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 is easy with software defaults
- 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" or "the process of ensuring an algorithmic solution is a good one: that it is fit for purpose"
- 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.)
- 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
- implausible or impossible imputed values can yield valid statistical inference
- but are we satisfied with the way that the imp were obtained?
- not just taking the answer as-is, but tracing back the black box to some extend, digging into the algorithms
- 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.

Imputation methodology

- imputation methods are developed to solve missing data problems
- what types of missing data problems? 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, we need an imputation model for each incomplete variable [explain framework here?]
- 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)
  - …?
- [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)
  - …?

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

Introducing chapter 1: `ggmice`

Wat is het probleem

Visualization of incomplete data, imputation models, and imputed data. Evaluation of 'applicablity' imputation models. Visualization of uncertainty due to missingness. Data exploration for building imputation models.

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

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)

Op welke principe is de nieuwe aanpak gebaseerd

- Grammar of Graphics: layered plots
- Open Source Software development/FAIR

Hoe zijn deze principes in de software vertaald

- `ggmice` produces `ggplot` objects: layered, easily adjustable
- GitHub, CRAN, Zenodo, …

Welke onderliggende aannames zijn hierbij gebruikt

…

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

…

Wat vinden toekomstige gebruikers van het gereedschap

…

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 sensitiviity analyses

Introducing chapter 2: convergence

- FSC is iterative, so convergence *should* be a requirement for valid inference, or is it?

Introducing chapter 3: evaluation

- based on literature review published in [REF]

Chapter 1

[TODO: link/insert]

Chapter 2

[TODO: link/insert]

Chapter 3

[TODO: link/insert]

Discussion

[TODO: write]

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 evaluate 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 seach 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.
  - Visual inspection aids imputers in developing and evaluating imputation models. The R package `ggmice` offers tools 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
  - 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

[TODO: rephrase 'havent's into future research (positive phrasing)]

- Computational evaluation is not a substitute for thinking. It may give a false sense of certainty, e.g. against MNAR mechanism.
  - Use expert insight
  - 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.
  - 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 may be able to better approximate the posterior predictive distribution of the missing values than `mice` methods
- I haven't tested whether the tools I developed actually improve the imputations of `mice` users.
  - 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).

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

CV

[TODO: write]