Comparison of Missing Data Imputation Techniques of Time Series: A Study on Real-World Application

By: Fabian P. Wennerlof

Advisor: Dr Ranadeep Daw

Abstract

Missing data is a common occurrence in any real-world data modeling task. They possess significant difficulty for any statistical methodology as the missing value imputation techniques need to be tailored to the structure and patterns of each different datasets. In this work, we assess various missing value imputation algorithms in the context of time series data, where the temporal dependency within the data plays a critical role in imputing the missing values. We

used a real-world dataset on bacterial counts, available thanks to Palmer Long Term Ecological Research program, to evaluate these algorithms. Metrics like root mean square error and mean absolute error are used to evaluate the performance of the algorithms on the dataset. In addition, this study offers suggestions for further analysis and future research extensions with this dataset.

Chapter I: Introduction

My appreciation for statistics has come naturally from my other extracurricular inspiration – golf. As a member of the golf team at the University of West Florida, I have experienced the crucial role of statistical analysis of athlete performance data in identifying areas for improvement. By collecting data from a golfer over a long period, one can identify patterns in both successful and unsuccessful performances, which could potentially reveal indicators of performance slumps via statistical analysis. This experience highlighted the importance of analyzing any data over time to understand meaningful patterns, which is also known as time

series analysis. This is the focus of this report, which we will perform utilizing a real-world bacterial dataset.

Statement of Problem:

Time Series

Let's begin by defining time series data. A time series is any data that is collected over time. Examples include a patients' blood pressure data, stock prices, daily temperatures, and animal abundance data. Unlike the independent datasets, a time series is fundamentally different due to the presence of the time component, which influences the interpretation in two primary ways. First, the time component introduces a natural ordering of the data, i.e., how the data originated chronologically over time. Second, this inherent ordering over time series leads to a dependence structure within the data, i.e., current observations are often correlated with past values. As an example, current temperature data is likely to have more similarity with last year's data than the pattern over a few decades ago. This makes the time series analysis special, because a time series "learns from itself", i.e., the historical patterns help model the current and predict the future. [1]

Missing values and imputation

Missing values are gaps in a dataset corresponding to an observation. They are a common occurrence in most real datasets due to various challenges such as data collection, data processing, clerical errors, and poor storage management of the data. Missing values have a significant effect on many of the statistical models since statistical learning comes from the available data. For an absent data, statisticians are faced with two choices - either to exclude the

observation or use a guess-estimate of the unavailable value – both leading to more uncertainty around that missing value. Moreover, in the context of time series, missing values possess an additional challenge - it disrupts the overall flow and trend of the data. Because each current observation is dependent on the preceding values and affects the ones after, missing data can significantly affect the time series models that usually require regular availability of the data. Consequently, modeling and prediction becomes more difficult.[2]

Relevance of Problem:

An approach to deal with gaps in the data (or generally with missing values) is to use missing value imputations - an educated estimation of the missing value based on the available data. This is intuitive to human practice, where we often try to guess any unavailable information using existing data. In larger studies, for example national health or epidemiological surveys, they usually have missing data and must be addressed with robust statistical methods to avoid bias and loss of information[3]. For a time series data with missing value, the imputation step is crucial as they allow application of standard time series models (e.g., ARIMA) that require complete observation at all the time points. There is also an additional advantage: the temporal patterns within the available data points can be useful to guess the overall pattern and hence for imputing missing values. This is the central focus of this report.

Literature:

Data Source and Availability

The dataset used in this project is on bacteria abundance over discrete water samples collected from the Western Antarctic Peninsula (WAP). This dataset is made available by the

Palmer Long Term Ecological Research (LTER) Program, a key component of the network of several LTER programs by the National Science Foundation (NSF)[4]. These programs are dedicated to gathering ecological data from various ecosystems, including the WAP, to facilitate scientific research and enhance our understanding of the environment. Collected data are typically available publicly and maintained in an accessible format. For further information, interested readers can refer to the resources such as the data catalogue maintained by the Rutgers university and ERDDAP [5].

The data in this study was collected from water column samples at consistent depths over a 17-year period between 2003 and 2019. Along with bacterial abundance and diversity, additional covariates like temperature, salinity, and chlorophyll-a concentrations are also available, which may lead to future research into their interrelationships. We focus on bacterial groups in the range from sea level to 100 meters below sea level, where aerobic and heterotrophic bacteria dominate. The bacterial counts, recorded against the available date of study, form the time series data and will be used for the following part of the report.

With missing values being so common in real-world datasets, time series imputation has become a popular topic to study due to the importance of having a complete dataset. One R package that offers useful methods like linear interpolation and seasonal decomposition is "imputeTS," developed by Moritz and Bartz-Beielstein [6]. Another package introduced in 2016 is "VIM," which provides multiple imputation techniques, including mean imputation and k-nearest neighbor (kNN) [7]. Both these packages have been used in this report to use and compare different imputation techniques on the bacterial time series dataset.

Limitations:

Missing values in the data

During the 17-year span this time series dataset has been collected, values weren't collected during all months of the year. The Palmer LTER program took trips up to Western Atlantic Peninsula between the months of October and March most years as long as weather allows only this period of summer for the trips to happen. That means that there are no data values for the months April to October which can make it harder to make accurate long term and seasonal analysis. There were a few years when weather prevented the researchers from going on the expedition to WAP, such as, 2007-08 have no collected data at all.[8]

Because of all the missing values during special months, it can make it harder to make good seasonal predictions. Nonetheless, we use the time series imputation techniques on this data by separating the data into a training and testing group. Training group is where a time series was trained, and then we evaluated it on the test group to understand which method works the best. Based on the imputed series, we also visualize the overall imputed series. We use multiple different imputation techniques to generate the best complete time series by utilizing only the available data and its inherent temporal patterns.

Goal of this report:

In this report, the goal is to compare several imputation techniques to address missing values in a time series dataset. We use the bacterial count data as our time series and evaluate the performance of a few missing imputation methods on this data using metrics such as RMSE (root mean square error), MAE (mean absolute error), alongside a visual comparison of the imputed time series to assess the preservation of its original patterns. The goal is to identify the imputation method that provides the most reliable time series over the entire period. In the

following, Chapter II will detail the comparison methodologies and the results. Chapter III will offer conclusions from our study and discuss potential future directions with this dataset.

Chapter II: Main body

Data preprocessing:

The values of the bacteria count are of order 10^8 (10 million), which is harder to interpret within a plot or any evaluation metric. Therefore, the data is first scaled by dividing the counts with 10^8. Alternative options such as a logarithmic scaling or a square root scaling were considered, but they introduce nonlinearity to the scaling of the data. Given that several imputation techniques used here are linear, scaling by a constant factor of 10^8 is preferred as it preserves the linearity of the data and keeps the visual representation of the patterns intact. Another transformation done is to use monthly averages inserted of the daily values. Figure 1 shows the data at the original daily scale, which shows large variation and substantial missing data. This is very complicated for interpretation, imputation, and modeling. Instead, averaging the data monthly helps reduce the daily noise and the unavailability issue, even though it comes at the slight cost of losing finer-scale information. Across the 17-year period, the resulting monthly data offers enough availability for the subsequent imputation process. See Figure 2 for the available data.

