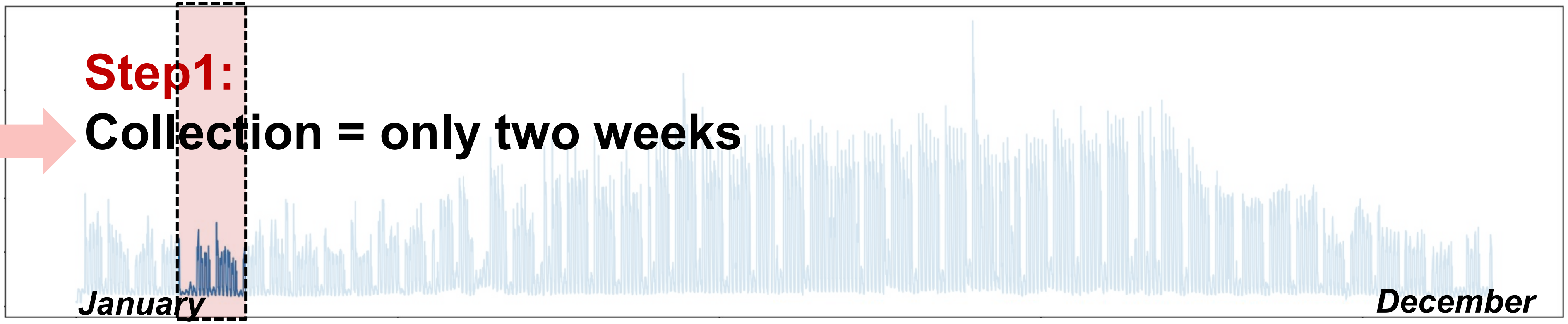
Target building



The Core is:
I can generate data after I see the full data distribution

Generative Model (GANs)

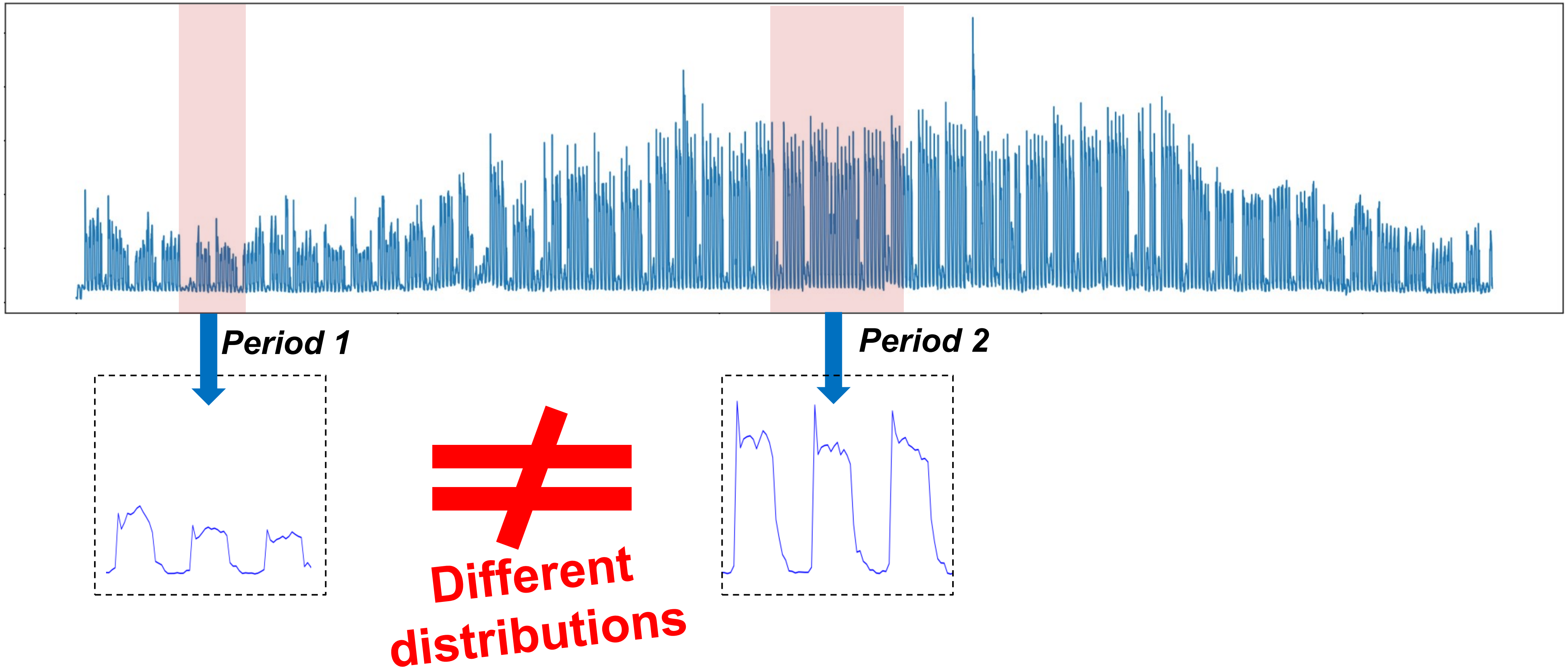
The problem under insufficient time coverage



However, the collection period can be **insufficient time coverage**, especially for the new buildings.

