Introduction

Missing data is a common occurrence in time series data and causes will vary on a case-by-case basis. Whether it is a sensor that gives an occasional non-response during a polling or something else entirely, most of the time it is out of control for researchers. Missing data can't be ignored as it has been proven that it results in lesser research results than more complete datasets[1][2][3]. The same goes for Building Management Systems(BMS) time series data which will be the focus of this paper.

There are papers that research imputation methods suited for time series data but there is no clear guideline for imputing BMS time series data. Existing research into imputing time series does exist and has been taken into account by this paper. The methods used in this paper come from earlier research into imputing time series data but have been applied to the specific case of BMS time series data. [4][5][6]

Examples of these existing methods are K-Nearest Neighbour (KNN), statistical approaches such as taking the average and Recurrent Neural Networks (RNN). During the research for this paper, it was decided upon to narrow down the explored imputation methods to four. These methods were chosen by their characteristics. The characteristics that were selected for are: statistical, neural network-based and mathematical. Early on in the research other methods were explored, but were deemed as less viable when selecting the four methods.

All of the methods evaluated in this paper have been directly taken from well-known Python libraries except for Hot Deck where a guide was followed[7]. To make sure this paper is reproducible the code for every method and evaluation method is posted on GitHub[].

Data sets

Integrated Climate Energy Module (ICEM) by Factory Zero

The data sets on which this paper's results and conclusions are based were provided by the research group (Energy in Transition) from The Hague University of Applied Sciences (THUAS). The data is from an Integrated Climate Energy Module (a type of BMS) made by a Dutch company called Factory Zero. The data contained 120 houses with BMS's installed. The data contained data for the entire year of 2019. Every house datasheet contains a smart meter, Alklima heat pump, thermostat, C02 sensor and fifteen other devices.[8]

The BMS module is supposed to poll every five minutes but due to varying circumstances that might not happen which creates gaps in the data. When a poll fails to happen no new row is created so the gaps are not immediately visible. The gaps only become clear when taking a look at the time differences between each data point.

KNMI

Meteorological data from the Royal Netherlands Meteorological Institute (KNMI) was used for the purpose of verifying and testing imputation methods on a different time series data set. The data contained in this data set were hourly weather pollings taken in De Bilt weather station between the first of January 2018 and the first of January 2021.

Scope

This paper will not touch on how the gaps in data can be prevented or how the data will be used afterward. The only goal for this paper is to give a clear outline of what method to use in which scenario when imputing BMS time series data. Due to a limited amount of time, it was not possible to test the imputation methods on every column present in the BMS house and KNMI data. For this reason, a selection of features was created.
The features selected from the BMS were: smart meter power usage, Alklima heat pump op_mode and Alklima heat pump flow temperature. The features that were selected from the KNMI data were as follows: temperature, solar radiation and humidity.

	Unit of meassurement		

Methodology

Data preparation

For loading the data into the pipeline the Pandas library was used to convert the data from the original Microsoft Excel or CSV file. The data sheets provided had no gaps in the data aside from the missing rows. The only change made to the data is converting the UNIX-based time stamp to a Python date time object format. The only change made to the KNMI data was placing NaN's on empty row values by Pandas.

Gap creation

Data that would be in place of the gaps is lost forever and can never be restored only guessed at. To evaluate the results of imputation compared to the original data artificial gaps are required. The difference with the "natural" gaps is that in artificial gaps the data is retained in another location for later comparison with the imputed data.

The locations of these artificial gaps are determined according to a random seed set beforehand in code. Gap sizes were chosen based on what was observed in the natural gaps. For testing purposes, multiple sizes were created to measure performance on different gap sizes (see table). In practice type one would result in a dataset with random gap sizes between five and sixty minutes taking away fifteen percent of the data.

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%

To avoid chaining multiple artificial gaps together there is a minimum distance of five indexes in between gaps. When two gaps get too close to each other the second gap doesn't get created.

