The Explainability of Time Series Downsampling

A Preprint

Morgan Frodsham
School of Computing
Newcastle University
Newcastle upon Tyne, UK
M.C.M.Frodsham2@newcastle.ac.uk

Matthew Forshaw
School of Computing
Newcastle University
Newcastle upon Tyne, UK
matthew.forshaw@newcastle.ac.uk

August 7, 2023

Abstract

Enter the text of your abstract here.

***K*eywords** blah · blee · bloo · these are optional and can be removed

1 INTRODUCTION

HM Government is committed to making data-driven decisions that engender public trust [1]–[4]. Data-driven decisions are considered to be "more well-informed" [1], effective [4], consistent [3], and better "at scale" [2]. Despite this, there is a lack of trust in government use of data [5]. This suggests that public trust in data-driven decisions goes beyond how the "data complies with legal, regulatory and ethical obligations" [3]. The UK public need to have "confidence and trust in how data, including personal data, is used" [2], and this requires transparency [5].

To make data-driven decisions, government decision-makers also need to trust how the data used (cite user research here). This means trusting which data points are selected, how this data collected and stored, and the capability of data practitioners to understand the quality, insights and limitations of it. At every stage of the data processing pipeline, data practitioners have the opportunity to communicate the impact of the assumptions and choices they are making to support decision-makers in trusting the data informing their decisions.

Time series data is used across HM Government [6] to inform decision-makers across various domains [7]. It is also widely generated and used by industry and research [8]. The volume of time series data is continuously increasingly [9], posing significant challenges for handling and visualising this popular data type [8]. Data practitioners must utilise methods that reduce data volumes to align with limitations like processing time, computing costs, storage capabilities, and sustainability ambitions [8], [10], [11].

Downsampling is an established technique [12], [13] that involves selecting a representative subset of data to preserve its original shape while reducing the number of data points [9], [14]. This is a vital part of making voluminous time series understandable for human observation [10] and an essential step in many time series database solutions [9]. However, little attention has been devoted to how downsampling impacts decision-makers trust in the data.

Despite widespread use, how to communicate the impact of downsampling algorithms on time series data remains also understudied [9], [10]. Downsampling expands the boundaries of risk for decision-makers as data practitioners may not realise the significance of the data being discarded. Such choices throughout the data pipeline may have disproportionately larger consequences later as their ramifications for future decisions are not fully understood by all. It is important, therefore, that data practitioners are able to communicate the impact of choices made throughout the data pipeline.

To address these challenges, this paper shares initial insights from user research on the impact of downsampling on decision-makers' trust in data and suggests a visualisation methodology for communicating the impact of downsampling algorithms on time series. This methodology combines user research with R packages `imputeTS` [15] and `Rcatch22` [16] to identify and visualise time series features that are most sensitive to downsampling. It is hoped this will improve decision-makers' trust in data by helping data practitioners to create transparency in the data processing pipeline, communicate the impact of downsampling, and support conversations about which algorithms or parameters are most appropriate for particular decision-maker use cases.

2 RELATED WORK

This section provides an overview of previous related work to create a clear understanding of the most relevant fields of research and identify the gaps being addressed by the paper.

##Data Transparency

Technology transparency, "including institutional data practices", is sociopolitical in nature [17]. There is a growing number of researchers reflecting on "societal needs in terms of what is made transparent, for whom, how, when and in what ways, and, crucially, who decides" [18].

The implicit assumption behind calls for transparency is that "seeing a phenomenon creates opportunities and obligations to make it accountable and thus to change it" [19]. However, without sufficient agency to explore the information being shared, seeing a phenomenon often results in "information overload" [20] that obfuscates or diverts [21]. Without agency, transparency is increasingly considered to be a fallacy [22].

Meaningful transparency is only realised when the information is provided with the tools to turn "access to agency" [19], [22]. This suggests that data practitioners communicating the assumptions and choices made throughout the data processing pipeline with decision-makers is not likely to create trust in how the data is used. Instead, data practitioners should be encouraged to find tools, such as interactive visualisations [9], that put agency into the hands of decision-makers.

