# Abstract

Completeness of data is vital for the study and forecasting of building management systems datasets. Gaps can result in lower comfort of living and more inaccurate forecasting models which could cause less efficient power usage. This study creates a guideline of imputation methods that can address the missingness in data. The guideline is created by a comparative experiment comparing four imputation methods of various backgrounds. The test data used in this study are real data from KNMI and 120 Building Management System Net Zero Energy Building houses. The focus of the paper is to impute trends back into the missing data and not on predicting every entry as accurate as possible. This makes Root Mean Squared Error less suited for evaluation and instead Variance in Error, Maximum Error and statistics such as Kurtosis and Skewness are used to evaluate the data.

# Introduction

Missing data is a common occurrence in time-series data, for this specific case causes include faulty sensors or errors in data storage. Missing data can cause downstream applications to malfunction and can thus have serious consequences. Missing data in Building Management Systems (BMS) can cause underperforming building services e.g., lower comfort or higher power usage, or in worst-case scenarios, building breakdown as system control decisions are based on the collected data.

Imputation methods evaluated in this paper are selected from previous research that has been done into the imputation of time series data. The methods that are selected for evaluation in this paper are: forward fill (FFIL or LOCF), backfill (BFIL), K-Nearest Neighbor regression (KNN), Recurrent Neural Networks (RNN), and Hot Deck (HD).

Hot Deck has been outperformed by machine learning-based models in the past as seen in (Sree Dhevi, 2014) but it is applicable due to the number of similar units. The time series imputation performance of different types of RNN's has been studied before in Che et al. (2018). Which concluded that when the Gated Recurrent Units (GRU) architecture was properly set up "*it pulled significantly ahead of non-deep learning methods"* [2].

Pazhoohesh et al. (2019) [3] found that for datasets where 10 to 30 %of the data is missing KNN regression does great compared with eight other methods. Poloczek et al. 2014 [4] analyzed the use of KNN regression and FFIL and found that both did well for the study, but that KNN regression dominated other methods.

There are limited studies to clarify how to deal with missing data in BMS datasets. Previous research has focused on lighting and occupancy [3] data or created a generic framework for imputing data from multiple sensors [5]. In the case of (Zhang,2020) it is stated that a more generic plug-n-play framework is to be further studied. This study will not build on the framework created by (Zhang,2020) but tries to give a guideline as to use the methods evaluated in this paper. The research focused on imputing trends rather than accurately imputing data in a single moment in time.

This paper aims to evaluate and compare the imputation performance of the following methods: KNN regression, FFIL, BFIL, RNN, and Hot Deck. The imputation performance has been evaluated by making use of various criteria to facilitate the choice of the most suitable method for each scenario.

The method section contains a description of the datasets used to evaluate the imputation methods' performance and a description of the selected imputation methods and performance criteria. The results section presents evaluation results for each imputation method per gap and each of the seven selected columns.

# Methodology

## Real building datasets

Building management systems store data from the building such as fluid temperature, flow rate, and working mode and outdoor information like solar radiation and temperature. To validate the performance of the imputation methods two data sets have been used, twenty-five weather stations from the Royal Netherlands Meteorological Institute (KNMI) and BMS data of hundred-twenty residential Net-Zero energy houses. The NZEB BMS time series data set contains data from 2019 and is supposed to have five-minute interval data measurements (105096 rows). The KNMI data set contains data from 2018 to 2020 and is measured at hourly intervals (17545 rows). The only change made to the datasets was converting the timestamps to Python Date Time objects.

## Columns selected for imputation

For the efficiency of research seven columns are selected across the data sets to evaluate imputation performance. The selected features from the NZEB BMS data set are power usage, heat pump operation mode (op_mode), heat pump flow temperature, and C02 sensor C02 measurements. KNMI columns that were selected for imputation are solar radiation, temperature, and humidity. The NZEB BMS columns are selected based on data classifications present in the datasets. This is done to measure the difference in performance for Imputing various data classifications. The KNMI columns were selected for being correlated, which was confirmed using the Pearson method.

