Dissertation

[TODO: create cover, add boilerplate stuff]

Acknowledgements

[TODO: write]

Table of Contents

[TODO: fix which sections are included]

Table of contents

Samenvatting

[TODO: write]

Introduction

What is this dissertation about?

- missing data & imputation are widespread
- imputation = filling in missing values
- imputation is easy with software defaults (e.g. in `mice`, van Buuren & Groothuis-Oudshoorn, 2011)
- good imputation is very hard still
- how do we know if we have a good imputation method?
- in this dissertation I study computational evaluation of imputation methodology
- computational evaluation at different levels:
  - building imp models (`ggmice`),
  - assessing imp models (`ggmice` and convergence),
  - developing new imp models/methods (evaluation paper)

What is computational evaluation of imputation methodology?

Computational evaluation

- computational evaluation =/= statistical evaluation
- models can be statistically valid, but not fitting
- computational evaluation is "to determine the quality of solutions attainable" (Dudek, 2004) or "the process of ensuring an algorithmic solution is a good one: that it is fit for purpose" (Curzon et al., 2014)
- computational evaluation is originally from information retrieval research: combining effectiveness (i.e., did the process yield the correct result, e.g., accuracy, precision, recall, etc.), system quality (e.g., speed, coverage of all possible results, etc.) and *importantly* user utility (e.g., happiness, productivity, etc.) (Manning et al., 2008)

> TODO: better define 'computational evaluation' in general, and how do we define the term in the context of imputation methodology? Something like the *systematic assessment of the algorithmic behavior and data-level outputs of imputation procedures*? Computational evaluation is something to check *conditional on* acceptable statistical properties, and does not replace statistical evaluation. It focuses on: algorithmic stability (convergence, failure modes), sensitivity to modeling choices, plausibility and coherence of imputed values, and usability for applied researchers. Emphasize that often these aspects are already taken into account by imputers, but I try to systematize and operationalize these aspects that are often informal or ad hoc. See also [https://en.wikipedia.org/wiki/Evaluation_measures_(information_retrieval)]

> - not specifically defined,
> - not defined for statistics at all
> - e.g. purpose in stats verandert elke keer: voor verschil tussen groepen, voor viz? niveau van data, sample, population
> - therefore, planting a flag in this dissertation
> - stat val vs comp evel: one before the other? no, see next TODO
> - back-tracing the procedure: you should *always* be able to retrieve the chain of the imp procedure
> - imp stays a custom job: conv of imp is conditional on imp model
> - today, imp are not generated for 1 analysis prob, but all analyses that may be … on the data –> all imp models should be compatible and congenieal –> more 'omvattende' imp models required
> - liefst wil je parameters tracken, daarom is dit zo belangrijk

- computational evaluation thus goes beyond statistical evaluation: even if we know the process yields correct inferences, we can evaluate the usability/appropriateness/plausibility of the solution (e.g. do the imputed values lie within the range of the observed values; no negative ages imputed?)
- imputation pragmatists/purists don't care for the imputed values
- Rubin (1976, 1987) developed the methodology to get correct population-level inferences from incomplete samples
- implausible or impossible imputed values can yield valid statistical inference
- but are we satisfied with the way that the imp were obtained?
- even without conv, we can get valid inference at sample/population level, but not at individual cell level
- what do we care about? bias/coverage/variance, empirical relevance/plausibility, software-engineering nitpicks like speed/convergence/fault rate.
- not just taking the answer as-is, but tracing back the "black box" to some extend, digging into the algorithms
- I will focus on inferential validity, algorithmic stability, data plausibility, user burden.
- my definition of comp eval is: … –> a suite of tools to investigate at least these domains

> TODO: what are the limitations of purely inferential metrics (bias, coverage)? and vice versa: when might plausibility mislead users into believing the imputations are statistically valid? e.g. PMM with parse data might lead to imputations that are too close?

> - think about case where stat eval is ok, but not the com eval
> - the term 'black box' is not correct: completely reproducible when all param at 0
> - tracing back the solution of an alg to its origin, if that's difficult some may refer to this as a black box: this also holds for mice
> - maybe future: com eval of variable importance within imp models

Imputation methodology

- imputation methods are developed to solve missing data problems
- what types of missing data problems? incomplete samples
- typically, some but not all entries are missing per row/column (i.e., item non-response, not unit non-response or entirely missing variables)
- missing observations in incomplete cases/variables can be filled in using an imputation model
- in the FCS framework [vanbuuren2006], we need an imputation model for each incomplete variable

> TODO: explain framework here? while "Rubin (Ch. 5) distinguished three tasks for creating imputations under an explicit model: the *modeling* task the *imputation* task and the *estimation* task." (van Buuren, 2007, p. 221), FSC does not require these explicit tasks. instead, FCS only requires an *imputation model* to describe how synthetic values may be generated, without explicit specification of the joint model.