##Time series visualisation

Time series data is commonly visualised as a line graph [10], [23]. Line graphs help the human eye to observe only the most important data points [10] by convey the overall shape and complexity of the time series data [9], [12]. The most effective time series visualisations are, however, interactive [23], [24], turning access into agency [22] by allowing the user to access details on demand. Evaluation of time series visualisation is, therefore, a growing field of research [23].

However, this growing field of research does not extend to visualisations of choices and assumptions made during data processing pipline. Indeed, such visualisations are a side effect of the research. This dynamic is exemplified by the R package `imputeTS` [15] where the impact of imputation choices made by the user is only visualised to support the user through the complete process of replacing missing values in time series [25]. The research set out in this paper harnesses the capabilities of `imputeTS` and its 'process' visualisations to help data practitioners communicate the impact of downsampling choices made in the data processing pipeline.

##Value Preserving Data Aggregation

Technological innovation has generated unprecedented amount of time series data and this data continues to grow [2], [8], [26], [27]. For example, tackling climate change is the UK Government's "number one international priority" [28], yet climate simulations that help inform decision-makers generate tens of terabytes per second [8], [29]. Downsampling (value preserving data aggregation) plays an important role in addressing how this voluminous data is processed, stored [8] and visualised [10], [30] by minimising computing resources needed [8], reducing network latency, and improving rendering time [9], [14].

An overview of commonly used downsampling (value preserving data aggregation) algorithms is provided in the table below:

insert table [9] - EveryNth, also known as sampling or decimation, selects n^{th} datapoint [31] - percentage change - Mode-Median Bucket. The bucket part in the algorithm name refers to the data being split up into buckets, each containing approximately equal number of data points. The algorithm then finds one data point within each bucket as follows. If there is a single y-value which has the highest frequency (the mode) then the leftmost corresponding data point is selected. If no such data point exists a point corrosponding to the median of the y-values is selected.@Sveinn - Min-Std-Error-Bucket [10] It is based on linear regression and uses the formula for the standard error of the estimate (SEE) to downsample data. The SEE is a measure of the accuracy of predictions made with a regression line when a linear least squares technique is applied. The greater the error the more discrepancy between the line and the data points. - MinMax preserves the minimum and maximum of every data bucket [14] - OM^3 maintains minimum and maximum values at every time interval that is used to rasterize a pixel column in the display window [32] - M4 combines EveryNth and MinMax, selecting the first and last values of each data bucket as well as its minimum and maximum [30], [33] - Longest line bucket [10] very similar to the MSEB algorithm but with some key differences. It starts off exactly the same, splitting the data points into buckets and calculating lines going through all the points in one bucket and all the points in the next bucket as was shown in figure 2.3 on page 8. The main difference is that instead of calculating the standard error for each line segment it simply calculates its length (Euclidean distance between the two points defining the line). - Largest-Triangle One-Bucket (LTOB) First all the points are ranked by calculating their effective areas. Points with effective areas as null are excluded. The data points are then split up into approximately equal number of buckets as the specified downsample threshold. Finally, one point with the highest rank (largest effective area) is selected to represent each bucket in the downsampled data. [10] - Largest-Triangle-Dynamic [10]

- Largest Triangle Three Buckets LTTB selects the data point that forms the largest triangular surface between the previously selected data point and the next data bucket's average value [14]
- MinMaxLTTB preselects data using MinMax before applying LTTB on the selected datapoints [14]

Data practitioners have made recent advances in the performance and evaluation of downsampling approaches [9], [12]–[14], [24], [30]–[32]. These advances focus on the effectiveness of the algorithm in delivering downsampled data that represents the original data as accurately as possible. This is vital part of enabling and improving data-driven decision-making, but is focused on supporting data practitioners in their analysis of the data. Instead, the research set out in this paper aims to support data practitioners to communicate the impact of their downsampling choices for decision-makers.

##Time series feature analysis

The increasing size of modern time series data sets has also generated new research into the dynamical characteristics of time series [34]. These characteristics are often used to identify features that enable efficient clustering and classification of time series data, especially for machine learning. A comprehensive library of such features is the *hctsa* (highly comparative time series analysis) toolbox. This shares the 4791 best performing features after computationally comparing thousands of features from across scientific time series analysis literature [35].