Data:

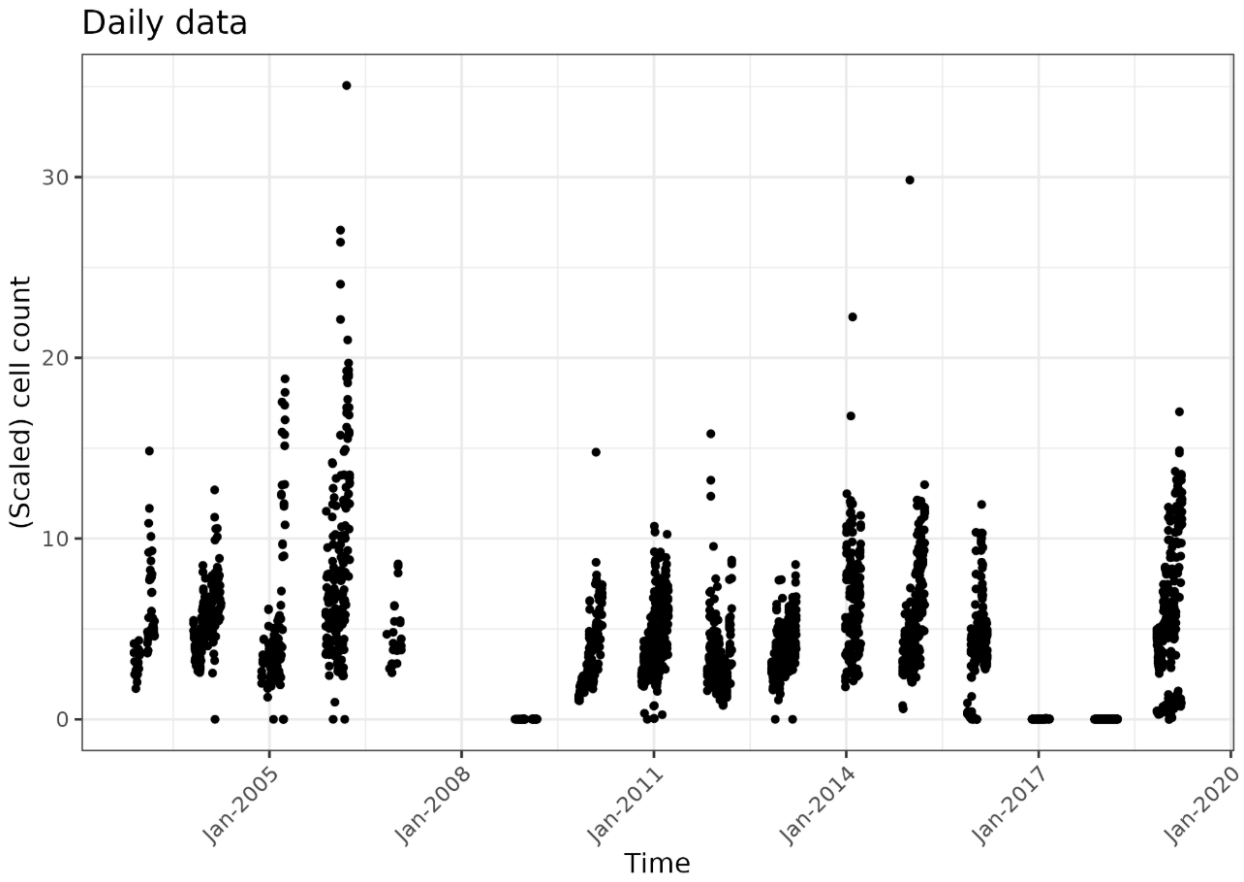

Figure 1: Daily bacterial cell count from 2003 to 2019.