**Table 1. Title: Columns selected for imputation**

**Description: Columns with dataset origin, device origin, unit of measurement, and classification**

| Column name | Dataset | Device | Unit of measurement | Classification |
|---|---|---|---|---|
| Temperature | KNMI | - | C (in 0.1c) | Interval |
| Global Radiation | KNMI | - | j per cm2 | Ratio |
| Humidity | KNMI | - | % | Ratio |
| Flow_temp | BMS | Alklima Heat Pump | C | Interval |
| op_mode | BMS | Alklima Heat Pump | 0-6 modes | Nominal |
| Power | BMS | Smartmeter | W | Ratio |
| C02 | BMS | C02 Sensor | PPM | Ratio |

## Evaluation method

In order to evaluate the performance of imputation methods under the same reproducible conditions, a pipeline has been developed. The pipeline performed the following tasks: loading data, creating gaps, imputing the artificial gaps, calculating imputation performance criteria and storing evaluation results. The pipeline code and trained models can be found in the appendix of this paper. No changes are made to the KNMI and BMS nZEB datasets.

## Gap creation

To evaluate the performance of each imputation method artificial gaps are created in both datasets. The gaps come in different sizes to evaluate the performance of each imputation method over various sizes. Gaps are created along the rules listed in the table below and are generated using a set random seed. The random seed is also made use of for gap locations and the actual gap size.

**Table 2. Title: BMS artificial gaps information**

**Description: NZEB BMS gap sizes with minimum size, maximum size and percentage of data missing**

| Nr. | Min_size | Max_size | % Of data |
|---|---|---|---|
| 1 | 5 min | 60 min | 15 |
| 2 | 1 hour | 6 hours | 4 |
| 3 | 6 hours | 24 hours | 1.5 |
| 4 | 24 hours | 72 hours | 0.5 |
| 5 | 72 hours | 168 hours | 0.01 |

**Table 3. Title: KNMI artificial gaps information**

**Description: KNMI gap sizes, minimum size, maximum size and percentage of data missing**

| Nr. | Min_size | Max_size | % Of data |
|---|---|---|---|
| 1 | 1 hour | 6 hours | 15 |
| 2 | 6 hours | 24 hours | 5 |
| 3 | 24 hours | 72 hours | 1.5 |
| 4 | 72 hours | 168 hours | 0.005 |

## Imputation methods

Three imputation methods have been applied in this research: Hot Deck, GRU RNN, KNN regression, FFIL, BFIL. The methods have been selected from techniques used in previous literature and aim to have a wide scope on the imputation approach to facilitate the characterization advantages and disadvantages of each method.

### Deterministic and stochastic regression 45 min

### KNN regression

K-Nearest Neighbor is a nonparametric imputation method that works by taking the average of a gaps' K-number of neighbors. By taking the average from the  K-number of neighbours KNN is inherently vulnerable to outliers. To mitigate the impact of outliers KNN is set up to weigh nearer neighbors heavier than further away neighbors. In Pazhoohesh et al. (2019) KNN achieved the best result compared to methods tested when selecting the K amount of neighbors value according to missingness percentage in data. In this study, the best K value is looked for again due to the difference in time intervals and different gap sizes. The K values tested in this paper are: 1,5,10,15,20.

### BFIL and FFIL 30 min

Back or forward filling (also called LOCF) works by carrying the last observation forward or backwards on the missing entries. BFIL and FFIL can introduce substantial bias in datasets that do have non-constant values [6]. Columns such as power usage will suffer the most from this due to the inherent spikiness it is expected that this will worsen with larger missing data sequences. LOCF is still a common imputation method to this day and has been compared before in time series imputation performance in [3-5].

## Hot deck

### *Introduction to Hot Deck*

Hot-deck imputation is a method for handling missing data in which each missing value from a recipient is replaced with an observed value from a similar unit (the donor). This method applies perfectly to this study since there are multiple units (different houses or different weather stations' data).

This method is used extensively in practice, but the theory behind the Hot Deck is not as well developed as that of other imputation methods, leaving researchers and analysts with limited guidance on how to apply the method, the main challenge being the donor selection.

In some versions, the donor is selected randomly from a set of potential donors, which is called the donor pool. In other, more deterministic, versions a single donor is identified, and values are imputed from that case, usually the "nearest neighbor" based on a dataset-dependent metric (i.e.: the mean when imputing temperature time series).

### *Implementing the donor selection*

#### In theory

In the case of this research, the donor selection was based on pattern recognition.

It works by taking an extract containing data before and after a series of missing values (a gap) found in the recipient.

To find the best matching segment of data from a donor, the recipient's extract would then be compared to similarly sized extracts from the same time period in a donor.

Using the difference in the mean of the donor's extracts and recipient's extract, the values from the donor's extracts can be shifted towards those of the recipient — except when imputing classification data —.

The sum of the absolute difference between the extracts can now be used to sort the comparisons: the smaller the sum, the better the pattern matches.

