Data and text mining

dbgap2x: an R package to explore and extract data from the database of Genotypes and Phenotypes (dbGaP)**Grégoire Versmée^{*,†}, Laura Versmée[†], Mikaël Dusenno, Niloofar Jalali and Paul Avillach**

Department of Biomedical Informatics, Harvard Medical School, Boston, MA 02115, USA

^{*}To whom correspondence should be addressed.[†]The authors wish it to be known that, in their opinion, the first two authors should be regarded as Joint First Authors.

Associate Editor: Jonathan Wren

Received on April 20, 2019; revised on July 26, 2019; editorial decision on August 25, 2019; accepted on August 27, 2019

Abstract

Summary: Based on the Genomic Data Sharing Policy issued in August 2007, the National Institutes of Health (NIH) has supported several repositories such as the database of Genotypes and Phenotypes (dbGaP). dbGaP is an online repository that provides access to large-scale genetic and phenotypic datasets with more than 1000 studies. However, navigating the website and understanding the relationship between the studies are not easy tasks. Moreover, the decryption of the files is a complex procedure. In this study we propose the dbgap2x R package that covers a broad range of functions for searching dbGaP studies, exploring the characteristics of a study and easily decrypting the files from dbGaP.

Availability and implementation: dbgap2x is an R package with the code available at <https://github.com/gversmee/dbgap2x>. A containerized version including the package, a Jupyter server and with a Notebook example is available at <https://hub.docker.com/r/gversmee/dbgap2x>.

Contact: gversmee@mail.harvard.edu

Supplementary information: [Supplementary data](#) are available at *Bioinformatics* online.

1 Introduction

Since the Genomic Data Sharing Policy was issued in 2007 (Contreras, 2015), the National Institutes of Health (NIH) has supported several repositories such as the database of Genotypes and Phenotypes (dbGaP), Genome Expression Omnibus (GEO) and the Cancer Genomics Hubs. dbGaP is an online repository created by the National Center for Biotechnology Information (NCBI) (Mailman *et al.*, 2007; Tryka *et al.*, 2014; Wong *et al.*, 2017). It was developed to distribute the results of studies that have investigated the interaction between phenotype and genome. The dbGaP platform provides access to large-scale genetic and phenotypic datasets. As of October 2018, dbGaP had 1169 studies; 384 987 variables; 7006 phenotype datasets and 1010 molecular datasets registered (<https://www.ncbi.nlm.nih.gov/projects/gapsolr/facets.html>). The platform provides two levels of access: a public one and a controlled one. The public access allows users to navigate and search the study metadata, variable names, detailed description of variables and the documentation to identify datasets that are suitable for researchers to allow users to develop hypotheses. The access to patient-level data is granted only to users authorized by the appropriate National Institute of Health (NIH) Data Access Committee after submitting a data request with a Data Use Certification.

However, browsing the website and understanding the relationship between the studies is not a straightforward task, and requires the user to know the accession levels of the studies (parent studies and child studies).

The files' decryption process is also complex and requires eight steps as follows: (i) download Aspera Connect software; (ii) select the files; (iii) download the NCBI SRA Toolkit (Sequence Read Archive Toolkit) (<https://www.ncbi.nlm.nih.gov/sra/docs/toolkitsoft>); (iv) configure it by going to the subdirectory and running some commands; (v) find your decryption key; (vi) open a terminal window; (vii) go to the directory where the key is located; and (viii) download and decrypt the files by running commands provided by the SRA Toolkit. In order to be more reproducible and to avoid potential errors, we automated this process into one function. While providing access to clinical and genomic data for researchers, dbgap is hard to explore and not suited for semi or automated processing. The 'Entrez' API from NCBI (<https://www.ncbi.nlm.nih.gov/books/NBK25501>) helps to search in the database, but not to download data. To the best of our knowledge, there is no other tool (packages or REST API) that both enable easy navigation of dbGaP and automatize the dbGaP data retrieval.

We propose an R package, dbgap2x (dbGaP to eXplore and to eXtract) for searching dbGaP studies, exploring the characteristics of a study and providing easy download and decryption of phenotypic and genomic data files.