- imputation models consist of the functional form of the model (e.g., stochastic regression) and imputation model predictors (i.e., other variables in the data)
- recurring question: how do we choose an appropriate imputation model?
- [in the building stage] how do we know which functional form to choose?
  - rely on defaults, tested in simulation studies (→ evaluation chapter)
  - assess the distribution of the incomplete variable (e.g., visual inspection → `ggmice` chapter)
  - …? (maybe ad transformations?)
- [in the building stage] how do we know which imputation model predictors to select?
  - association of incomplete variable with other variables in the data (→ `ggmice` chapter)
  - association of missingness indicator with other variables in the data (→ `ggmice` chapter)
  - external input (e.g., branching patterns, or expert knowledge to correct for MNAR)
  - …?
- [after imputation] how do we assess imputation model misfit?

  - non-convergence in the algorithm (→ convergence chapter)
  - mismatch between observed and imputed data distributions (e.g., visual inspection → `ggmice` chapter)
  - …?

> TODO: link imputation methodology back to computational evaluation. "Imputation models bypass the need to specify P(Y, X, R), though their use creates new responsibilities for substantiating its correctness for a given statistical analysis." (van Buuren, 2007, p. 222).

RQ: how can applied researchers be aided in developing and evaluating imputation methods?

> TODO: sharper RQ. what is the generalizable scientific contribution of this dissertation?

Sub-questions per chapter:

- 'What data-level diagnostics are informative for assessing imputation model adequacy in practice (i.e., when inferential validity cannot be evaluated)?'
- 'How does non-convergence in iterative imputation affect inferential validity, and how can non-convergence be diagnosed?'
- 'How can computational evaluation criteria be systematically incorporated into simulation studies for imputation methodology?'

> TODO: clearly define scope: FCS, item non-response, statistical inference (not prediction or causal inference), M(C)AR

Introducing chapter 1: `ggmice`

- **RQ: how can incomplete and imputed data be visualized?**
- software published on Zenodo Oberman (2022), with open code and bug reporting via GitHub
- peer-review not formal via journal, but via Issues and Pull Requests on GitHub (10x external Issue, 13x external PR)
- impact as citations (5x according to Google Scholar), CRAN downloads (852x/month; 21k total) and conda downloads (100+/month; https://anaconda.org/channels/conda-forge/packages/r-ggmice/overview)

Wat is het probleem

- Visualization of incomplete data, imputation models, and imputed data.

- Data exploration for building imputation models: inspect marginal and joint distributions of incomplete variables
- Evaluation of 'applicablity' imputation models.
- Visualization of uncertainty due to missingness after imputation.

Welke methoden bestaan al

- `naniar` for incomplete data
- `VIM` for incomplete and imputed data (but not `mice`)
- `mice` for imputed (but not `ggplot2`)
- 'artisinal' code solutions
- print of excel export for imputation models

Wat zijn de voor- en nadelen van de huidige methoden

- missing visualization tool compatible with both incomplete data and `mice` imputations: `mice` lacks incomplete data, other packages lack support for `mice`-imputed data
- missing visualization tool for imputation models (i.e., pred and meth)

Welke nadelen/problemen beoogt de nieuwe aanpak op te lossen

- no methodology for visualizing `mice` models
- plotting tools for `mids` objects not easily editable/publication-ready
- no direct comparison tool for `mice` between incomplete and imputed data

In welke opzicht vult de nieuwe methode een lacune

- unified solution for: missingness, imputation models, imputations
- direct comparison between data before and after imputation with `mice`
- complex imputation models are easier to review through visualization (as opposed to console prints/excel exports of predictor matrix)

Op welke principes is de nieuwe aanpak gebaseerd

- Grammar of Graphics: layered plots (Wilkinson, 2012)
- Open Source Software development/FAIR
- evaluation of the distribution of observed and imputed data -> imp model should fit the observed part, imp model should be congenial

Hoe zijn deze principes in de software vertaald

- `ggmice` produces `ggplot` objects: layered, easily adjustable
- GitHub, CRAN, Zenodo, …
- providing tools to see obs and imp side-by-side

Welke onderliggende aannames zijn hierbij gebruikt

- make explicit what these plots/analyses can and cannot tell us: imp models should fit the observed part of the data, but no way to determinately state miss mech.