The operation can then be repeated throughout each donor of the donor pool, for each gap, to find the best possible match before finally importing data into the recipient.

#### *In application*

This donor selection method has been applied in two versions for this paper.

Whereas the first iteration had a focus on precision, using interpolation to have the most accurate value between two data measurements, for example. The second iteration focused on improving processing time, by vectorizing the search algorithm.

But the processing time improvements had a negligible cost in precision. Which was even more diminished by the ability to compare the recipient's extract to superior amounts of data from each donor for every gap (equivalent to a month plus the gap size).

## GRU RNN

| **Method** | **Abbreviation** | **Category** | **Description** | **Library used** |
|---|---|---|---|---|
| Backfilling | BFIL | Simple | Use the next cell to fill a gap | Pandas.DataFrame.fillna |
| Forward filling | FFIL | Simple | Use the last cell before the gap to fill a gap | Pandas.DataFrame.fillna |
| KNN regression | KNN | Simple | Take the average of K-number of nearest neighbors | Sklearn.impute.KNNImputer |
| GRU RNN | RNN | Neural Network | | |
| Deterministic-Regression | DREG | Linear | | Sklearn.linear_model.LinearRegression |
| Stochastic-Regression | SREG | Linear | | Sklearn.linear_model.LinearRegression |
| Hot deck | HD | Statistical | Take data from a different unit with the same trend. | **None** |

## Imputation evaluation criteria

Given the fact that the aim of the research is to create a selection of the most suitable imputation methods for certain applications, it is necessary to use several criteria to properly characterize the performance of the analyzed imputation methods. The selected criteria are Variance in Error (VE), Maximum Error (ME), Percent Bias (PB) and Root Mean Squared Error (RMSE).

VE and ME are selected for giving insight into the capacity of a given imputation method to follow trends. PB is used to understand whether a method tends to over or underestimate compared to the original values. RMSE is a popular statistic when evaluating the performance of imputation methods but for this study, it will only serve as a generic performance indicator. This is done because RMSE gives no insight into the potential shift in time for trends and only cares about the average error of imputation for each entry. Looking only at RMSE or other statistics like Mean Absolute Error (MAE) or Mean Absolute Percentage Error (MAPE) gives no insight into the way an imputation method imputes trends in gaps.

To give insight into the effect of imputation on the dataset the Skewness and Kurtosis are calculated. These statistics can give an insight into the shift in the distribution of values and spikiness of the data in original and imputed data. Predicting trends is a focal point of this study and skewness and kurtosis can give insight into the shift in data from original to imputed data.

# Results

From the tables and graphs the following conclusions can be made:

- Conclusion 1

# Future work

For future work, this paper only evaluates the imputation methods only on metrics and it doesn't compare the performance of forecasting using imputed data of the selected methods. The effect on forecasting and downstream application performance is to be considered too and can provide a more complete view of imputation performance.

This paper also only contains a guideline of what imputation methods to use in which scenario. For future work, this could be improved to an automated framework. The pool of imputation methods for this study was also limited so this framework could expand on that also.

# References

1. Sree Dhevi, A. T. (2014). Imputing missing values using Inverse Distance Weighted Interpolation for time series data. 2014 Sixth International Conference on Advanced Computing (ICoAC). https://doi.org/10.1109/icoac.2014.7229721

2. Che, Z., Purushotham, S., Cho, K., Sontag, D., & Liu, Y. (2018). Recurrent Neural Networks for Multivariate Time Series with Missing Values. Scientific Reports, 8(1). https://doi.org/10.1038/s41598-018-24271-9

3. Pazhoohesh, M., Pourmirza, Z., & Walker, S. (2019). A Comparison of Methods for Missing Data Treatment in Building Sensor Data. 2019 IEEE 7th International Conference on Smart Energy Grid Engineering (SEGE). https://doi.org/10.1109/sege.2019.8859963

4. Poloczek, J., Treiber, N. A., & Kramer, O. (2014). KNN Regression as Geo-Imputation Method for Spatio-Temporal Wind Data. Advances in Intelligent Systems and Computing, 185–193. https://doi.org/10.1007/978-3-319-07995-0_19

5. Zhang, L. (2020). A Pattern-Recognition-Based Ensemble Data Imputation Framework for Sensors from Building Energy Systems. Sensors, 20(20), 5947. https://doi.org/10.3390/s20205947

6. Little, R., & Yau, L. (1996). Intent-to-Treat Analysis for Longitudinal Studies with Drop-Outs. Biometrics, 52(4), 1324–1333. https://doi.org/10.2307/2532847