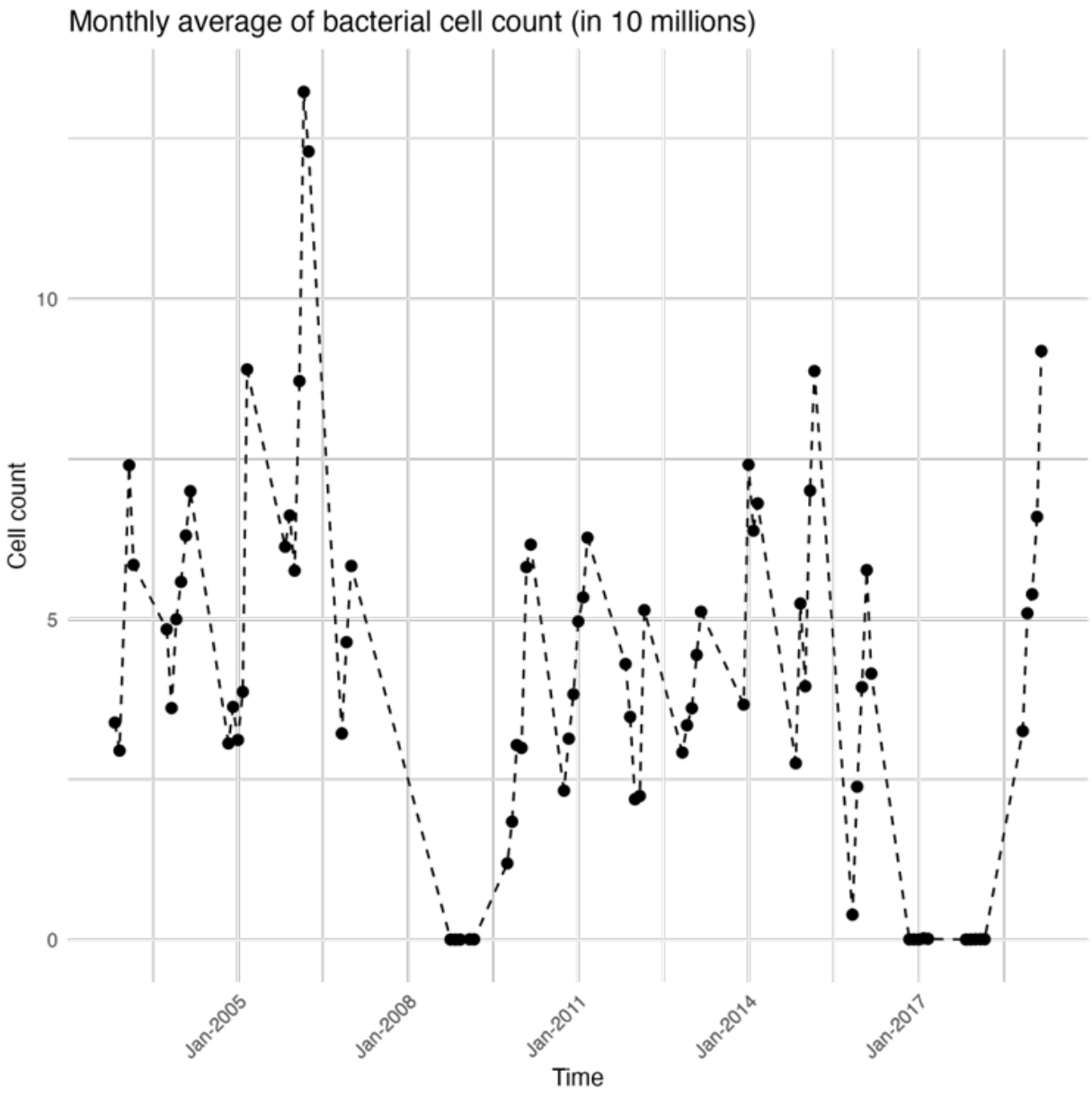

Figure 2: Monthly averaged bacterial cell count data, improved with less noise and easier to fit imputation techniques and modeling.

Some extreme outlier values were also capped, or also called replaced with the 1st and 99th percentiles. This was done to minimize the influence of unusually high or low values that

could distort the overall pattern of the time series. By reducing their impact, it helps to ensure the imputation methods were applied more effectively and accurately.

Imputation techniques

In this chapter, we discuss the theoretical details of the different imputation techniques. Throughout this section, we assume a time series $\{y_1 , y_2 ,...,y_{\{t-1\}},y_t*, y_{\{t+1\}} , \ldots , y_n\}$ where $y*$ stands for the missing value at the time point t, and other y-s are assumed observed and n is the number of known value points. For example, $y_{\{t-1\}}$ is the observed value right before the missing value y* and y_{t+1} is the observed value right after the missing value y*. We will use the notation z_t as the imputed value of $y*$ at time t. The following imputation methods are considered:

Random imputation

This is one of the simplest techniques, where a random value from the interval is used to impute the unknown value. There is not enough mathematical rationale behind this method, but this could be used as a baseline comparison for the other methods.

Last Observation Carried Forward (LOCF) imputation

LOCF is among the simple techniques, where the last observed value is used as the estimated missing value at any given time point. The inherent assumption is that the unavailable time series remains the same since the last observed one. In mathematical form, the imputation formula can be written as:

Formula: $z_t = y_{t-1}$ (1)

Mean imputation

This is also a very simple, and possibly ineffective technique, and serves as another baseline method. For the mean value imputation, all the missing values are replaced with the mean of all available values in the dataset. It assumes that the overall trend in the dataset can be best captured by the global mean of the time series. The mathematical formula is given by:

Formula: $z_t = (1/n) \sum_{i=1}^{n} \quad y_t$ (2)

Linear imputation

For linear imputation, the assumption is that the pattern in the time series changes linearly within the unavailable range. This means that a straight line would be fitted between the first 15 available points around the missing value and the estimate would lie on the straight line. The mathematical formula I given by:

Formula: $z_t = y_{\{t-i\}} + (i/(i+j))(y_{\{t+j\}} - y_{\{t-i\}})$ (3)

(k)- Nearest neighbor imputation

The k-nearest neighbor interpolation (kNN) is an extension of the LOCF method. In LOCF, we only search for the last available time. Compared to that, kNN finds k closest available time points to the given missing time point and then takes the average. This helps the process learn both from the past and the future patterns within a nearby interval. Like the regular kNN algorithm, a higher value of k leads to a smoother prediction surface, whereas a small k leads to a bumpy prediction surface. The imputation formula is written as:

Formula: $y_t = (1/k) \sum_{i=1}^{k} \quad y_{\{ji\}}$ (4)

where the subscripts stand for the nearest observations from the missing time point t.

Moving average imputation

The next imputation is the moving average imputation which is a similar but slightly restricted version of the kNN imputation. It uses a specific number of the surrounding values around the missing value to calculate an average, i.e., exactly k values before and after y. In kNN, we don't make the restriction on which direction to find the neighbors, but allow detection of the closest ones. For this task, the imputation formula looks like the following:

Formula: $y_t = (1/(2k+1)) \sum_{i=-k}^{k} y_{\{t+i\}}$ [9] (5)

Seasonal splitting imputation

This is the first imputation method among all that uses seasonal trends in the imputations. Seasonality of a time series is a pattern that repeats in a short interval, e.g., the rise in temperature in the summer months every year. Seasonal splitting imputation divides the data into a few seasonal groups with similarity within each group. Then, it separately performs imputation within each group. The methodology is advantageous when the overall data has large seasonal patterns and individual subgroups of the data are easier to handle separately.

Seasonal decomposition imputation

The seasonal decomposition idea follows the last splitting technique, but handles the pattern differently. It first finds the seasonal variability within the available time series, which follows a removal procedure of the seasonal component. This way, only the trend component is used for missing value imputation. The idea here is that when one can guess the repeating

seasonal pattern within a dataset, we do not need to worry about that component over the missing time points. Hence, imputing only based on the overall trend becomes less noisy and could lead to better imputed series.

Kalman filtering

Lastly we have Kalman filtering which is the most complex imputation method. It involves multiple different steps of estimating missing values based on prior data but then it adjusts data as more observations are made. It's a complicated approach to imputation that goes beyond the classes I have taken but I have implemented it due to the ease of the code. A description of the Kalman filter by Tony Lacey explains;

"The standard Kalman filter derivation is given here as a tutorial exercise in the practical use of some of the statistical techniques outlied in previous sections. The filter is constructed as a mean squared error minimiser, but an alternative derivation of the filter is also provided showing how the filter relates to maximum likelihood statistics." [10]

Implementation details

We first divided the existing data into two parts. A random subset of size 80% of the entire available data was used as train data to train the missing imputation algorithms. The remaining 20% data was used to evaluate how good our imputation methods performed on this sub-part of the dataset. Ideally one could repeat this train-test procedure for multiple train-test folds, which is also known as the cross validation. However, for this analysis, we only implemented it using a single train-test division.

For software implementation, R and RStudio have been used [11]. We have used a few computational packages for the missing value imputation task. The caret package was used to perform the train-test division and to calculate the evaluation metrics (RMSE and MAE). The ImputeTS package was primarily used for many of the implementation packages such as LOCF, moving average, linear, Kalman filter, random, and seasonal methods [6]. Additionally, the VIM package was used for imputation [7]. Additionally, the spatialEco package was used for the polynomial interpolation [12]. Finally, for visualization, we have used the ggplot package, one of the most common and useful tools among all the R packages [13].

Results:

These results do include both a blue line and black dots. The black dots show the original observed data points that the imputation methods have been applied around to estimate the missing values. Then the blue line is the imputed values and although it looks like a continuous line, it represents a series of individual imputed points. I chose to make it a line because it makes it easier to see the overall structure in the graph.