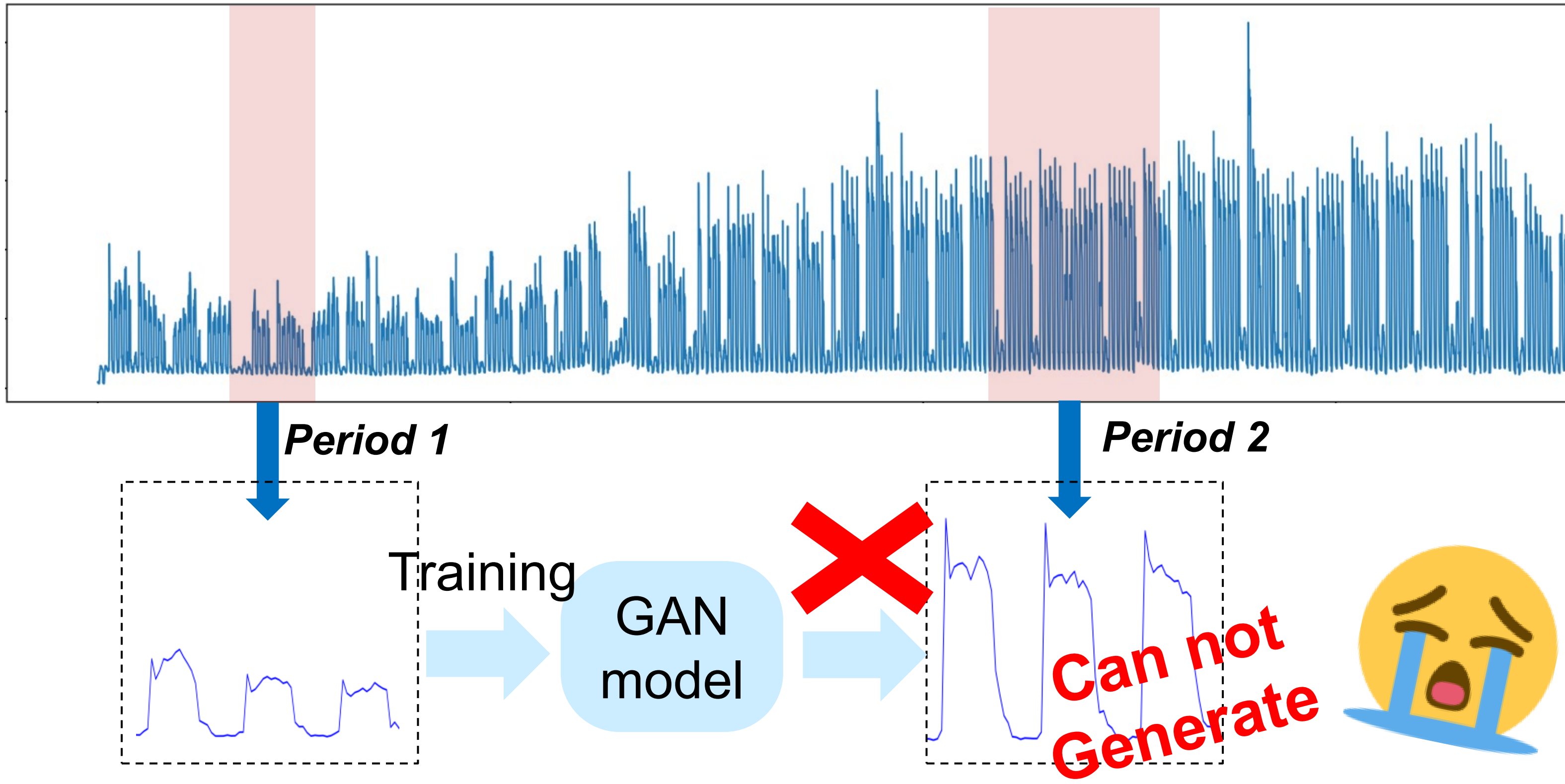
The problem under insufficient time coverage

Target building



The problem under insufficient time coverage


Target building



Directly applying GAN is not suitable for this scenario.

Potential Approach:

Data Augmentation based on *Time-series decomposition*

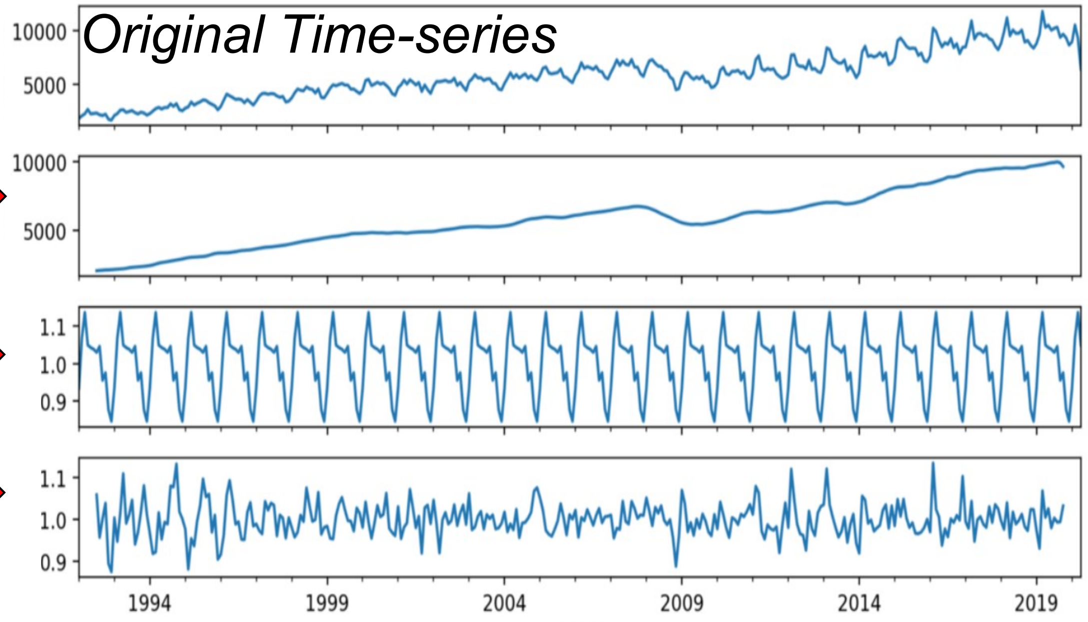
Time-series decomposition

- **Time series decomposition** assumes that the time-series can be regarded as a collection of several components.

