DMP title

Project Name DMP C16/21/002 - DMP title

Project Identifier C16/21/002

Grant Title C16/21/002

Principal Investigator / Researcher Nick Vannieuwenhoven

Description Our objective is to develop the ManiFactor framework for surrogate models of maps into manifolds based on a generalization of the factor analysis model that includes generic mathematical techniques, practical learning algorithms, and research software. Numerical simulation data will be generated to demonstrate the efficacy of the developed algorithms.

Institution KU Leuven

1. General Information

Name of the project lead (PI)

Nick Vannieuwenhoven

Internal Funds Project number & title

C16/21/002 - ManiFactor: Factor analysis for smooth maps between manifolds

2. Data description

2.1. Will you generate/collect new data and/or make use of existing data?

- Generate new data

2.2. What data will you collect, generate or reuse? Describe the origin, type and format of the data (per dataset) and its (estimated) volume. This may be easiest in a numbered list or table and per objective of the project.

Work package	Type of data	Format	Volume	How created
All	Julia source code	.jl	< 50 MB	Implementation of algorithms developed during this project.
WP2.1, WP2.2, WP2.3	Numerical simulation performance data (error, timings, memory consumption, etc) of the developed algorithms.	.csv	< 100 MB	Running Julia code.
WP3.1	Numerical data measuring the condition number, including performance data (timings, memory consumption, etc) of the algorithms for computing the condition number.	.csv	< 100 MB	Running Julia code.
WP3.2	Convergence and error data of approximability of example target functions.	.csv	< 100 MB	Running Julia code.
WP2.1	Magnetic resonance imaging data.	primarily .nii and compressed .nii.gz	< 5GB / subject	Downloaded from https://doi.org/10.6084/m9.figshare.c.5315474.v1

3. Ethical and legal issues

3.1. Will you use personal data? If so, shortly describe the kind of personal data you will use. Add the reference to the file in KU Leuven's Record of Processing Activities. Be aware that registering the fact that you process personal data is a legal obligation.

The anonymized data set is described by Tian, Fan, Witzel, et al. (2022) in Scientific Data 9, no. 7 and consists of diffusion MRI scans of human brains. Ethical approval and written consent of cognitively normal individuals was obtained by these authors to collect and distribute this data set. This data set is distributed under CC0 - No Copyright.

While the MRI scans reveal anatomic details of individual brains, no personally identifying information (names, social security numbers, address, etc) is included with the data set that would enable precise or even plausible identification of any individual. Since the anonymized data is freely available on a public figshare repository and no distribution restrictions apply, with the subjects's consent, none of the essential data management aspects (privacy, access control, secure storage, sharing and redistribution, retention period) apply.

The data set will be used exclusively to generate realistic diffusion tensor images, which consist of a grid of small-scale positive semi-definite matrices. This grid of matrices will be approximated (for the purpose of error correction, data compression, and missing data completion) using the ManiFactor algorithms.

All other numerical data we generate with our Julia codes is not personal data.

3.2. Are there any ethical issues concerning the creation and/or use of the data (e.g. experiments on humans or animals, dual use)? If so, add the reference to the formal approval by the relevant ethical review committee(s).

The purpose of our analysis of the reused data set is to apply diffusion tensor imaging techniques to obtain small-scale positive semi-definite matrices arranged on a regular grid. These will then be investigated from a purely mathematical point of view as regards to the potential of the developed methods to clean (noise reduction) such data sets and to evaluate insofar as such data could be effectively and accurately (but lossy) compressed using the developed mathematical methods. We will not redistribute this data set or modifications thereof.

3.3. Does your research possibly result in research data with potential for tech transfer and valorisation? Will IP restrictions be claimed for the data you created? If so, for what data and which restrictions will be asserted?

If the scientific hypotheses are all largely correct, we may anticipate that ManiFactor algorithms could be used to compress, among others, tensor diffusion imaging data. Such a technology could be valorized and may even be commercially exploitable. We do not believe it plausible that IP restrictions will be claimed for the derived numerical predictions we will construct from a subset of the MRI data set.

The predictions made by ManiFactor algorithms and the compression results will not be distributed as a new data set. These comprise our raw data.

The employed data set (https://doi.org/10.6084/m9.figshare.c.5315474.v1) is licensed under CC0 - No Copyright, so no IP restrictions apply.

3.4. Do existing 3rd party agreements restrict dissemination or exploitation of the data you (re)use? If so, to what data do they relate and what restrictions regarding reuse and sharing are in place?

The reused data set (https://doi.org/10.6084/m9.figshare.c.5315474.v1) is licensed under CC0, so no copyright restrictions apply. We will not distribute any derived data sets.

The data we will generate ourselves is not subject to any restrictions.

4. Documentation and metadata

4.1. What documentation will be provided to enable understanding and reuse of the data collected/generated in this project?