Random imputation

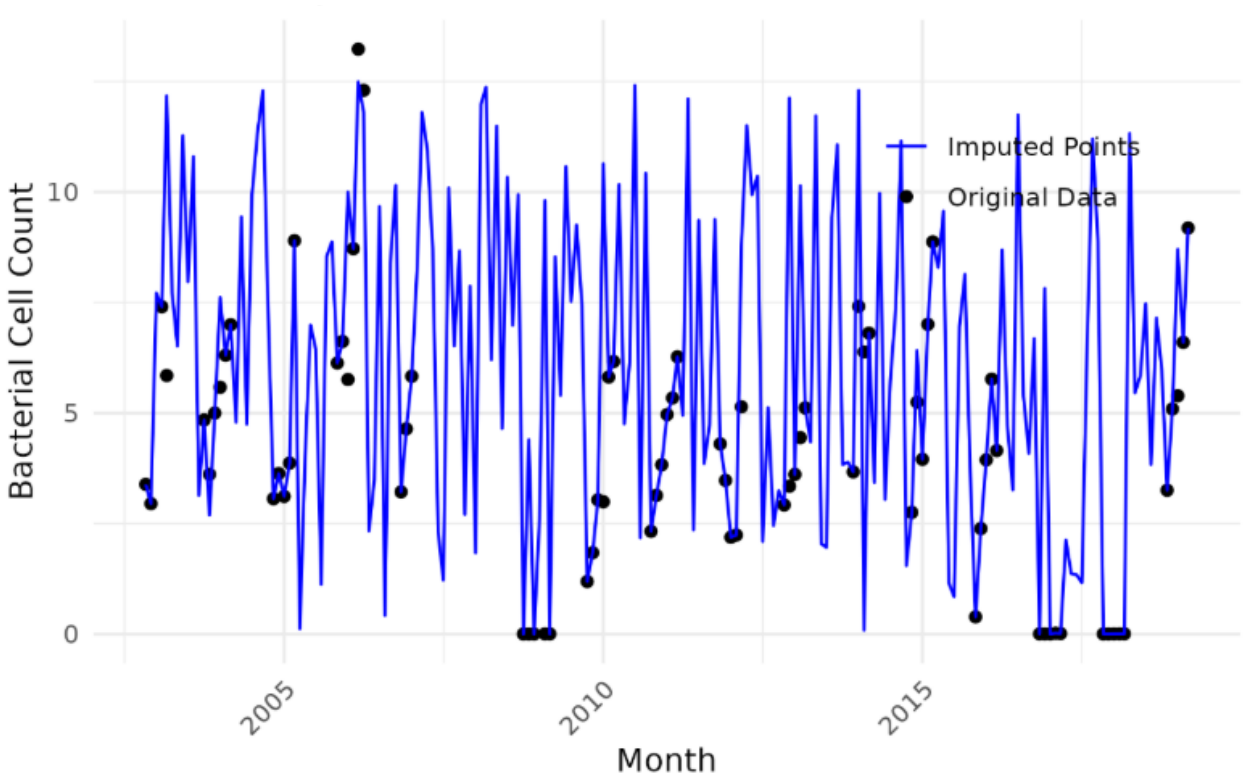

Figure 3: Effect of random imputation in time series imputation.

Random imputation has filled in every missing value with a completely random value from the data set, resulting in inconsistent data that doesn't follow the original trend. This makes it difficult to interpret or model statistically. It's not a reliable choice for this time series.

Last Observation Carried Forward (LOCF) imputation

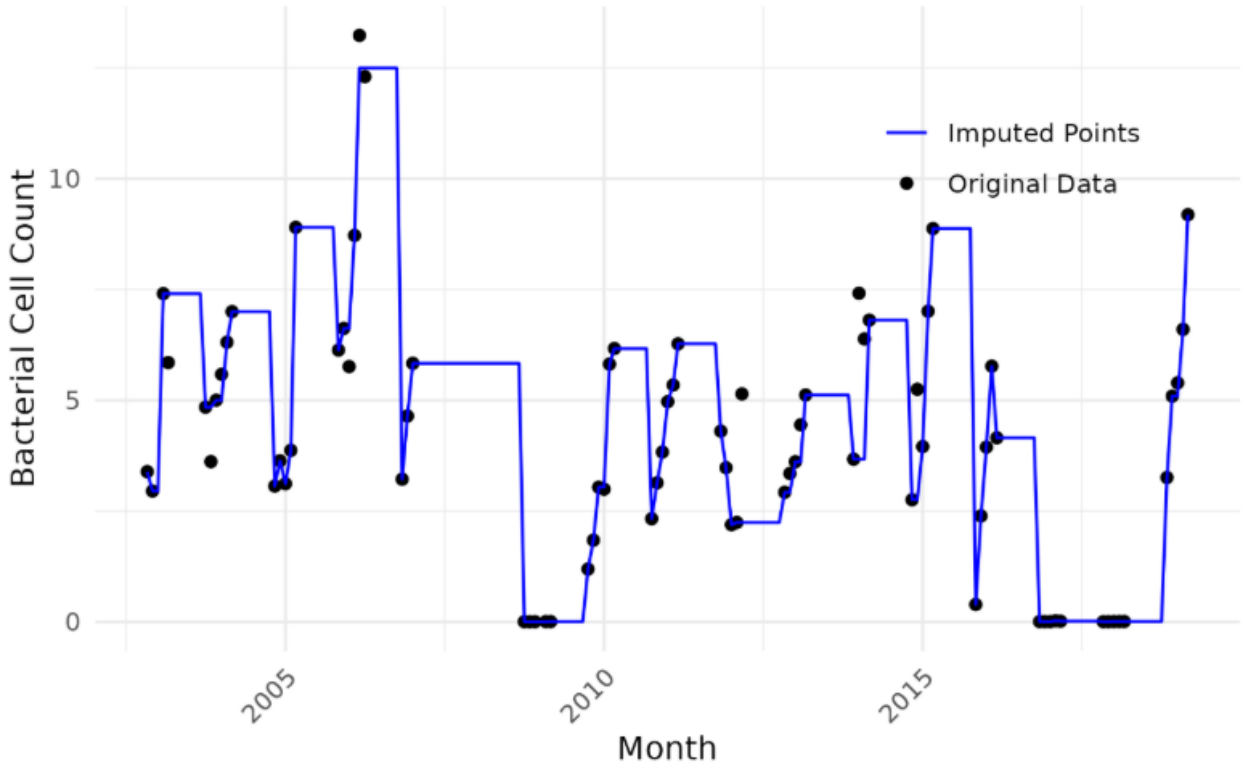

Figure 4: Effect of LOCF imputation in time series imputation

LOCF (Last Observation Carried Forward) fills in missing values by repeating the last known value until a new one appears. This creates flat lines in the graph and makes the data look better than the original but also unrealistic, especially when so many values are the same and some It doesn't account for trends and doesn't seem to work that well for a time series with this many missing values.