Hoe kun je de huidige en nieuwe gereedschappen het beste met elkaar vergelijken

- data types (e.g. `mids`)
- how 'publication-ready' the visualizations are (e.g. number of lines of code needed*)
- number of functions*
- number of downloads* `ggmice` has 800 monthly compared to `VIM` 13k and `naniar` 22k

Functionaliteit	ggmice	VIM	naniar	amelia	mice (lattice)
Visualisatie van de missingness	V	V	V	(beperkt)	V
Visalisatie van het imputatie EN nonresponse model	V	X	X	X	X
Imputatie diagnostiek (visueel) + evt in cijfer?	V	V (gedeeltelijk)	X	V	V
Ondersteuning mids en andere data typen	V	X	X	X	V (alleen lattice)
Grammar of Graphics	V	X	V	X	X
Aanpasbaarheid	hoog	laag	hoog	laag	laag
Tidyverse compatible?	volledig	beperkt	volledig	beperkt	geen
Meteen klaar voor publicatie?	hoog	middel	hoog	laag	laag

Levert het nieuwe gereedschap in de praktijk betere resultaten

- visualization =/= objective truth

Wat vinden toekomstige gebruikers van het gereedschap

- StackOverflow, GitHub issues, feedback at conferences

Wat zijn relevante indicatoren voor adoptie

- CRAN downloads
- GitHub stars and issues
- StackOverflow mentions/questions
- citations of the package (Zenodo reference)

Welke richtlijnen en advies gelden bij gebruik

- not a replacement of evaluation but support

Wat zijn beperkingen van de methode

- interpretation remains subjective (e.g. convergence, MAR assumption, etc.)

Wat zijn vragen voor nader onderzoek

- visualizing `mira` objects
- add posterior predictive check plots
- quantifying uncertainty at pattern/cell level
- tools for sensitivity analyses

Introducing chapter 2: convergence

- **RQ: FSC is iterative, so convergence *should* be a requirement for valid inference, or is it?**
- simulation study, published as pre-print (11x cited according to Google Scholar), presented at ICML

Introducing chapter 3: evaluation

- **RQ: what to look out for when designing simulation studies for the evaluation of imputation methodology?**
- published as Oberman & Vink (2024), cited 27x according to Google Scholar (of which 11x CrossRef)
- based on literature review Liu et al. (2023)

Chapter 1

[TODO: link/insert]

Chapter 2

[TODO: link/insert]

Chapter 3

[TODO: link/insert]

Discussion

Conclusion

- Computational evaluation of imputation methodology is never finished.
    - In practice, you cannot validate whether the imputation model was appropriate (???), because what you would need to do so is exactly the thing you don't have: the missing part of the incomplete data, while you only have the observed part.
    - The best thing we can do is to rely on a combination of assumptions/theoretical justifications/substantive expertise and indirect evidence of imputation model fit: check the observed versus imputed data distributions, and inspect the imputations for signs of non-convergence. This dissertation offered insight and tools to do so.
    - The only setting where we do have the comparative truth (the true but missing values) is simulation studies. Imputers have to rely on simulation study results in order to choose the best imputation method for their incomplete variables. That's why simulation studies for imputation methodology should be comparable with one another, so imputers can make a 'grounded' assessment which imputation methods may be applicable, and choose the most appropriate one.

–> example: NRI search for lower bound, so impute as 'positive behavior', but per definition wrong statistical inf, because we intentionally bias in one direction

Implications

- The main findings in this thesis are…
    - Non-convergence is hard to diagnose, and typical thresholds to evaluate non-convergence may not be applicable to FCS algorithms: MICE reaches stable output before non-convergence metrics would indicate so. There is a mismatch between theoretical convergence criteria and practical algorithmic behavior.
    - Visual inspection aids imputers in developing and evaluating imputation models, because formal tests are not available. The R package `ggmice` offers tools for the computational evaluation of the imputations models that were previously unavailable.
    - Defining and testing new imputation methods through simulation studies requires careful consideration of the simulation design and evaluation to make imputation methods comparable across studies.
- Recommendations for fellow missing data methodologists:
    - design simulation studies structured and reproducible (e.g., report design choices, publish code)
    - talk to your intended audience (i.e., applied researchers), or at least check what they are struggling with (e.g., StackOverflow)

  - …?

- Recommendations for fellow imputers:
  - think before you code
  - look at the data
  - …?