Utilising such a library, however, is computationally expensive [34]. C. H. Lubba et. al have attempted to address this by identified a subset of 22 features that are tailored to time series data mining tasks [34], [36]. Although further research is needed to evaluate the relative performance of different feature sets on different types of problems, `catch22` performs well against other feature libraries across 800 diverse real-world and model-generated time series [37].

Features used to classify time series data could provide a common framework by which to consistenly compare different downsampling algorithms and parameters. The research set out in this paper utilises the `Rcatch22` subset of features to explore impact of downsampling and create a visual tool for explaining this impact.

3 MOTIVATION

- user research insights

4 METHODOLOGY

- Two basic downsampling approaches applied Annotated Change Turing Data sets
- Used imputeTS to calculate number of nas, na gaps and remaining volume of data as well as visualise gaps
- Used imputTS to apply simple linear interpolation
- Ran Rcatch22 over original downsampled and interpolated data
- Calculate difference from original data to choose subselection
- visualise variation of method over features
- visualise variation of data loss over features

You can use equation in blocks

$$\xi_{ij}(t) = P(x_t = i, x_{t+1} = j|y, v, w; \theta) = \frac{\alpha_i(t)a_{ij}^{w_t}\beta_j(t+1)b_j^{v_{t+1}}(y_{t+1})}{\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i(t)a_{ij}^{w_t}\beta_j(t+1)b_j^{v_{t+1}}(y_{t+1})}$$

But also inline i.e $z = x + y$

4.1 ImputeTS

4.2 Rcatch22

4.3 Downsamplng Impat

4.4 User Research

5 RESULTS AND EVALUATION

6 FUTURE WORK

The data pipeline developed by C. H. Lubba et. al could be used to generate other subsets of time series features for distinct tasks in any domain. This is likely to be important for highly specialised tasks or domains.

7 CONCLUSION

8 REFERENCES

9 Examples of citations, figures, tables, references

You can insert references. Here is some text **kour2014real?**, **kour2014fast?** and see **hadash2018estimate?**.

The documentation for `natbib` may be found at

You can use custom blocks with LaTeX support from **rmarkdown** to create environment.

`http://mirrors.ctan.org/macros/latex/contrib/natbib/natnotes.pdf%7D`

Of note is the command `\citet`, which produces citations appropriate for use in inline text.

You can insert LaTeX environment directly too.

```
\citet{hasselmo} investigated\dots
```

produces

> Hasselmo, et al. (1995) investigated...

`https://www.ctan.org/pkg/booktabs`

9.1 Figures

You can insert figure using LaTeX directly.

See Figure 1. Here is how you add footnotes. [^Sample of the first footnote.]

But you can also do that using R.

```r
plot(mtcars$mpg)
```

You can use **bookdown** to allow references for Tables and Figures.

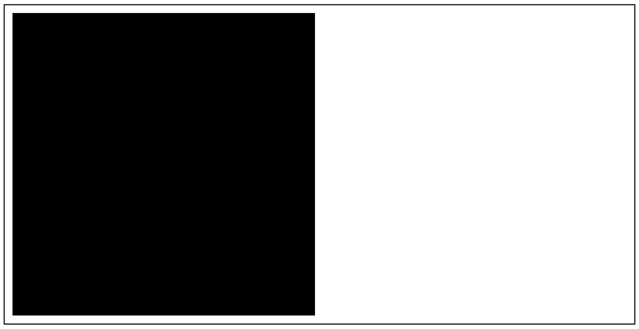

Figure 1: Sample figure caption.