Example: "Car Sales in the USA"

The Definition [1]:

- **Trend**: long-term progression.
- **Seasonality**: repeated and periodic fluctuations.
- **Irregular**: irregular influences, e.g., noise.

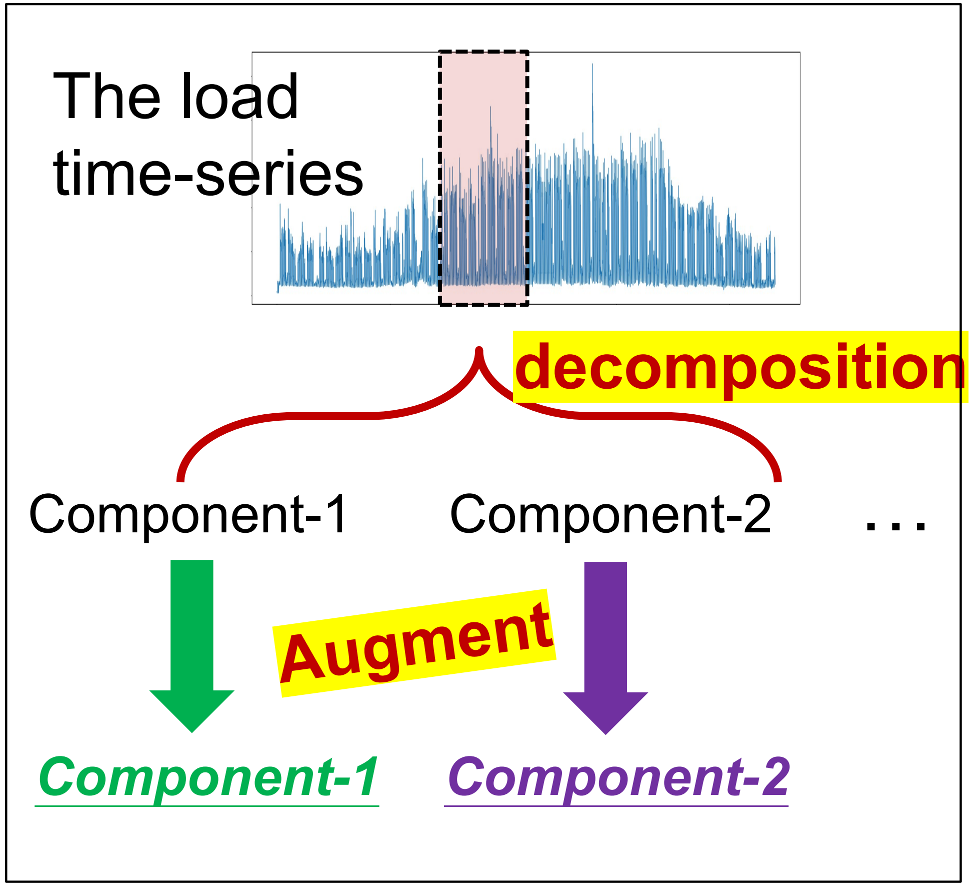


[1] M. West. 1997. Time series decomposition. Biometrika (1997).

Decomposition-based Data augmentation

The rationale:

Augmenting the data of each component can be a more targeted process.



Does these components exist in building load data?

Trend ?

Seasonality ?

Irregular ?

- Analyzed data:
  - **407** USA buildings from Genome dataset [2]

BUILDING DATA GENOME PROJECT

[2] C. Miller, A. Kathirgamanathan, et al. 2020. The building data genome project 2, energy meter data from the ASHRAE great energy predictor III competition. Scientific data (2020).

Does these components exist in building load data?

Trend → - e.g., the rate at which equipment is aging.

Seasonality

Irregular

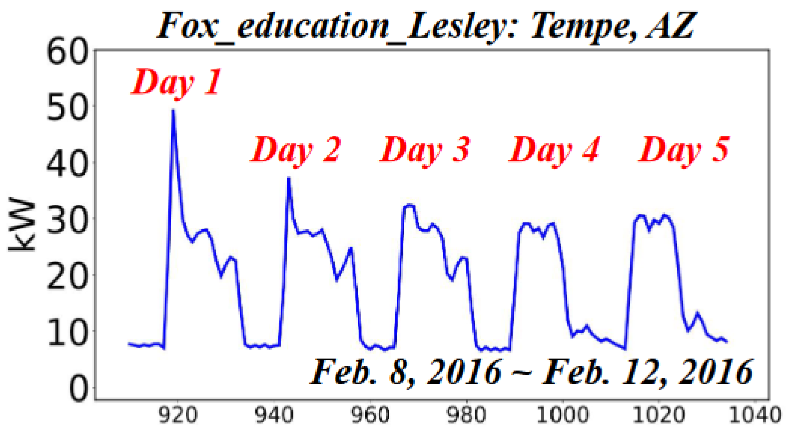
Does these components exist in building load data?