All of the numerical data generated in this project, both raw and processed, will be the result of running computer codes. These codes describe in a mathematically rigorous, completely precise and unambiguous way how the data is generated. The codes for generating and processing the data will be documented in a human-readable format as well (as part of the Julia code files), explaining what they intend to do.

The comma separated values-files from all work packages are intimately linked with the code that generated them. This linking is described in a bash script file that calls the code and generates the output file. The comments in these bash scripts will describe what numerical experiment the bash script will execute. For each experiment, a ReadMe file documents the computer architecture on which the experiments were carried out.

The foregoing means that for each experimental data file X.csv, there is a corresponding X.sh that when executed generates the X.csv data file, and finally an X.txt plain text file that describes the intention of the experiment, the computer architecture and computer parameters that were used, and if applicable, the precise table, figure, or section that the X.csv provides the data for in the preprint based on this data. The

.csv-files will always start with a header that identifies the information contained in each column in a human-readable format. For example "relative backward error", "time (s)", "peak memory consumption (MB)", etc. The mathematically precise definition of the data items is rigorously encoded in the program code.

The default file structure, limited to the data aspects, we use is:
>AuthorInitials
-> code
-> experiments
--> date of experiment
---> relevant code files
---> X.csv
---> X.sh
---> X.txt
-> literature
-> paper
-> plots
--> date of processing
---> relevant processing code files
---> generated figure output files (.eps, .pdf)
---> generated table output files (.txt, .tex)

The raw data in the .csv-files will be processed exclusively using additional computer codes in, among others, Julia (.jl), Python (.py), Perl (.pl), Matlab/Octave (.m), or Gnuplot (.plot). They will generate, as necessary, the relevant plots in vector graphic formats like .eps and .pdf, or the relevant tables in LaTeX code, either as a standalone, importable LaTeX file (.tex) or simply as the relevant table environment that can be directly copied into the paper (.txt).

4.2. Will a metadata standard be used? If so, describe in detail which standard will be used. If not, state in detail which metadata will be created to make the data easy/easier to find and reuse.

No standard metadata formats exist in our domain due to the highly variable types of data that is generated as a result of numerical simulations. In addition, as described above, the data-generating process is described in a fully formal way through the program code that generates the data. As such, most aspects that one would otherwise describe with metadata are formally encoded in the (human-readable and machine-executable) program code. In some sense, this code is the metadata.

5. Data storage and backup during the project

5.1. Where will the data be stored?

All of the program codes, including the ones to generate and process the data, will be developed using a version management system, in particular KU Leuven GitLab. One (private) GitLab repository will be created wherein the codes will be developed by the authors. This private repository is also used to host and develop the LaTeX files of the scientific article based on the developed theory, algorithms, and data. This repository will be completely private with read and write access restricted to the authors involved in performing the research.

When the final article is completed, a new fully public GitLab repository is created, hosting the final program codes and developed algorithms, including the codes for generating the data files and producing the processed data (i.e. figures and tables). This repository will have public read rights, but no write rights. Our articles will then reference this public repository. The articles themselves are posted as preprints to the arXiv.org preprint servers.

Upon acceptance of the article, the final article's pdf-file, figures, code, and processed data is archived both to the PI's KU Leuven OneDrive (identified by the author's last names initials, then a dash, and then the arXiv identifier), as well as to the Department of Computer Science's NextCloud storage, under the modalities of the NUMA division's data management plan.

During theoretical and practical development, the article files and codes will also be present on the author's personal work computer. After completing small chunks of work (e.g., once per day), these files are to be synchronized with the version management system (i.e., KU Leuven GitLab). During development, generated raw data files will be stored exclusively on the author's personal work computer (or potentially on external compute servers such as the VSC's). These data files are not crucial and can always be generated from the current version of the code in the private GitLab repository.

5.2. How will the data be backed up?

The program code will be stored in a KU Leuven GitLab repository, which provides both version control (i.e. tracking what changes were made when and by who) and automatic backups. Note that researchers developing the program code will keep their updates synchronized with the GitLab repository by committing and pushing the changes at least on a daily basis.

The data generated and used by any of our articles will be stored in KU Leuven OneDrive, which provides automatic backups to KU Leuven servers. In addition, the paper, codes, figures, and processed data will be archived on the Department of Computer Science's NextCloud service upon completion and final acceptance of each article. These servers are also automatically backed up (in the Department of Computer Science).

The data that is stored only on the author's work computers is not critical and can be generated from the code files or downloaded from public repositories (in the case of the MRI data files). Nevertheless, the Department of Computer Science automatically provides monthly backups of the home directories of all work computers. No specific backup policy is enforced on these files due to their transient and noncritical status.

5.3. Is there currently sufficient storage & backup capacity during the project? If yes, specify concisely. If no or insufficient storage or backup capacities are available, then explain how this will be taken care of.

The storage offered within KU Leuven OneDrive will suffice for our long-term data storage needs, which are expected to be well below a hundred megabytes (in uncompressed format).

5.4. What are the expected costs for data storage and backup during the project? How will these costs be covered?

We will use free solutions offered by KU Leuven.