Limitations

- Computational evaluation is not a substitute for thinking. It may give a false sense of certainty, e.g. against MNAR mechanism (untestable assumption).
  - Use expert insight: use comp eval as complement to (not a replacement for) substantive knowledge and sensitivity analyses
  - Take the analysis model (if available) as starting point for imputation workflow, for congeneality
  - …?
- This entire dissertation relies on the R language and centers the `mice` package (TODO rephrase as not a flaw: I focus on a dominant ecosystem, and concepts generalize beyond).
  - While this set-up is very popular, the data science landscape has expanded and is ever evolving
  - Many machine learning and deep learning methods are not (yet) implemented as imputation models in `mice`. These might be able to better approximate the posterior predictive distribution of the missing values than `mice` methods, but research should show that still.
- ~~I haven't tested whether the tools I developed actually improve the imputations of `mice` users~~ This dissertation evaluates tools and diagnostics at the methodological level; assessing their causal impact on user behavior and imputation quality remains an important direction for future empirical research.
  - GitHub issues and StackOverflow are indication
  - Onmogelijk om evidence-based conclusies te trekken, maar misschien wel evidence-informed?
- The visualizations implemented in `ggmice` are generally better suited for numeric data, as opposed to categorical variables (or even open text field data, which is not supported by `mice` currently). Future research on: associations of cat vars with miss indicators, and conv parameter other than SD of imputed values.

> [TODO: rephrase 'haven't's into future research (positive phrasing: reframe as scope conditions, not shortcomings)]

Outlook

- This chapter ends with a future perspective…
- Missing data remains a problem indefinitely. I expect imputation methodology to be developed and applied for the foreseeable future. But the way we deal with data might change.
- With the rise of machine learning and deep learning methods (or "AI"), evaluation will become ever more important. Novel methods are being developed under a prediction framework, not statistical inference framework: ignoring the uncertainty in the estimates, and thus not incorporating between-imputation variance. This leads to too narrow CIs, and too low p-values. Which, in turn, causes spurious results.
- AI models should propagate uncertainty!
- trust in science requires honest reflection of uncertainty in our analyses
- be open about uncertainty, and limitations of scientific studies
- publish null results (e.g., registered reports or pre-registrations)
- share data and code with other scientists, and the public
- publish open access, educational materials too!
- change the way we evaluate science: this dissertation is an example of the new recognition and rewards vision. it includes software as an actual chapter, not something extra.
- I hope to inspire others to reflect on their own view of what scientific output is. And most of all, I hope to set an example for other PhD candidates to do the same

References

Curzon, P., Dorling, M., Ng, T., Selby, C., & Woollard, J. (2014). *Developing computational thinking in the classroom: A framework*. Computing at School.

Dudek, G. (2004). Computational Evaluation. In G. Dudek (Ed.), *Collaborative Planning in Supply Chains: A Negotiation-Based Approach* (pp. 165–213). Springer. https://doi.org/10.1007/978-3-662-05443-7_7

Liu, D., Oberman, H. I., Muñoz, J., Hoogland, J., & Debray, T. P. A. (2023). Quality Control, Data Cleaning, Imputation. In F. W. Asselbergs, S. Denaxas, D. L. Oberski, & J. H. Moore (Eds.), *Clinical Applications of Artificial Intelligence in Real-World Data* (pp. 7–36). Springer International Publishing. https://doi.org/10.1007/978-3-031-36678-9_2

Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press. https://doi.org/10.1017/CBO9780511809071

Oberman, H. I. (2022). *Ggmice: Visualizations for 'mice' with 'Ggplot2'* [Computer software]. Zenodo. https://doi.org/10.5281/zenodo.6532702

Oberman, H. I., & Vink, G. (2024). Toward a standardized evaluation of imputation methodology. *Biometrical Journal*, *66*(1), 2200107. https://doi.org/10.1002/bimj.202200107

Rubin, D. B. (1976). Inference and Missing Data. *Biometrika*, *63*(3), 581–592. https://doi.org/10.2307/2335739

Rubin, D. B. (1987). *Multiple Imputation for nonresponse in surveys*. Wiley.

van Buuren, S. (2007). Multiple imputation of discrete and continuous data by fully conditional specification. *Statistical Methods in Medical Research*, *16*(3), 219–242. https://doi.org/10.1177/0962280206074463

van Buuren, S., & Groothuis-Oudshoorn, K. (2011). Mice: Multivariate Imputation by Chained Equations in R. *Journal of Statistical Software*, *45*(1), 1–67. https://doi.org/10.18637/jss.v045.i03

Wilkinson, L. (2012). The Grammar of Graphics. In *Handbook of Computational Statistics* (pp. 375–414). Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-21551-3_13

CV

[TODO: write]