Imputing data on the pipeline

The methods that were tested for the purpose of imputing BMS time-series data were selected based on previous research into imputation methods for time series data. When implementing a method from research into the pipeline existing libraries were used as much as possible. For Hot-Deck and neural network-based methods this wasn't possible as it needed to be optimized for this research project.

Due to a limit in time and resources, not every feature of both KNMI and house BMS could be used in evaluating imputation methods. For BMS data the power usage, flow temperature and heat pump op_mode columns were selected. The reason for this selection is that these columns each represent a data classification type. The KNMI selection was made with the idea in mind of utilizing correlated data for imputation.

Selecting imputation methods to further research/evaluate

Only four imputation methods will be compared in this paper but during research, the following methods were also tested: interpolation, KNN, missforest, regression, soft impute and other statistical approaches such as median and mode. These methods were dropped due to doing worse in early evaluation and are not included in this paper for this reason.

Evaluation on the pipeline

When an imputation method is done running it returns the given data with the imputed values where gaps used to be. The imputed then gets run through a number of statistics on which the performance of the method can be assessed.

For assessing the performance of an imputation method the following criteria are used: Root Mean Squared Error, Raw Bias (RB), Absolute Bias (AB), Percent Bias(PB), Maximum Error (ME), and Variance in Error(VR). These evaluation criteria all have been used in previous research[][][]. Raw bias and Variance in Error are criteria that are used to evaluate the imputation methods' performance in predicting trends.

RMSE is included in the evaluation criteria but gets less important as an indicator of imputation performance. The reason for this is that RMSE doesn't take a delayed trend into account. From early observations, it was clear that imputation methods predicted trends that were taking place but only shifted in time. With the trends shifted in time the imputation method's RMSE would be worsened while the raw bias could be close to the original. In columns such as the power usage where the total amount of electricity usage and creation is of importance, a poor RMSE score will give false insight into the imputation methods performance.

The imputed data also get compared on the difference in skewness and kurtosis compared to the original data. These statistics can bring important insights into the performance of the imputation methods. An example of this would be that if the skewness difference is high and a maximum error as well it could be that the trend/spike is later in the imputed data compared to the original values.

Hotdeck

- Donor selection (Completeness score is not yet explained)
- Vectorization
- Where did you find this in a research paper?
- General explanation of how it works (I guess)

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 we have access to data from 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 neighbour" 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).

Resources

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3130338/

Recurrent Neural Network (RNN)

- Genetic algorithm for improvement
- Where did you find this in a research paper?
- Why this type of neural network was chosen over others
- If you have a paper for this say that we dropped working on a CNN? <= CHECHU
- Link to optimized hyperparameter model

To be determined method here!

- Where did you find this in a research paper?
- If this method is regression we need to specify what regressors we used and how we decided upon those e.g. correlation finder.
- If KNN defend on the choice of amount of neighbours!

Data collection

The previously mentioned statistics get stored in a CSV file along with date and time, the gap type, the method used, what house was used and what column was used. The pipeline also generates visualizations that give a visual insight into the imputation methods' performance.

The data generated from running the imputation methods will be used to evaluate the performance. The raw data can be found here[].

Results

After running the selected imputation methods on multiple gap sizes he follw

https://www.scribbr.com/dissertation/results/

Discussion

References

[1]
https://www.sciencedirect.com/science/article/abs/pii/S0378873305000511

[2]
https://www.sciencedirect.com/science/article/abs/pii/S027269630500077X

[3]
https://journals.sagepub.com/doi/abs/10.1177/0013164404272502

[4]
https://bmcmedresmethodol.biomedcentral.com/articles/10.1186/s12874-017-0442-1

[5]
https://study.sagepub.com/sites/default/files/newman.pdf

[6
]https://www.researchgate.net/publication/334695903_A_Review_on_Missing_Data_Value_Estimation_Using_Imputation_Algorithm

[7] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3130338/

[8]
https://factoryzero.nl/producten/