5.5. Data security: how will you ensure that the data are securely stored and not accessed or modified by unauthorized persons?

No sensitive data will be processed. Recall that the MRI data is anonymous and contains no personally identifying information. Since this data is available in a public persistent figshare repository, we will not keep publically accessible copies of this data. The only copies will be local working copies on the researcher's computer.

Aside from this MRI data from a public repository, our data mainly consists of numerical experimental data that we will generate and which is absolutely not sensitive, as it consists of algorithmic performance data. Consequently, no specific security measures will be taken, other than password protection of the work computers and usage of a private GitLab repository during the research phase of the project. This GitLab repository is secured with the KU Leuven Authenticator (which uses two-factor authentication).

6. Data preservation after the end of the project

6.1. Which data will be retained for the expected 10 year period after the end of the project? If only a selection of the data can/will be preserved, clearly state why this is the case (legal or contractual restrictions, physical preservation issues, ...).

The data that is reused will not be retained, as it is freely publically available in a persistent figshare repository and fully documented in a scientific article published in Scientific Data 9, no. 7. All generated convergence and performance data will be retained in the private GitLab repository for this project.

All final data that constitutes the basis of our scientific articles will be replicated in the public GitLab repository for that paper. In addition, a version will be archived on the Department of Computer Science's NextCloud solution (as per the data management plan of the NUMA division). Finally, another copy will be archived in the PI's KU Leuven OneDrive repository, employing the naming conventions outlined above.

Note that all of our data can be generated anew using our program codes. These codes are also stored in the private GitLab (for the working copies and research version), the public GitLab (for the final version), and private NextCloud and OneDrive (archived, final versions).

6.2. Where will these data be archived (= stored for the long term)?

The final data, including the program codes that can generate these data anew, that are the basis of our scientific articles will be archived both in NextCloud (as per our division NUMA's data management plan) and the PI's OneDrive. In addition, the fully public GitLab page corresponding to the article, as explained above, will also host the final data and codes.

In the unlikely case that the volume of data is too large to store on GitLab, only the codes that can generate that data will be stored. Any researcher can then generate the data anew using these codes.

6.3. What are the expected costs for data preservation during these 10 years? How will the costs be covered?

Since the data volumes are expected to be very small (well below a few hundred megabytes), the storage limits of the Department of Computer Science's NextCloud, KU Leuven's OneDrive, and KU Leuven's GitLab will not be exceeded. Hence, the expected cost is zero.

7. Data sharing and re-use

7.1. Are there any factors restricting or preventing the sharing of (some of) the data (e.g. as defined in an agreement with a 3rd party, legal restrictions or because of IP potential)?

All final research data that support our scientific articles will be made publically available in the aforementioned public GitLab repository. The reused data set will not be shared, since it is already available in the public domain in a public figshare repository.

7.2. Which data will be made available after the end of the project?

All final research data that support our scientific articles will be made publically available in the aforementioned public GitLab repository. The articles themselves will additionally be posted to the arXiv.org preprint servers.

7.3. Where/how will the data be made available for reuse?

- In an Open Access repository

All final research data, including our computer codes that can generate these data anew, that support our scientific articles will be made publically available in the aforementioned public GitLab repository.

7.4. When will the data be made available?

- Immediately after the end of the project

All final research data including our computer source code will be made publicly available in aforementioned GitHub repository after completing the project and uploading the resulting scientific article to the preprint server arXiv.org.

7.5. Who will be able to access the data and under what conditions?

All final data, excluding the computer source codes, will be released into the public domain through the aforementioned public GitLab repository of the article. No access control is necessary for this. The source codes will be released under a CC-BY-SA 4.0 licence, in the GitLab repository. This means that anyone can modify, update, and incorporate our code, as long as any redistribution of said codes documents these changes, attributes the original source, and imposes the same CC-BY-SA 4.0 license conditions.

7.6. What are the expected costs for data sharing? How will these costs be covered?

Since the data volumes are expected to be very small (well below a few hundred megabytes), the storage limits of KU Leuven's GitLab will not be exceeded. Hence, the expected cost is zero.

8. Responsibilities

8.1. Who will be responsible for the data documentation & metadata?

The Ph.D. students assigned to the work packages are responsible for documenting the data, following the data protocols, and supplying the metadata (including computer codes).

8.2. Who will be responsible for data storage & back up during the project?

The Ph.D. students assigned will be responsible for uploading all relevant data to GitLab. The PI Nick Vannieuwenhoven will be responsible for final archiving in the Department of Computer Science NextCloud storage system and in KU Leuven's OneDrive.

8.3. Who will be responsible for ensuring data preservation and sharing?

The PI bears the end responsibility of ensuring the data is preserved in NextCloud, OneDrive, and GitLab.

8.4. Who bears the end responsibility for updating & implementing this DMP?

The end responsibility for updating and implementing the DMP is with the supervisor (promotor).