Mean value

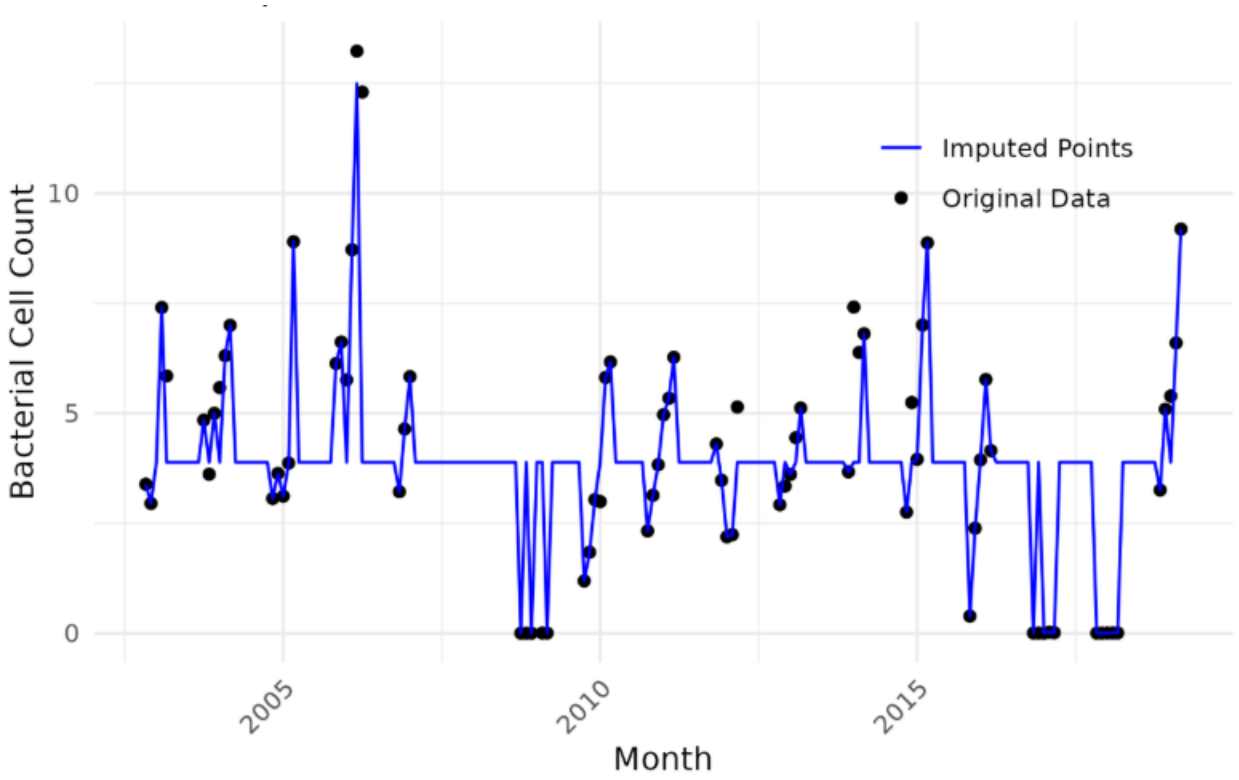

Figure 5: Effect of mean value imputation in time series imputation.

It's very clear in this graph what the imputed points are because they all are the same, the mean of the train subset. It doesn't follow any trend or take in consideration what values are around the missing value. This doesn't create any trend at all and makes the original data points stick out from the imputed values instead of the imputed values bringing the data together.

Linear filtering

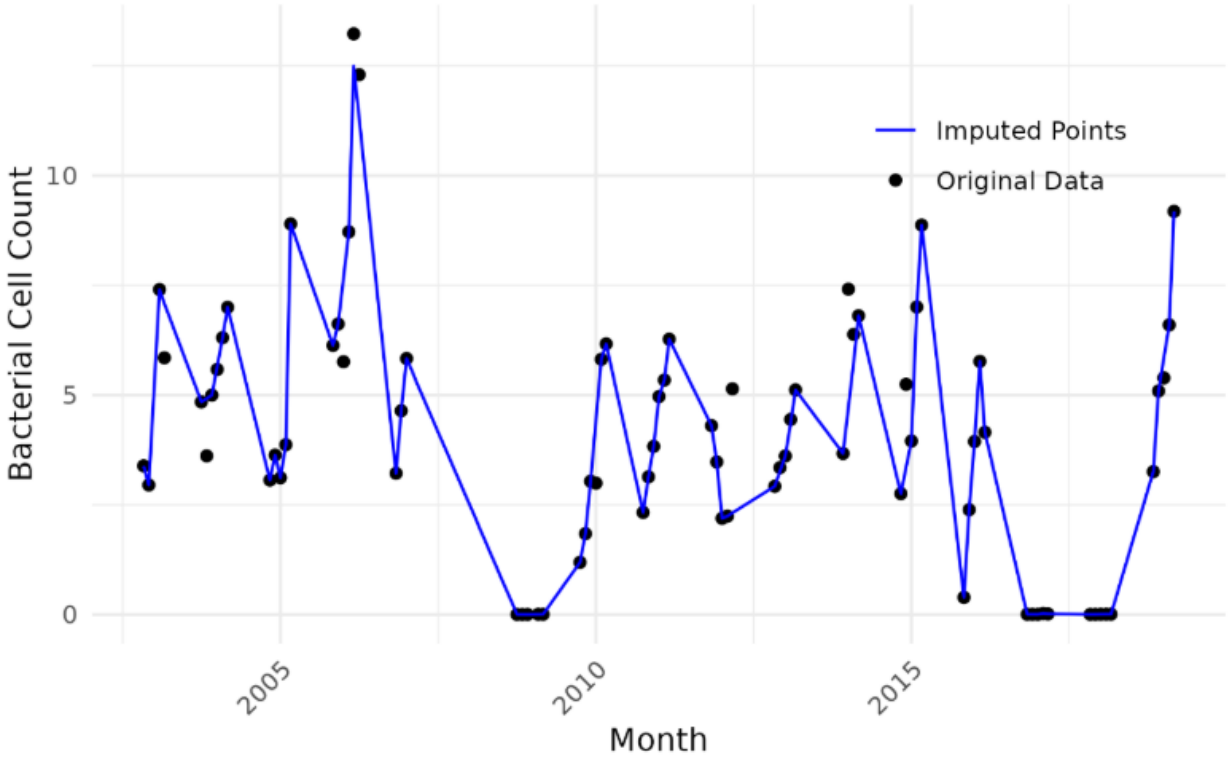

Figure 6: Effect of linear imputation in time series imputation.

This method follows the smaller gaps well and looks realistic when there are only one or a few months that are imputed. When it comes to the bigger gaps, there isn't anything to smoothen it out or account for trends. In theory it's a good method because the idea is how it constantly increases or decreases between known points. For this time series I'd say that it's decent for the small gaps, but not ideal when larger gaps of data are missing.

