Dissertation

Eisen onderzoek

Bron: promotiereglement.

1. De promovendus is er verantwoordelijk voor dat het onderzoek dat ten grondslag ligt aan het proefschrift voldoet aan de volgende vereisten:

a. de promovendus levert een oorspronkelijke bijdrage aan wetenschappelijk onderzoek die de in Nederland gebruikelijke kwaliteitstoetsing door vakgenoten kan doorstaan;
b. de promovendus heeft aangetoond zelfstandig de wetenschappelijke methoden van het vakgebied toe te kunnen passen in de ontwikkeling, interpretatie en toepassing van nieuwe kennis;
c. de promovendus heeft kennis genomen van en gewerkt met een substantiële 'body of knowledge', welke in ieder geval omvat de principes en methoden van de internationale wetenschapsbeoefening en de theorievorming, methoden en studies van het desbetreffende vakgebied;
d. de promovendus beschikt over het vermogen om een omvangrijk project voor de ontwikkeling van nieuwe kennis te ontwerpen en te implementeren;
e. de promovendus is in staat om de kennis en methoden van het desbetreffende specialisme en/of vakgebied adequaat over te dragen;
f. de promovendus is in staat om de maatschappelijk verantwoordelijkheid ten aanzien van het uitvoeren, toepassen en benutten van het eigen onderzoek te dragen.

2. Het onderzoek is verricht in overeenstemming met wettelijke en universitaire voorschriften en gedragsregels

Inhoud proefschrift

Introductie

Thema: computationele evaluatie van imputatiemethodologie.

H1: Standardized evaluation

- Inhoud hoofdstuk: review paper (manuscript).
- Referentie: 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
- Kwaliteitstoetsing (eis 1a) door peer-review.
  - Gepubliceerd, 20x geciteerd (bron: Google Scholar).

H2: Missing the point

- Inhoud hoofdstuk: simulatiestudie (manuscript).
- Referentie: Oberman, H. I., van Buuren, S., & Vink, G. (2021). Missing the Point: Non-Convergence in Iterative Imputation Algorithms. https://arxiv.org/abs/2110.11951
- Kwaliteitstoetsing (eis 1a) door openbaar maken vroege versie (zie ook Publish Review Curate model).
  - Pre-print gepubliceerd (4p summary presented at ICML). 11x geciteerd (bron: Google Scholar).

H3: R package `ggmice`

- Inhoud hoofdstuk: onderzoekssoftware (verwijzing naar software repository/software vignette).
- Referentie: Oberman, H. I. (2022). `ggmice`: Visualizations for 'mice' with 'ggplot2'. Zenodo. https://doi.org/10.5281/zenodo.6532702
- Kwaliteitstoetsing (eis 1a) door open bug reporting (GitHub Issues).
  - Code openbaar via GitHub, versie gepubliceerd via Zenodo. 5x geciteerd (bron: Google Scholar).
  - GitHub Issues as peer review (10x external Issue, 13x external PR).
  - CRAN downloads as impact (852x/month; 21k total).

Discussie

Plaatsing proefschrift in bredere ontwikkelingen wetenschap.

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?

- imputation methods are developed to solve missing data problems: typically, some but not all entries are missing per row/column (i.e., item non-response, not unit non-response or missing variables)
- in the FSC framework, we need an imputation model for each incomplete variable
- 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)
- 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?

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

@Gerko: graag advies!

TODO: table with visualization types in all packages (compare `VIM`, `naniar`, `amelia`)

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
- quantifying uncertainty at pattern/cell level

Chapter 2: convergence

Chapter 3: evaluation