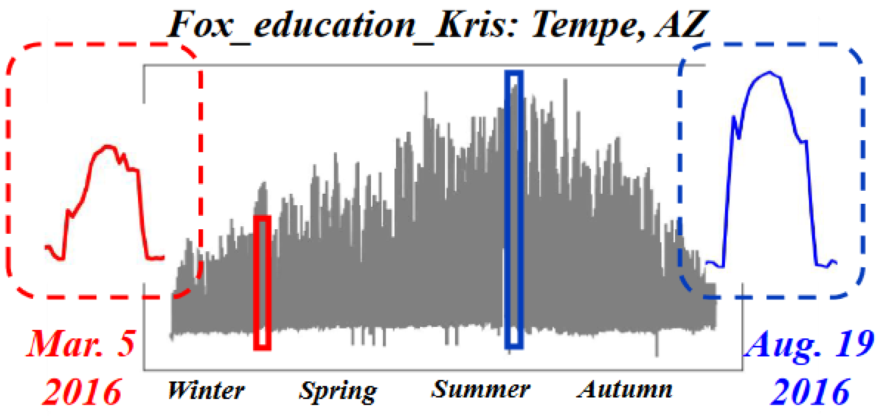
Trend

Seasonality

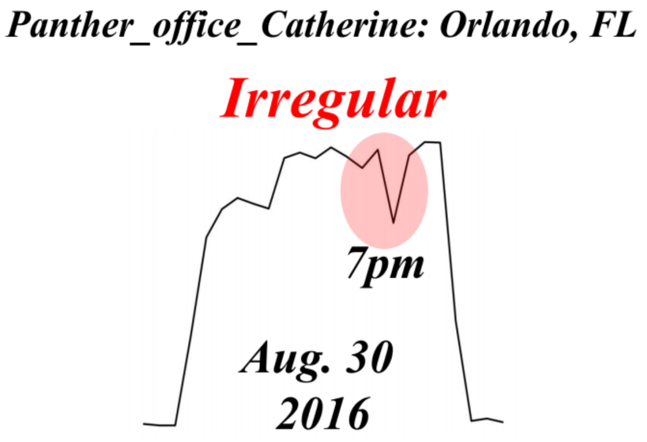
Irregular

- ✓ Daily-level (Daily load)
- ✓ Yearly-level (Seasonal contexts)





Does these components exist in building load data?



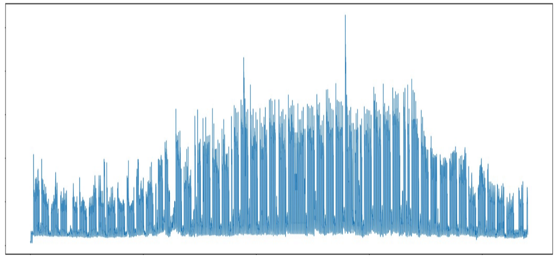
Trend

Seasonality

Irregular

- ✓ Reflects noises for a short duration, i.e., spike/dip loads within a stable load period

Analysis Summary



Trend

Seasonality
- *Daily load*
- *Seasonal context*

Irregular
- *Noise*

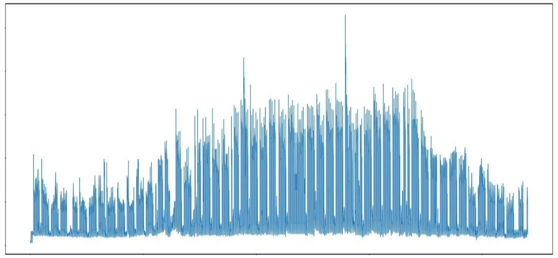
Tips:

1. All the components exist in the load → decomposition fits our scenario.

Analysis Summary

Tips:

1. All the components exist in the load → decomposition fits our scenario.
2. Seasonality and Irregular are more obvious.



