Correlation between number of marriages in the EU/austria and students of public universities in austria

A Data Management Plan created using DMPonline

Creator: David Hinterndorfer

Affiliation: Other

ORCID iD: https://orcid.org/0000-0002-9909-227X

Template: European Commission (Horizon 2020)

Project abstract:

Experiment to find out if and how much the number of marriages in the EU and austria correlates with the number of students on public universities in austria.

Last modified: 10-04-2019

Correlation between number of marriages in the EU/austria and students of public universities in austria - Detailed DMP

1. Data summary

State the purpose of the data collection/generation

The aim of this experiment is to find out if and how much the number of marriages in the EU and austria correlates with the number of students on public universities in austria.

Explain the relation to the objectives of the project

The experiment has been conducted with python scripts (including the libraries pandas and plotly). The scripts transform the input data, writes and plots the results. We have included instructions (README.md) on how to run the experiment either directly or via Docker.

Specify the types and formats of data generated/collected

Input data

The project accesses three external CSV datasets and uses it as input for the pyhon scripts. All datasets have been downloaded along with the source code in the folder *input*.

The formats of the datasets are as follows.

1. Studierende an öffentlichen Universitäten nach Uni. (input/stud.csv)
   File format: CSV
2. Studierende an öffentlichen Universitäten nach Uni: C-SEMESTER-0 mapping (input/stud_mapping.csv)
   File format: CSV
3. Marriages by sex and previous marital status (input/marriage.csv)
   File format: CSV

Produced data

This project produces on the one hand a **aggregated dataset in CSV** format, that contains data points that combines the number of students on public universities in austria with the numbers of marriages in austria and the EU and on the other hand a **correlation plot** of these **in HTML** format.

Specify if existing data is being re-used (if any)

As mentioned before the project accesses three external CSV datasets which are listed above. The origins of this data is specified in the next section.

Specify the origin of the data

The datasets are coming from two different repositories.

- The data about students on public universities in austria and the mapping data was retrieved from data.gv.at and was published bei "Statistik Austria" (https://www.data.gv.at/katalog/dataset/stat_studierende-an-offentlichen-universitaten-nach-uni/resource/6a0f4051-f269-4773-b4cf-29b6b9ddf8dc - Accessed on 01 April 2019)
- The data about marriages was retrieved from eurostat. (http://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=demo_nmsta&lang=en - Accessed on 01 April 2019)

State the expected size of the data (if known)

Input data

1. Studierende an öffentlichen Universitäten nach Uni. (input/stud.csv)
   File size: ~26 kB
2. Studierende an öffentlichen Universitäten nach Uni: C-SEMESTER-0 mapping (input/stud_mapping.csv)
   File size: ~2 kB
3. Marriages by sex and previous marital status (input/marriage.csv)
   File size: ~639 kB

Produced data

1. Aggregated dataset
   File size: ~340 Bytes
2. Correlation plot
   File size: ~3 MB

Outline the data utility: to whom will it be useful

As descibed later the data of the project is published to data repsotiories and licensed for reuse. Therefore the data is useful for every person which is interested and wants to work with this data.

2.1 Making data findable, including provisions for metadata [FAIR data]

Outline the discoverability of data (metadata provision)

To make the project findable and increase discoverability of the data we applied different methods. The following gives a short overview.

- Metadata
- Indentifiers (DOI, OCID)
- Repositories (Zenodo, Github)
- Naming conventions
- Versioning
- Search keyword

Metadata

To describe the metadata of the data/project we added a metadata file (*documentation*/metadata.xml), which describes the project. The following list contains the included information.

- Title of the project
- Author Information
- Date of data collection
- Licenses/access restrictions placed on the data
- Data formats
- Project type
- Time range
- Language

Additionally a descriptive file is added, which explains the axes and units of the generated output. The description can be found in a txt-file in the documentation folder of the project (*documentation*/description.txt).

