Modeling Mortality Data

A Case Study in Data Management for Computer Science Research

Anna Newman - Simmons College School of Library and Information Science - anna.newman@simmons.edu

<introduction>

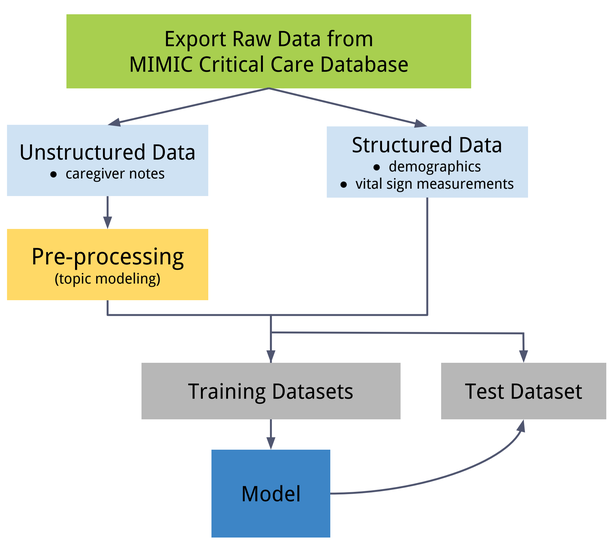
This case study, completed as part of the Simmons College Scientific Research Data Management course, identifies data management needs and best practices for computer science research. The focus of the case study is a research project that used computational methods to study factors that influence patient mortality in Intensive Care Units.

<method>

> Develop interview instrument
> Interview doctoral student from the laboratory
> Create research narrative
> Map data management practices and needs to the New England Collaborative Data Management Curriculum Modules for Managing Research Data
> Create data management plan, outlining best practices for data management in computer science research

<research question & process>

How can structured and unstructured data from electronic healthcare records be used to automatically identify factors that influence patient outcomes in Intensive Care Units?



<modules for managing research data>

Types & Formats of Data

- Exporting and pre-processing structured and unstructured data
- Multiple, randomly generated training and testing datasets
- Multiple programming languages (SQL, Python, MATLAB) and file formats (.csv, .sql, .sqf, .py, .txt., .m, .mat)
- Creation and deletion of intermediate datasets

Contextual Details

- No standard procedure for source code documentation
- File naming and organization conventions
- Version control

Data Storage & Security

- Security guidelines from MIMIC Critical Care Database
- Storage of intermediate data
- Multiple collaborators on source code

Legal & Ethical Concerns

- Using de-identified healthcare data
- Using data from a third party
- Ownership of source code

Data Sharing & Reuse

- Restrictions on data sharing for medical data from a third party
- Plan for source code sharing and reuse

Repositories, Archiving & Preservation

- Ensure preservation of proprietary programming files

<data management best practices>

> Establish and document file naming conventions
> Use built-in metadata generation capabilities of research software
> Implement version control software (e.g., Git)
> Develop plan for source code documentation and use documentation software (e.g., Natural Docs)
> Curate intermediate datasets generated during research
> Create preservation-ready versions of files
> Release source code under open source license (e.g., GNU General Public License)
> Deposit source code in repository for preservation, GitHub for sharing and reuse, with readme file for documentation

<conclusion>

As more scientific disciplines adopt computational methods in their research, information professionals will need to develop the knowledge and expertise to respond to the specific challenges to effective data management presented by computational research, including documentation, ownership, and preservation of source code. In particular, it is important to value and manage source code as data, in order to ensure reproducibility of results and open access to research data. Studying data management practices in computer science research allows for the identification of a set of key concerns and best practices that can be applied to computational research in a range of other disciplines.

<acknowledgements>

Tristan Naumann, Elaine Martin, Regina Raboin, Julie Goldman, Simmons LIS 532G, Fall 2015