Trend

Seasonality

- Daily load

- Seasonal context

Irregular

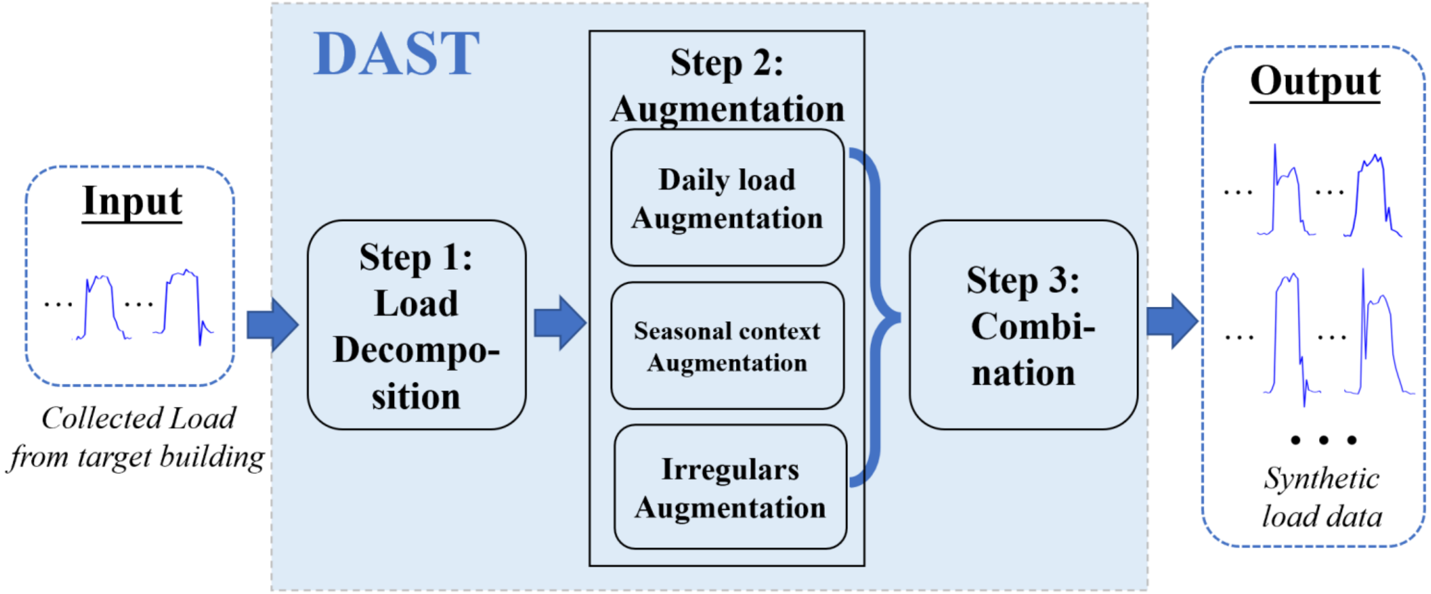
- Noise

This work focus on these three.

Data Augmentation scheme Design

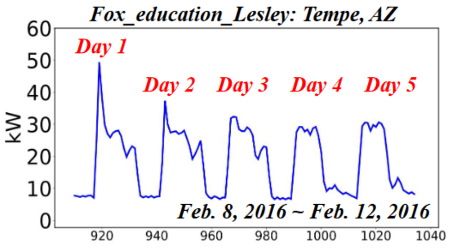
Design Overview

- Goal
  - To minimize the distance between the synthetic load and the real data, given the target building with an insufficient distribution of data collection.
- Decomposition-based augmentation scheme (DAST)

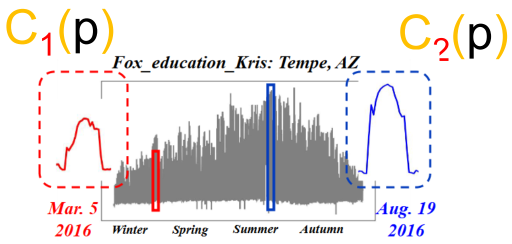


Step 1: Load time-series decomposition

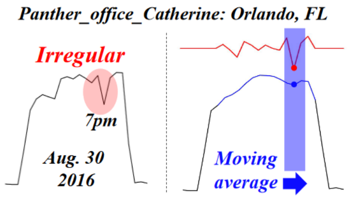
- **Challenge 1:** Decompose the sequence of { D } into { $C(p)$ } and { R }



daily pattern p



Seasonal Context $C()$



the irregular R

$$D(\, C(p),\, R \,)$$

Daily load

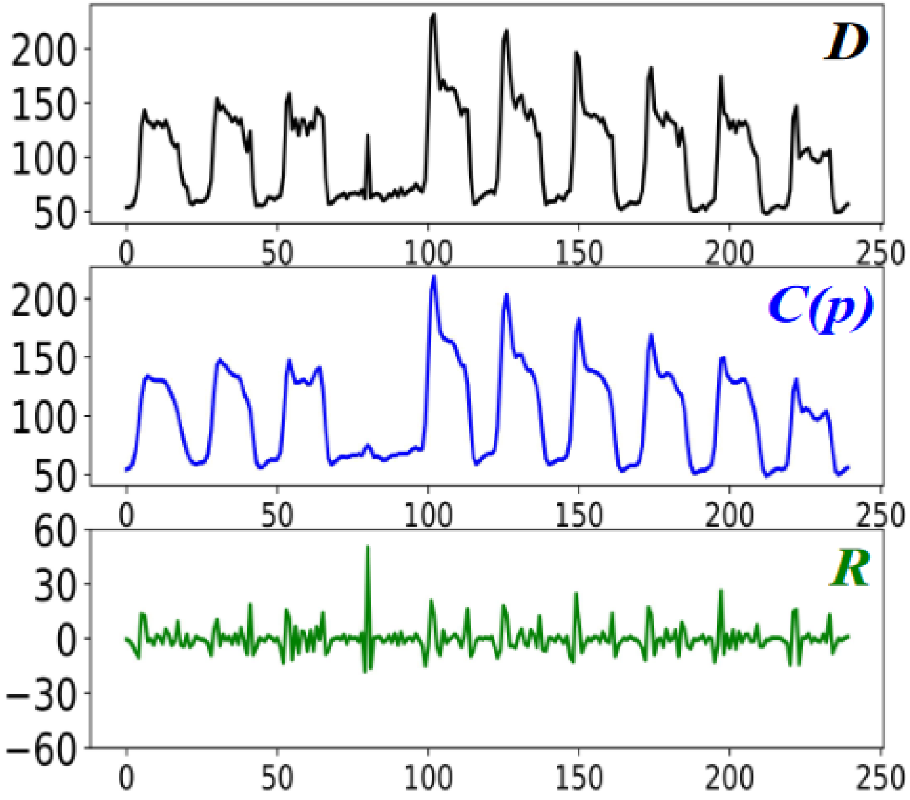
$C(p)$ R

Step 1: Load time-series decomposition

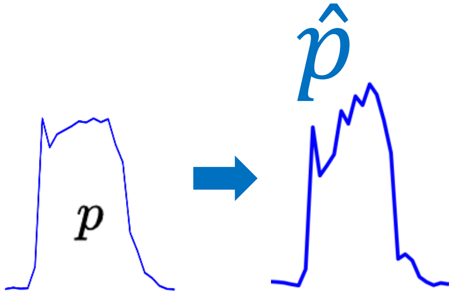
- Challenge 1: Decompose the sequence of { *D* } into { *C(p)* } and { *R* }

- **Solution 1:** A classical season-trend decomposition (STL) method, with a calibration for day type.