Project structure

The following section descibes the structure of the project.
Python scripts are located in the projects root folder, where also all the commands, described in the README.md, are executed from. The external datasets are located in the *input* folder. Intermediate results as well as the final plot gets written into the*output* folder. Metadata and documentation can be found in the *documentation* folder.

Other important resources, like the README.md, LICENSE or the Dockerfile, are located in the projects root directory.

Outline the identifiability of data and refer to standard identification mechanism. Do you make use of persistent and unique identifiers such as Digital Object Identifiers?

The project was published to *Github and Zenodo*. Through the upload to Zenodo the project got a **DOI** (Digital Object Identifier) as identifier to improve findability. The Zenodo respository is linked with the Github repository.

Also the creator of the project created a **ORCID** and linked the project to find the project over the researcher.

Outline naming conventions used

For filenames of the project the following conventions were applied:

- Use '_' or '-' to delimit words, not spaces.
- Keep file names short, but meaningful.
- Use meaningful folder names.
- Mark the order of scripts by a consecutive number at the beginning (e.g. 01_*, 02_*, 03_*)
- Only use lower case letter (except: README.md, LICENSE, Dockerfile)

Outline the approach towards search keyword

Different keywords describing the project were provided to the repositories. The keywords added are the following:

- data
- stewardship
- dmp
- marriage
- students
- correlation

- plotly
- pandas
- python
- austria
- eu

Outline the approach for clear versioning

The versioning of the project is done and managed with Github.
The version seperates into three parts (a.b.c)

- the first number marks a major version make incompatible API changes (a)
- the second number marks minor version changes (b)
- the third number marks patch changes like bug fixes (c).

Specify standards for metadata creation (if any). If there are no standards in your discipline describe what metadata will be created and how

As described before we added a metadata file (*documentation*/metadata.xml).
This file was written in XML format by using the DCMI (Dublin Core metadata initiative) Metadata Terms and there XML scheme as standard.

2.2 Making data openly accessible [FAIR data]

Specify which data will be made openly available? If some data is kept closed provide rationale for doing so

All the data used in the project is made openly available by publishing it to*Github and Zenodo.*

Specify how the data will be made available

Code repository

The source code as well as the data of the project was pushed to a public git repository on Github (https://github.com/Hido1994/exercise1-dmp).

Data repository

Also we uploaded the projects data to the data repository Zenodo. Through the upload to Zenodo the project got a DOI which is used to identify the data (https://doi.org/10.5281/zenodo.2634722).

Specify what methods or software tools are needed to access the data? Is documentation about the software needed to access the data included? Is it possible to include the relevant software (e.g. in open source code)?

To access the project the only thing needed is a browser. The scripts, documentation and results are also readable offer the browser or a simple text editor.

To reproduce the results either docker or python (including pandas and plotly) is needed (pip is recommended to install pandas and plotly).

Specify where the data and associated metadata, documentation and code are deposited

As descibed before the data including the metadata, documentation and code is deposited to a code repository (Github) as well as a data respository (Zenodo) and will be available even after the end of the project.

Specify how access will be provided in case there are any restrictions

The data is hosted in Github and Zenodo. It can be accessed by everyone (read permissions). The write permissions are limited to the researchers working on the project.

Permissions are managed via Github's and Zenodo's account system.

2.3 Making data interoperable [FAIR data]

Assess the interoperability of your data. Specify what data and metadata vocabularies, standards or methodologies you will follow to facilitate interoperability.

The project aims to collect and document the data in a standardised way to
ensure that, the datasets can be understood, interpreted and shared in isolation alongside
accompanying metadata and documentation. In the following sections some measures to make the project interoperable are descibed.

Metadata

As described before a metadata file was created which holds information about the project and references datasets used. The metadata was written in XML format by using the DCMI (Dublin Core metadata initiative) Metadata Terms and there XML scheme.

The following list contains the included information.

- Title of the project
- Author Information
- Date of data collection
- Licenses/access restrictions placed on the data
- Data formats
- Project type
- Time range
- Language

Formats

The data generated with the project uses formats which can be easily re-used by other researchers.