Moving average

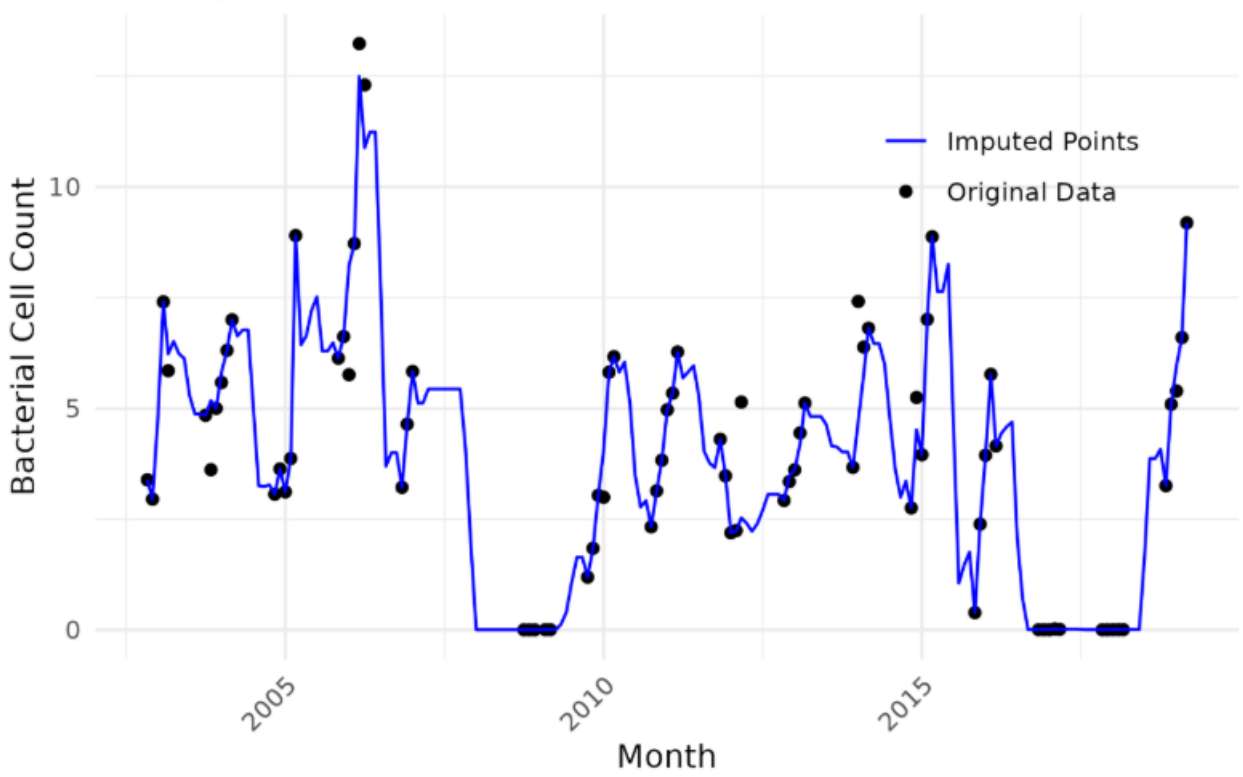

Figure 7: Effect of moving average imputation in time series imputation.

Moving Average is similar to K-Nearest Neighbor and does a good job of following the overall trend. It smooths out the data and handles the smaller gaps well, this makes the imputed values look more realistic than in earlier methods. However, by taking the average of surrounding values it can hide sudden spikes or drops in bacterial growth. It still gives the graph a cleaner and more realistic graph than some earlier methods.

(k)-Nearest neighbor

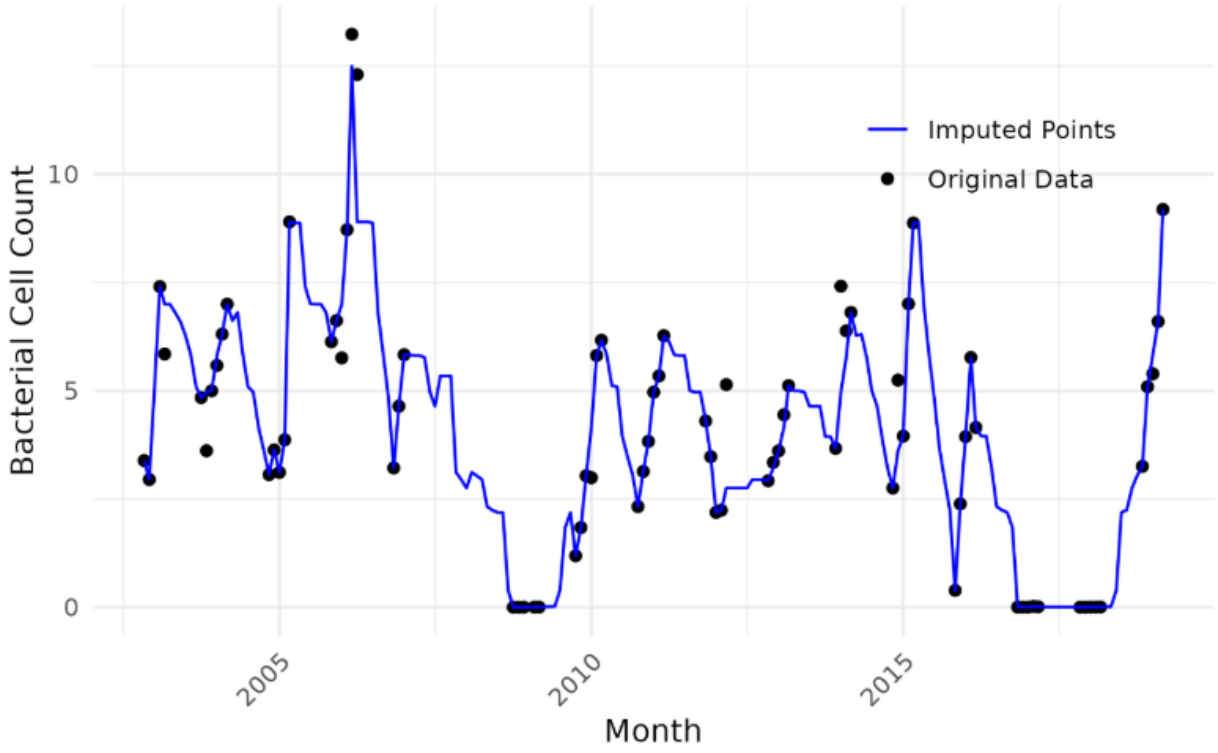

Figure 8: Effect of (k)-nearest neighbor imputation in time series imputation.

I chose to put k=5. The data seems to follow the original data in the smaller gaps, which point towards that this method is good at keeping the original trend in the imputed data. Even for the bigger gaps, the imputation follows the data around the gaps in a good way. I can also see how even if one already existing value spike, it doesn't necessarily mean that the next value has to spike, unlike some other imputation methods like LOCF do. With the imputed data both following the original data trends and filling in the larger gaps more effectively, it makes the graph more reliable and realistic than the earlier imputation methods.