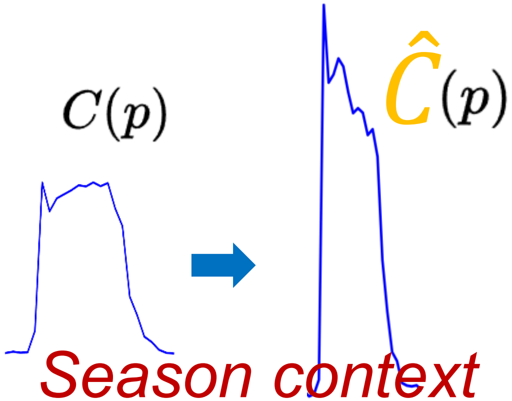
Result:



Step 2: Augmentation for the components



Daily pattern augmentation



Season context augmentation

Irregular augmentation

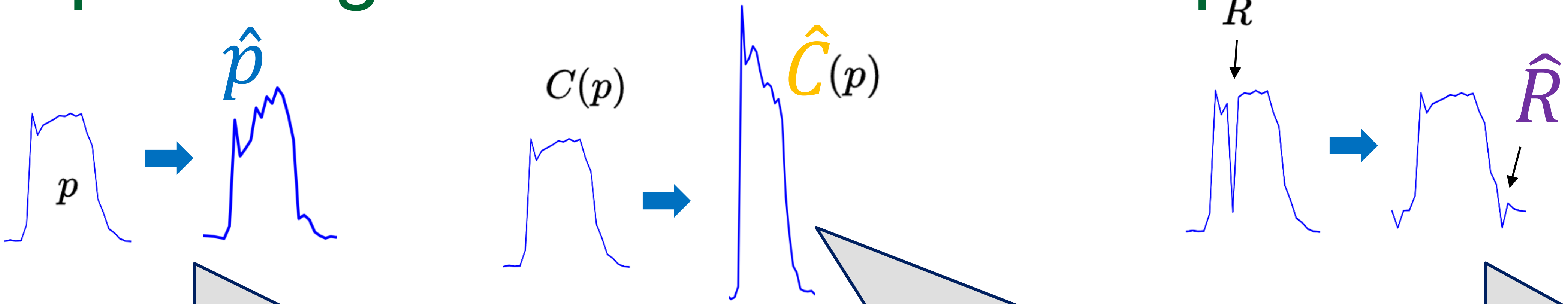
$\hat{C}(\hat{p})$ $\hat{R}$

$D(\ C(p),\ R\ )$

Daily load

- **Challenge 2:** Minimize the distance between the augmented component and the real ones.

Step 2: Augmentation for the components



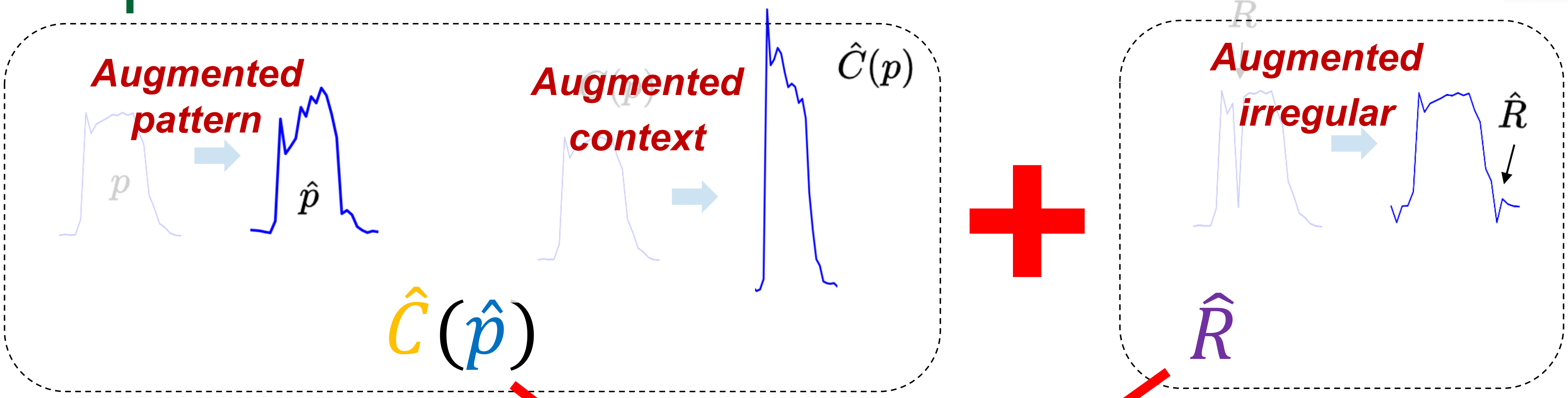
Solution:
- **A k-means clustering** to recognize the daily patterns.
- **A classical Pattern mixing method** to generate daily patterns.

Solution:
- **A learning-based domain translation algorithm** to learn transformation operations.
- **A metadata-clustering method** to search the training data.