- The combined datasets are written as CSV file so it can be accessed and altered with a default text editor.
- The resulting plot is generated in HTML format. By accessing it with the browser different plots (where data is removed or added) can be extracted as image.
- The scripts are written in python which also can be accessed, altered and extended with a default text editor. To run the scripts just python (including pandas and plotly) is needed.

Specify whether you will be using standard vocabulary for all data types present in your data set, to allow inter-disciplinary interoperability? If not, will you provide mapping to more commonly used ontologies?

As descibed above the data can be easily reused and standard vocabulary was used as much as possible (e.g. DCMI Metadata Terms to descibe the metadata)

2.4 Increase data re-use (through clarifying licenses) [FAIR data]

Specify how the data will be licenced to permit the widest reuse possible

All code, data and documentation is available on Github and is licensed under the MIT license.

The external datasets are using permissible licenses which allows us the usage and redistribution of the following data:

- Students on public universities - Creative Commons Attribution License 3.0
- Marriages by sex and previous marital status - Free to use and distribute according tohttps://ec.europa.eu/eurostat/about/policies/copyright

Specify when the data will be made available for re-use. If applicable, specify why and for what period a data embargo is needed

As descibed all code, data and documentation of the project is published and licensed in a way that it can be reused by others.

Specify whether the data produced and/or used in the project is useable by third parties, in particular after the end of the project? If the re-use of some data is restricted, explain why

The data of the project is made available for re-use through uploads to a git repository on Github (https://github.com/Hido1994/exercise1-dmp) as well as Zenodo (https://doi.org/10.5281/zenodo.2634722). There it can be accessed and downloaded by everyone even after the project.

Describe data quality assurance processes

As mentioned in the previous sections there are different methods applied to assure quality and make the data FAIR.
The following list gives an overview of the applied methods.

- Clear versioning
- Naming conventions
- Metadata
- Using Standards
- Licensing
- Clear project structure
- Deposit data to repositories
- Link data
- etc.

Specify the length of time for which the data will remain re-usable

As mentioned in the previous sections all the data of the project will be available after the end of the project and simply can be used, altered and extended by others without the need of any proprietary software. Therefore all the project data have long-term validity.

3. Allocation of resources

Estimate the costs for making your data FAIR. Describe how you intend to cover these costs

The data was made FAIR by using only services which are free and accessible by everyone. The data is store in repositories on Zenodo and Github where the data is hosted and even after the conclusion of the project there are no additional costs to consider.

Clearly identify responsibilities for data management in your project

The researcher of the project is responsible for data management in this project.

Describe costs and potential value of long term preservation

As descibed there are no additional costs in preserving the data and therefore it should be no problem in preserving the data in long term.

The following files are relevant to reproduce the experiment and should be considered for long term preservation:

- README.md - Text file containing instructions on how to run the experiment
- 01_transform.py - Python script to transform the input datasets into aggregated dataset
- 02_visualize.py - Python script to generate the correlation plot
- Dockerfile - To build a docker container for running the experiment
- documentation/architecture.png - Architectual diagram of the experiment
- documentation/description.txt - Text file describing the correlation plot
- documentation/metadata.xml - Metadata relevant to the experiment

The input datasets were not added to this because they already maintained by other repositories and easy to get.

4. Data security

Address data recovery as well as secure storage and transfer of sensitive data

Store and backup

During the project the data is stored and hosted in a git repository on Github. After conclusion of the project will be also published to Zenodo.

Security

Read access is open to everyone. Write-access is limited to the researchers working on the project. Permissions are managed via Github's account system.

The project does not contain any sensitive data which has to be secured.

5. Ethical aspects

To be covered in the context of the ethics review, ethics section of DoA and ethics deliverables. Include references and related technical aspects if not covered by the former

The project does not use any sensitive data, which cause ethical issues and therefore no measures were taken.

6. Other

Refer to other national/funder/sectorial/departmental procedures for data management that you are using (if any)

There are no other procedures for data management used in this project, which are not already mentioned.