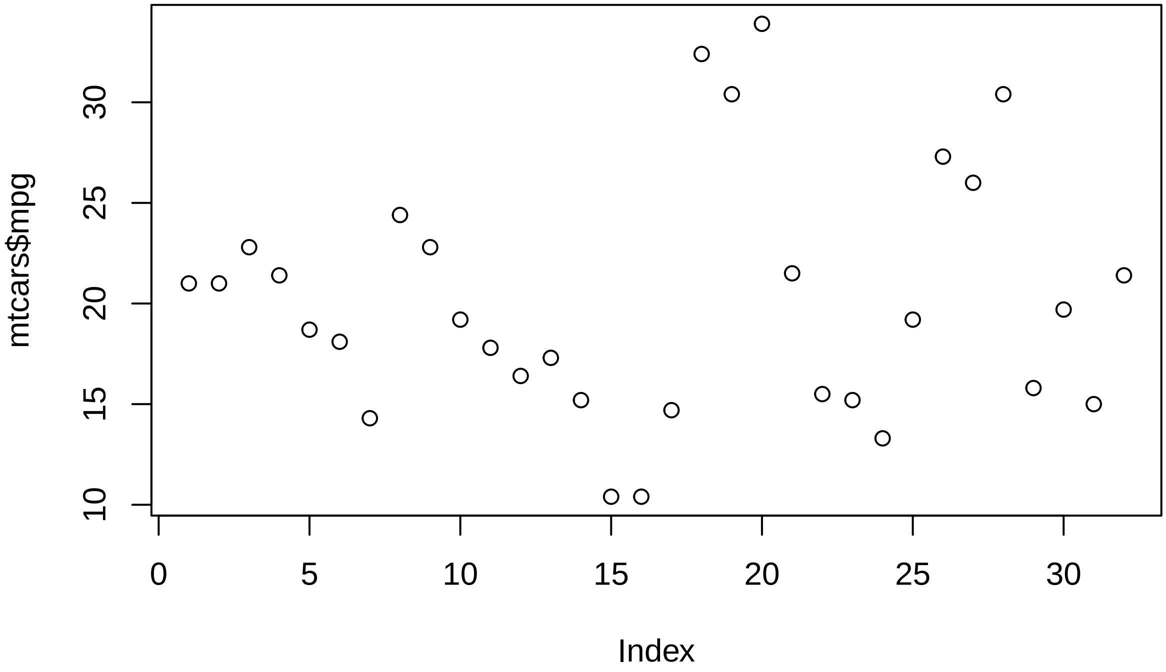

Figure 2: Another sample figure

Table 1: Sample table title

Part		
Name	Description	Size (μm)
Dendrite	Input terminal	$\sim$100
Axon	Output terminal	$\sim$10
Soma	Cell body	up to 10^6

9.2 Tables

Below we can see how to use tables.

See awesome Table~1 which is written directly in LaTeX in source Rmd file.

You can also use R code for that.

```r
knitr::kable(head(mtcars), caption = "Head of mtcars table")
```

Table 2: Head of mtcars table

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21.0	6	160	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag	21.0	6	160	110	3.90	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	18.7	8	360	175	3.15	3.440	17.02	0	0	3	2
Valiant	18.1	6	225	105	2.76	3.460	20.22	1	0	3	1

9.3 Lists

- Item 1
- Item 2
- Item 3

[1] Cabinet Office and Government Digital Service, "Government transformation strategy: Better use of data." HM Government; https://www.gov.uk/government/publications/government-transformation-strategy-2017-to-2020/government-transformation-strategy-better-use-of-data, 2017.

[2] Department for Digital, Culture, Media & Sport and Department for Science, Innovation & Technology, "National data strategy." HM Government; https://www.gov.uk/government/publications/uk-national-data-strategy/national-data-strategy, 2020.

[3] M. of Defence, "Data strategy for defence," *GOV.UK*. HM Government; https://www.gov.uk/government/publications/data-strategy-for-defence/data-strategy-for-defence, 2021.

[4] Central Digital & Data Office, "Transforming for a digital future: 2022 to 2025 roadmap for digital and data." HM Government; https://www.gov.uk/government/publications/roadmap-for-digital-and-data-2022-to-2025/transforming-for-a-digital-future-2022-to-2025-roadmap-for-digital-and-data, 2022.

[5] Centre for Data Ethics & Innovation, "Addressing trust in public sector data use." https://www.gov.uk/government/publications/cdei-publishes-its-first-report-on-public-sector-data-sharing/addressing-trust-in-public-sector-data-use#introduction--context.