Solution:
- **A statistics-based method (KDE)**

- Challenge 2: Minimize the distance of the augmented component and the real ones.

Step 3: combination



$$D'(\hat{C}(\hat{p}), \hat{R})$$

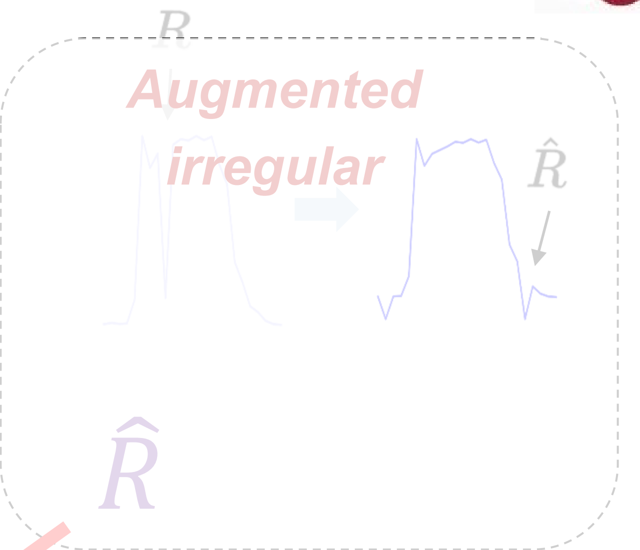
Synthetic Daily load

- **Challenge 3:** To remove the low-fidelity D’

Step 3: combination

Solution:

- **A contrastive learning-based module** to learn the relation of *C(p)* and *D*, and then recognize the low-fidelity *D’* based on binary-classification.



$$D'(\hat{C}(\hat{p}), \hat{R})$$

Synthetic Daily load

- Challenge 3: To remove the low-fidelity D’

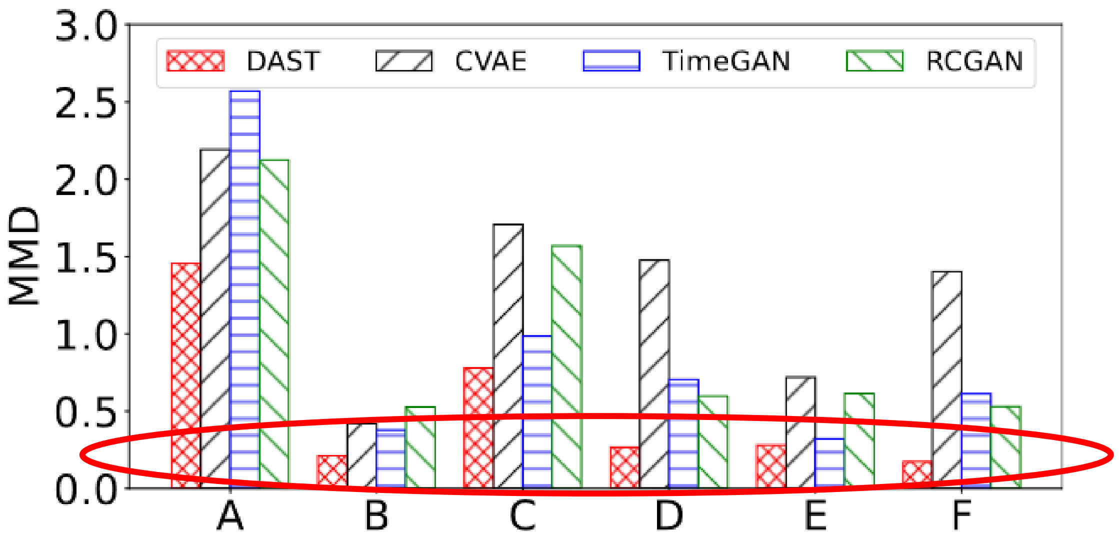
Evaluation Setup

- Dataset
  - Six buildings according to different dimensions:
    - building type, location, and building size.
- Settings of given data size
  - Two weeks / one month / three months
- Baselines:
  - RCGAN
    - RCGAN adopts RNN in Vanilla GAN, a benchmark model for time-series data augmentation.
  - cVAE
    - conditional variational autoencoder. It can generate data for specific types of months.
  - TimeGAN
    - A state-of-the-art GAN for time-series scenarios.

Building	The ID in Genome	Location	Floor area (sqft)
A	Rat_public_Emilee	Washington	22500
B	Rat_public_Isabel	Washington	16576
C	Fox_office_Joy	Tempe	70837
D	Fox_education_Virginia	Tempe	12773
E	Bear_education_Iris	Berkeley	58733
F	Peacock_education_Ophelia	Princeton	120836