Seasonal splitting imputation

Seasonal splitting imputation wasn't able to perform any imputation on this dataset due to the technique being unable to find a consistent seasonal trend. Since this technique divides the data

into groups based on months, and with this dataset missing a fair amount of values, the algorithm couldn't find any seasonality. This resulted in the algorithm not being able to impute any numbers for this dataset.

Seasonal decomposition imputation

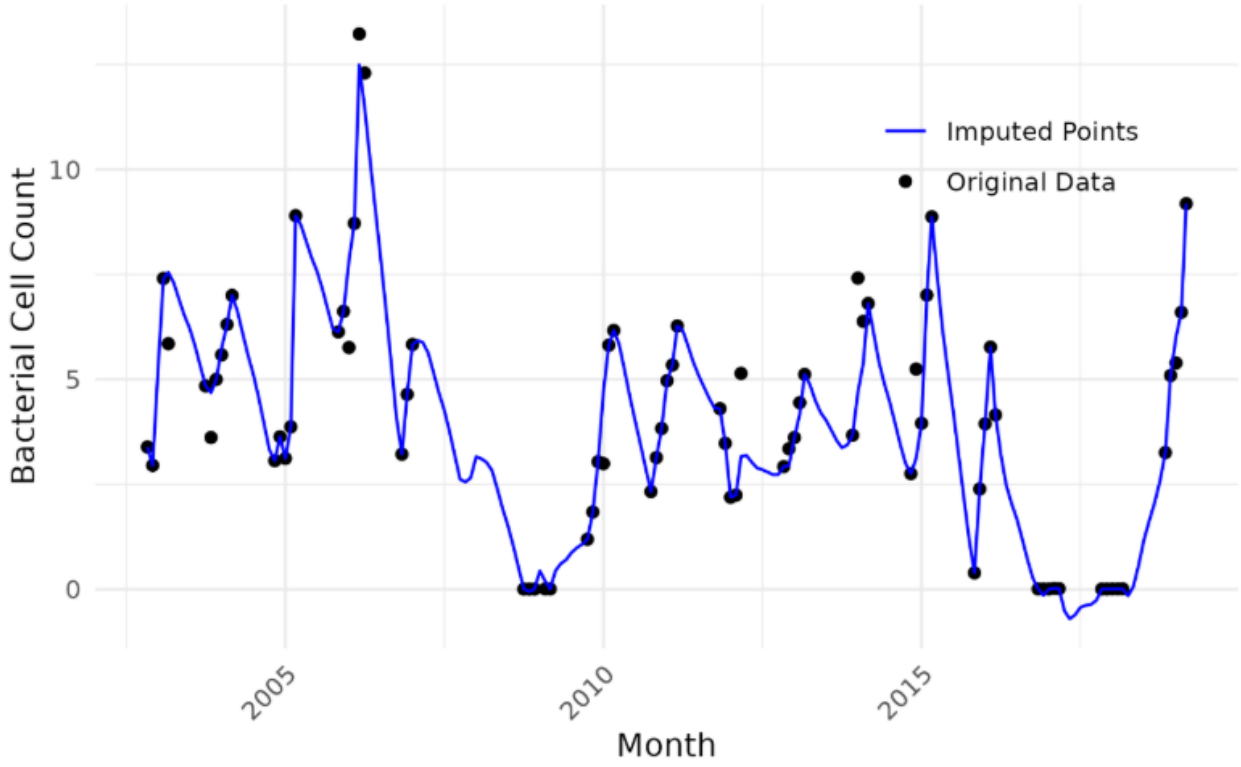

Figure 9: Effect of seasonal decomposition in time series imputation.

With the seasonal decomposition method, it doesn't do much in the smaller gaps. Looks a lot like the linear imputation technique in the smaller gaps. But when it comes to the larger gap around 2008, it shows a little increase like there are most years. So instead of skipping 2008 and just decreasing until the known values in 2009, it made a little transition between the years which

make it look realistic. However this won't be recognized in the RMSE and MAE calculations as no values from the "test" subset are in that gap.

Kalman filtering

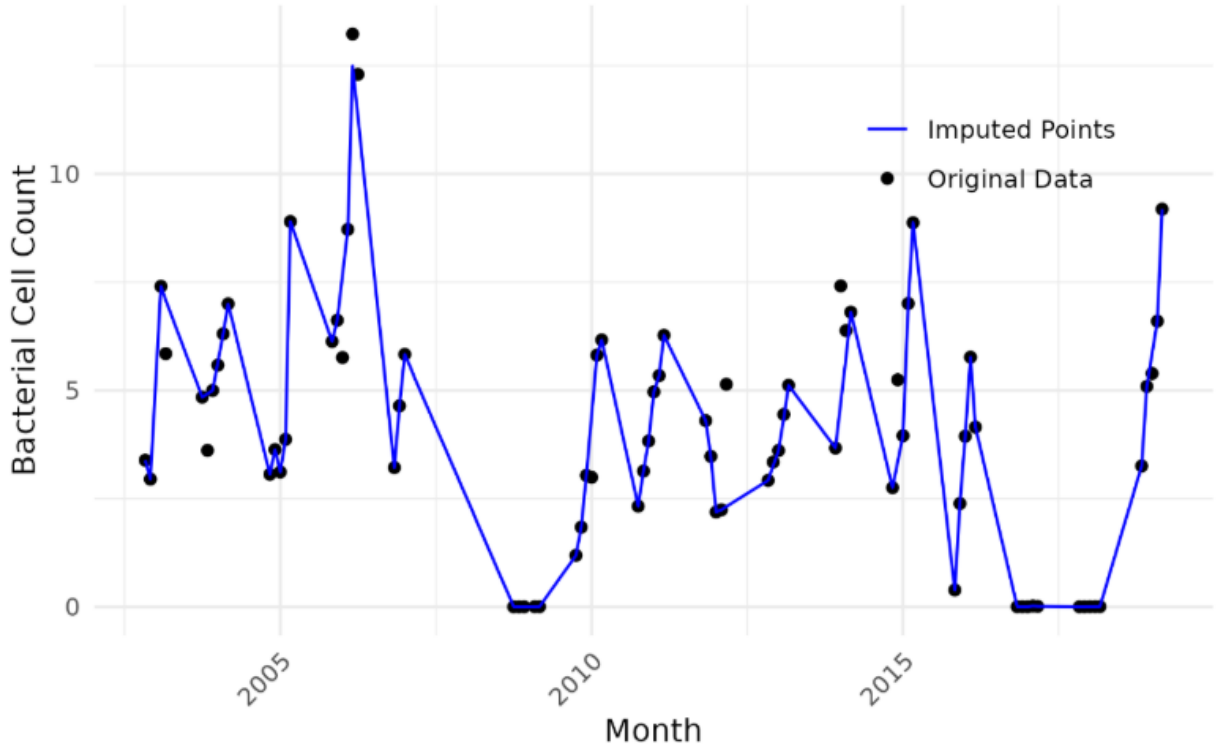

Figure 10: Effect of Kalman filtering in time series imputation.