[6] Government Analysis Function, "Types of data in government learning pathway." https://analysisfunction.civilservice.gov.uk/learning-development/learning-pathways/types-of-data-in-government-learning-pathway/, 2022.

[7] Office for National Statistics, "Time series explorer." https://www.ons.gov.uk/timeseriestool?query=&topic=&updated=&fromDateDay=&fromDateMonth=&fromDateYear=&toDateDay=&toDateMonth=&toDateYear=&size=50, Unknown.

[8] Y. An, Y. Su, Y. Zhu, and J. Wang, "TVStore: Automatically bounding time series storage via time-varying compression," in *Proceedings of the 20th USENIX conference on file and storage technologies*, in USENIX conference on file and STorage technologies. Santa Clara, CA, USA: USENIX Association, 2022, pp. 83–99.

[9] J. Donckt, J. Donckt, M. Rademaker, and S. Hoecke, "Data point selection for line chart visualization: Methodological assessment and evidence-based guidelines." 2023. doi: 10.48550/arXiv.2304.00900.

[10] S. Steinarsson, "Downsampling time series for visual representation." University of Iceland, Faculty of Industrial Engineering, Mechanical Engineering; Computer Science, School of Engineering; Natural Sciences, University of Iceland, Reykjavik, Iceland, 2013.

[11] The Shift Project, "Implementing digital sufficiency," 2020.

[12] W. Aigner, S. Miksch, W. Muller, H. Schumann, and C. Tominski, "Visual methods for analyzing time-oriented data," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 1, pp. 47–60, 2008, doi: 10.1109/TVCG.2007.70415.

[13] B. C. Kwon, J. Verma, P. J. Haas, and C. Demiralp, "Sampling for scalable visual analytics," *IEEE Computer Graphics and Applications*, vol. 37, no. 1, pp. 100–108, 2017, doi: 10.1109/MCG.2017.6.

[14] J. Donckt, J. Donckt, M. Rademaker, and S. Hoecke, "MinMaxLTTB: Leveraging MinMax-preselection to scale LTTB." 2023. Available: https://arxiv.org/abs/2305.00332

[15] S. Moritz and T. Bartiz-Beielstein, "imputeTS: Time series missing value imputation in r," vol. 9.1. R Journal, 2017. doi: 10.32614/RJ-2017-009.

[16] C. H. Lubba, B. Fulcher, T. Henderspn, B. Harris, O. r. TL, and O. Cliff, "catch22: CAnonical time-series CHaracteristics." R Journal, 2022. doi: 10.5281/zenodo.6673597.

[17] K. E. Levy and D. M. Johns, "When open data is a trojan horse: The weaponization of transparency in science and governance," *Big Data & Society*, vol. 3, no. 1, 2016, doi: 10.1177/2053951715621568.

[18] J. Bates, H. Kennedy, I. Medina Perea, S. Oman, and L. Pinney, "Socially meaningful transparency in data-based systems: Reflections and proposals from practice," *Journal of Documentation*, vol. ahead–of–print, 2023, doi: 10.1108/JD-01-2023-0006.

[19] M. Ananny and K. Crawford, "Seeing without knowing: Limitations of the transparency ideal and its application to algorithmic accountability," *New Media & Society*, vol. 20, no. 3, pp. 973–989, 2018, doi: 10.1177/1461444816676645.

[20] R. Matheus, M. Janssen, and T. Janowski, "Design principles for creating digital transparency in government," *Government Information Quarterly*, vol. 38, no. 1, 2021, doi: https://doi.org/10.1016/j.giq.2020.101550.

[21] N. A. Draper and J. Turow, "The corporate cultivation of digital resignation," *New Media & Society*, vol. 21, no. 8, pp. 1824–1839, 2019, doi: 10.1177/1461444819833331.

[22] J. A. Obar, "Sunlight alone is not a disinfectant: Consent and the futility of opening big data black boxes (without assistance)," *Big Data & Society*, vol. 7, no. 1, 2020, doi: 10.1177/2053951720935615.