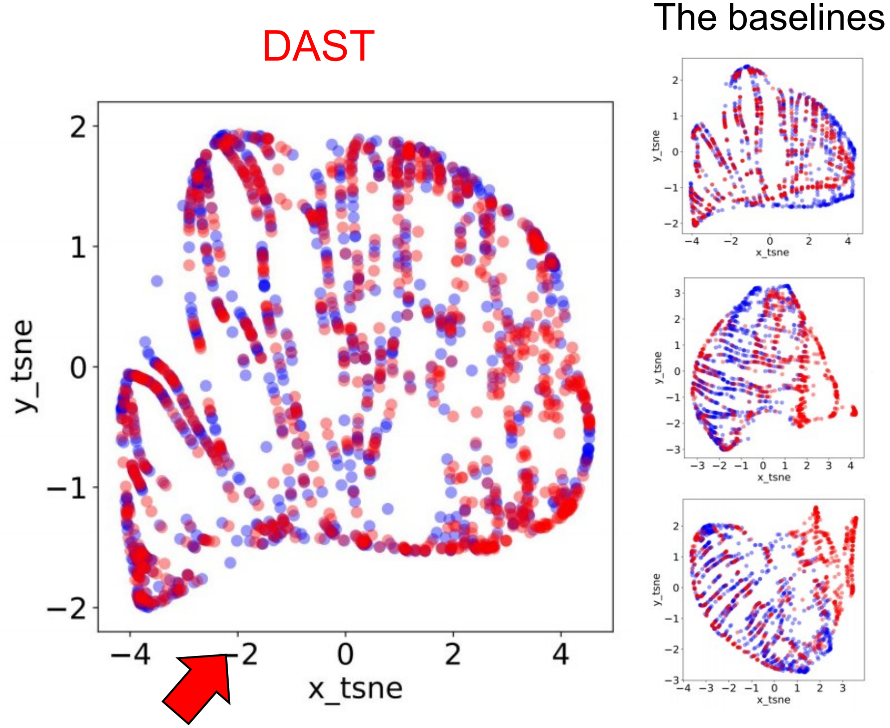
Improvement of DAST

- Quantitative evaluation:
  - MMD (maximum mean discrepancy), lower the better



Reduce the distance (distribution between real data and synthetic data) by 49%.

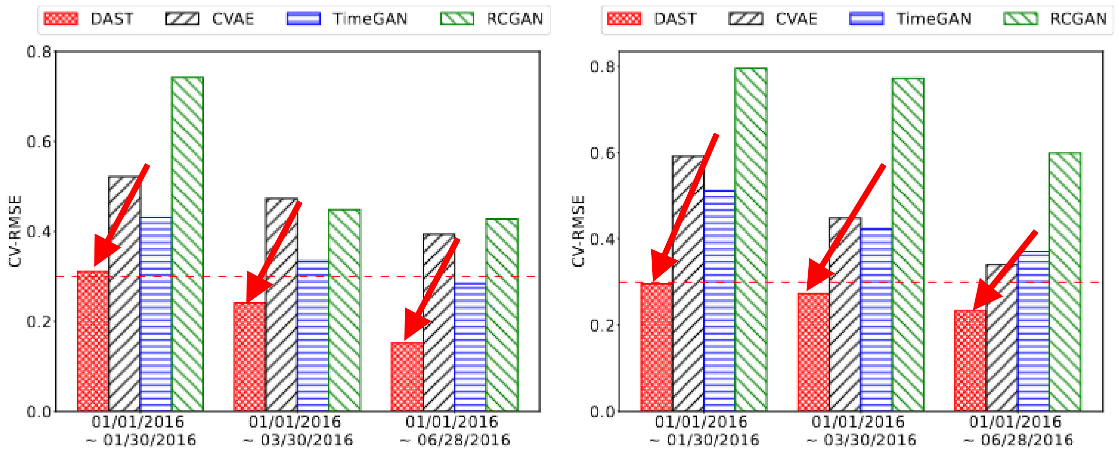
- Qualitative evaluation:
  - t-SNE visualization



Notably better overlap with the original data.

Case study: the benefit to building load forecasting (BLF) tasks

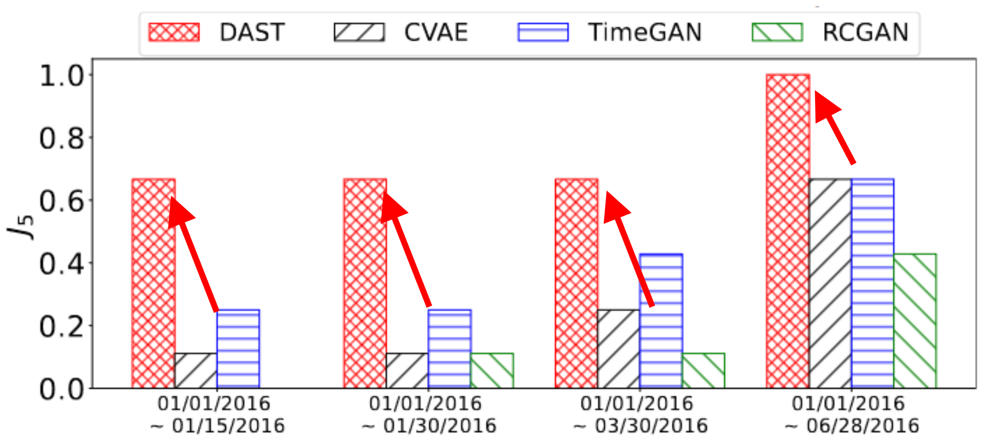
- Task 1: BLF model ***training***
  - Use DAST to support RNN, LSTM training
  - Train-on-synthetic and test-on-real (TSTR)



(a) RNN model (b) LSTM model

Accuracy improved by 46% and 48% for RNN /LSTM.

- Task 2: Trained BLF model ***testing***
  - Differentiate the performance of 30 BLF models for a target building.
  - Top-5 ranking.



Ranking accuracy improved by 64%.

Conclusion

1. We proposed DAST, a decomposition-based data augmentation scheme for insufficient data distribution scenario.
2. We analyzed decomposed components in real buildings and developed appropriate augmentation schemes.
3. We conducted qualitative and quantitative experiments to assess the quality of generated data.
4. We conducted case study on building load forecasting tasks.

Thanks for listening!