Ended up being the same graph as the linear imputation method.

RMSE & MAE:

To calculate how well each imputation method performs, we use RMSE (Root Mean Square Error) and MAE (Mean Absolute Error). RMSE gives more weight to large errors by squaring the difference, while MAE treats all errors equally. In both RMSE and MAE, it shows that an imputation method did a better job the lower the value is. [14]

For these formulas we have $\hat{y}_i$ representing the predicted value for the i^{th} data point, y_i is the actual observed value at the i^{th} data point, and n is the total number of observations.

RMSE formula: $$\frac{1}{n}\sum_{i=1}^{n}\left|y_i - \widehat{y_i}\right| \quad (6)$$

MAE formula: $$\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \widehat{y_i}\right)^2} \quad (7)$$

Method	**RMSE**	**MAE**
Moving average	1.30	0.92
Linear imputation	1.35	0.97
Kalman filter	1.35	0.97
Seasonal decomposition	1.36	1.07
kNN	1.39	1.00
LOCF	1.62	1.12
Mean imputation	3.08	2.37
Random imputation	5.57	4.84

The RMSE and MAE results show that the moving average imputation technique performed best overall, with having both lowest RMSE and MAE values. Linear imputation, Kalman filter, and seasonal decomposition also showed relatively low errors, making them

reasonable options. Random imputation was the least accurate by far, with the highest RMSE and MAE. But methods like LOCF and mean value imputation also had higher error values, which isn't a surprise with those three having simpler ideas than some other imputation methods used in this project.

Chapter III: Conclusion

Summary:

The goal of this study is to find what of the eight used imputation methods imputes this dataset of bacterial growth trends the best. The data was collected in the Antarctic by the Palmer Long Term Ecological Research (LTER) program. Every one of these eight imputation methods have their own way to estimate missing values while still trying to keep the overall trend in the time series. From the beginning we had some problems in this dataset, which were missing a large amount of values and especially from April to October most years and even most values 2007-2009 and 2017-2018.

These imputation methods were applied using the ImputeTS and VIM packages in RStudio. I chose a mix of different imputation techniques, some that work in similar ways to estimate the missing values and others that use different approaches. This was done to make sure we tested a wide range of methods and found the best one for this time series data. To determine how well the imputation handled the missing data, I used the train-test method to see how well the imputation methods followed the subset of the test values. I calculated this by using RMSE (root mean square error) and MAE (mean average error).

All imputation methods had their positives and negatives, but some clearly outperform others. Based on the train-test method and the RMSE and MAE results, moving average was the most accurate overall. It imputed the best values that were the closest to the test value subset and it performed better than simpler methods like LOCF and random imputation. Linear imputation and Kalman filtering also showed good results by impute values that follow the original data in the test subset. Seasonal decomposition made the time series look like I expected bacterial growth to look like but didn't perform as well on the train-test method. Simpler methods like mean imputation, LOCF, and especially random imputation had the highest errors and often ignored the structure of the original data.

To summarize, moving average ended up being the best imputation method of the eighth applied in the project with linear imputation and Kalman filtering also performing well. Simpler methods like mean and random imputation were less reliable, often failing to preserve the original trends in the time series.

Suggestions for Further Study:

One thing this study could do in a further study is to look into if it would be better to mix some of the imputation methods. For example, use one method to fill in the bigger gaps and the another to smooth the whole time series. Both of the seasonal imputations (Seasplit and Seasonal Decomposition) could possibly work better if another imputation method is applied to the time series before applying them.

Another thing is to apply the train-test method more times to not use the same subset for training and the same for tests every time. By doing this multiple times it would provide a more overall view of each method's performance and help the result to possibly be less biased.

Lastly, I could also bring in other covariates within the imputation algorithms. By bringing in other variables like temperature or depth in the water, it could possibly help the imputation methods make a more calculated prediction which would lead to better imputations.

Work Cited

[1] Time series analysis: Definition, types, techniques, and when it's used. Tableau. (n.d.). https://www.tableau.com/analytics/what-is-time-series-analysis

[2] Abulkhair, A. (2023, June 13). Data imputation demystified: Time Series Data. Medium. https://medium.com/@aaabulkhair/data-imputation-demystified-time-series-data-69bc9c798cb7

[3] Liu, Y., & De, A. (2015). *Multiple imputation by fully conditional specification for dealing with missing data in a large epidemiologic study*. International journal of statistics in medical research. https://pmc.ncbi.nlm.nih.gov/articles/PMC4945131/#S2

[4] Palmer LTER Datasets. Palmer Station LTER. (n.d.). https://pallter.marine.rutgers.edu/data/

[5] Palmer LTER. (n.d.). Station bacteria dataset. Palmer Long Term Ecological Research. Retrieved from https://pallter-data.marine.rutgers.edu/erddap/info/StationBacteria/index.html

[6] Moritz S, Bartz-Beielstein T (2017). “imputeTS: Time Series Missing Value Imputation in R.” _The R Journal_, *9*(1), 207-218. doi:10.32614/RJ-2017-009 <https://doi.org/10.32614/RJ-2017-009>

[7] Alexander Kowarik, Matthias Templ (2016). Imputation with the R Package VIM. Journal of Statistical Software, 74(7), 1-16. doi:10.18637/jss.v074.i07[5]

[8] Ducklow, H. W., Fraser, W. R., Meredith, M. P., Stammerjohn, S. E., & Doney, S. C. (2015, October 2). West Antarctic Peninsula: An ice-dependent coastal marine ecosystem in transition. Oceanography. https://doi.org/10.5670/oceanog.2013.62

[9] Moritz, S. (n.d.). Missing value imputation by weighted moving average. R. https://search.r-project.org/CRAN/refmans/imputeTS/html/na_ma.html

[10] Lacey, T. (n.d.). *Tutorial: The kalman filter*. web.mit. https://web.mit.edu/kirtley/kirtley/binlustuff/literature/control/Kalman%20filter.pdf

[11] RStudio Team. (2023). *RStudio: Integrated Development Environment for R*. RStudio, PBC. http://www.rstudio.com/

[12] Evans JS, Murphy MA (2023). _spatialEco_. R package version 2.0-2, <https://github.com/jeffreyevans/spatialEco>.

[13] H. Wickham. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York, 2016.

[14] Whitehead, A., & Jeon, C. (n.d.). Handling data gaps in time series using imputation ... Klick Health. https://conferences.oreilly.com/strata/strata-ny-2019/cdn.oreillystatic.com/en/assets/1/event/300/Handling%20data%20gaps%20in%20time%20series%20using%20imputation%20Presentation.pdf