[23] J. Walker, R. Borgo, and MW. Jones, "TimeNotes: A study on effective chart visualization and interaction techniques for time-series data," *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, 2016, doi: 10.1109/TVCG.2015.2467751.

[24] J. Donckt, J. Donckt, E. Deprost, and S. Hoecke, "Plotly-resampler: Effective visual analytics for large time series," *IEEE Visualization and Visual Analytics*, 2022, doi: 10.1109/VIS54862.2022.00013.

[25] S. Moritz and T. Bartiz-Beielstein, "imputeTS: Time series missing value imputation in r." R Journal, 2017. Available: https://cran.r-project.org/web/packages/imputeTS/vignettes/imputeTS-Time-Series-Missing-Value-Imputation-in-R.pdf

[26] A. Visheratin *et al.*, "Peregreen – modular database for efficient storage of historical time series in cloud environments," in *2020 USENIX annual technical conference (USENIX ATC 20)*, USENIX Association, 2020, pp. 589–601. Available: https://www.usenix.org/conference/atc20/presentation/visheratin

[27] T. Schlossnagle, J. Sheehy, and C. McCubbin, “Always-on time-series database: Keeping up where there’s no way to catch up,” *Commun. ACM*, vol. 64, no. 7, pp. 50–56, 2021, Available: https://doi.org/10.1145/3442518

[28] HM Government, “Global britain in a competitive age: The integrated review of security, defence, development and foreign policy.” GOV.UK, 2021. Available: https://www.gov.uk/government/publications/
global-britain-in-a-competitive-age-the-integrated-review-of-security-defence-development-and-fo
global-britain-in-a-competitive-age-the-integrated-review-of-security-defence-development-and-fo

[29] I. Foster *et al.*, “Computing just what you need: Online data analysis and reduction at extreme scales,” in *Euro-par 2017*, F. F. Rivera, T. F. Pena, and J. C. Cabaleiro, Eds., in Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics). Germany: Springer Verlag, 2017, pp. 3–19. doi: 10.1007/978-3-319-64203-1_1.

[30] A. Kohn, D. Moritz, and T. Neumann, “DashQL – complete analysis workflows with SQL.” 2023. doi: 10.48550/arXiv.2306.03714.

[31] U. Jugel, Z. Jerzak, G. Hackenbroic, and V. Markl, “VDDA: Automatic visualization-driven data aggregation in relational databases,” *The VLDB Journal*, vol. 25, 2016, doi: 10.1007/s00778-015-0396-z.

[32] W. Yunhai *et al.*, “OM3: An ordered multi-level min-max representation for interactive progressive visualization of time series,” in *Proc. ACM manag. data*, ACM, 2023. Available: https://doi.org/10.1145/3589290

[33] U. Jugel, Z. Jerzak, G. Hackenbroich, and V. Markl, “M4: A visualization-oriented time series data aggregation. Proceedings of the VLDB endowment,” vol. 7, 2014, Available: https://www.vldb.org/2014/program/http://www.vldb.org/pvldb/vol7/p797-jugel.pdf

[34] C. H. Lubba, S. S. Sarab, P. Knaute, S. R. Schultz, B. D. Fulcher, and N. S. Jones, “catch22: CAnonical time-series CHaracteristics,” *Data Mining and Knowledge Discovery*, vol. 33, 2019, doi: 10.1007/s10618-019-00647-x.

[35] B. D. Fulcher and N. S. Jones, “Highly comparative feature-based time-series classification,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 12, pp. 3026–3037, 2014, doi: 10.1109/TKDE.2014.2316504.

[36] A. Bagnall, J. Lines, A. Bostrom, J. Large, and E. Keogh, “The great time series classification bake off: A review and experimental evaluation of recent algorithmic advances,” *Data Mining and Knowledge Discovery*, vol. 31, 2017, doi: 10.1007/s10618-016-0483-9.

[37] T. Henderson and B. D. Fulcher, “An empirical evaluation of time-series feature sets,” in *2021 international conference on data mining workshops (ICDMW)*, 2021, pp. 1032–1038. doi: 10.1109/ICDMW53433.2021.00134.