2 Materials and methods

The dbgap2x package consists of a set of 17 functions: 15 functions to explore public data and two functions to download and decrypt

dbGaP datasets. The `dbgap2x` package is written in R language. It requires R and Docker for the download and decryption processes. The exploring functions use the dbGaP platform either by extracting and parsing the data via HyperText Transfer Protocol (HTTP) or directly by File Transfer Protocol (FTP). The downloading and decrypting functions use a containerized version of the SRA Toolkit (version 2.9) available at <https://hub.docker.com/r/gversme/sratoolkit>. SRA Toolkit is a software that helps the user to decrypt the dataset from dbGaP by converting them from the SRA format into a different format such as fasta, fastq and sff. It is also used in our case to download files from dbGaP, as well as decrypting the phenotypic data files. The characteristics of the R installation used for this demo is available in [Supplementary Material](#).

3 Results

The software can be installed via the `install_github()` function of the `devtools` package and is available at github.com/gversme/dbGaP2x. To test the software, we created a containerized version that includes a Jupyter server with a Jupyter notebook ([Kluyver et al., 2016](#)), an R installation with the proper packages and the `dbgap2x` package with a tutorial. This version as well as the Docker file are available at <https://hub.docker.com/r/gversme/dbgap2x> and can be launched and tested on a local environment.

3.1 Exploring dbGaP

3.1.1 Search on dbGaP

By applying the following functions, the user will be able to extract the information about the name of the study, the association between studies, the latest version of a study and the associated websites for a study.

- `search.dbgap()`: allows finding a study related to the term searched in the web browser.
- `parent.study()`, `is.parent()`, and `sub.study()`: describe the relationship between the studies
- `browse.study()`: returns the website associated with a study.
- `phs.version()`: returns the latest version of a study.

3.1.2 Explore the phenotypic characteristics of a study

Each dbGaP study can have multiple consent groups that have their specificities. The total number of patients will be changed according to each specific consent group. The following functions are providing descriptive information about the different consent groups as well as the number of variables in each study.

- `consent.groups()`: provides the number and the name of all the consent groups.
- `n.pop()` returns a descriptive table of the population per consent group.

- `n.variables()` give you the number of variables of a study.
- `datatables.dict()` returns a table with a description of a study.
- `variables.dict()` provides a table with the description of all the variables.

3.2 Download and decrypt data from dbGaP

The investigator can download the encrypted files (clinical and genomic data) using a containerized version of the SRA Toolkit. We used Docker in order to preserve the maintainability as well as the pre-configuration of the SRA Toolkit software for the user. The users will need to provide their dbGaP repository key file location to download and decrypt the files. They will also need to provide the cart file, to indicate the files they want to download. The description of how to get this cart file is available in [Supplementary Material](#). The `dbgap.download()` function will download and decrypt the files at the same time, and push them in the directory indicated by the user. The `dbgap.decrypt()` will decrypt files that have been previously downloaded by the user.

4 Conclusion

`dbgap2x` is a new tool for exploring, downloading and decrypting data from dbGaP. This user-friendly package is suited for all researchers using dbGaP, for facilitating and accelerating the navigation of dbGaP data and understanding the relationship between studies.

Funding

This work was supported by the National Institutes of Health [Data Commons PIC-FAIR OT3 OD025466-01]; and the National Heart, Lung, and Blood Institute [DataStage OT3 HL142480-01].

Conflict of Interest: none declared.

References

- Contreras,J.L. (2015) NIH's genomic data sharing policy: timing and trade-offs. *Trends Genet.*, **31**, 55–57.
- Kluyver,T. et al. (2016) Jupyter Notebooks—a publishing format for reproducible computational workflows. In: Loizides,F. and Schmidt,B. (eds) *Positioning and Power in Academic Publishing: Players, Agents and Agendas*. The authors and IOS Press, Amsterdam, The Netherlands, pp. 87–90.
- Mailman,M.D. et al. (2007) The NCBI dbGaP database of genotypes and phenotypes. *Nat. Genet.*, **39**, 1181–1186.
- Tryka,K.A. et al. (2014) NCBI's database of genotypes and phenotypes: dbGaP. *Nucleic Acids Res.*, **42**, D975–9.
- Wong,K.M. et al. (2017) The dbGaP data browser: a new tool for browsing dbGaP controlled-access genomic data. *Nucleic Acids Res.*, **45**, D819–D826.