The Journal of Open Source Software

densify: An R package to reduce empty cells in data frames of typological linguistic data

Anna Graff [1,2,3¶], **Marc Lischka** [4], **Taras Zakharko** [1,3], **Reinhard Furrer** [3,4], **and Balthasar Bickel** [1,3]

1 University of Zurich, Department of Comparative Language Science **2** University of Zurich, Department of Evolutionary Biology and Environmental Studies **3** University of Zurich, Center for the Interdisciplinary Study of Language Evolution **4** University of Zurich, Department of Mathematical Modeling and Machine Learning ¶ Corresponding author

DOI: 10.21105/joss.07024

Software

- Review
- Repository
- Archive

Editor: Øystein Sørensen

Reviewers:

- @elenlefoll
- @yjunechoe
- @stefanocoretta

Submitted: 05 June 2024
Published: 28 August 2024

License
Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License (CC BY 4.0).

Summary

The R package `densify` provides a procedure to prune input data frames containing empty cells (or cells with values {?} or {NA}) to denser sub-matrices with fewer empty cells. The pruning process trades off a series of variably weighted concerns, including data retention, coding density (proportion of non-empty cells) and taxonomic diversity of rows (representing for example phylogenetic relations). Users can adapt the relative weights given to these concerns through various parameters so that the densification process best fits their needs. As such, the software is useful for several purposes, including the densification of sparse input matrices and the subsampling of large input matrices according to a procedure that is sensitive to taxonomic structure.

Statement of Need

Linguistic typological data is increasingly available in large-scale databases, and many analyses that aim at exploring diversity or testing hypotheses rely on such databases. Some of these resources have information for nearly all features (sometimes also called characters, parameters or variables) across nearly all languages in the database (i.e. they have complete or near-complete coding density), but the dataframe may be too large for certain computationally intensive analyses (e.g., PHOIBLE (Moran & McCloy, 2019), Grambank (Skirgård et al., 2023)). Other databases (e.g., WALS (Dryer & Haspelmath, 2013), AUTOTYP (Bickel et al., 2023), Lexibank (List et al., 2023)) exhibit features that are coded for different sets of languages, resulting in sparse language-feature matrices. Combining data from various databases via language identifiers like glottocodes (Hammarström et al., 2024) usually increases sparsity because the language samples do not match.

When datasets are too large or too sparse for computational applications, it can thus be necessary to generate and subsequently operate on a subset of the data represented in a smaller and denser matrix. Thereby, researchers might be particularly interested in maintaining taxonomic diversity in the languages represented, preferentially removing languages belonging to clades represented by many other languages in the sample and penalizing the removal of language isolates or languages which represent small language families.

While certain packages exist to generate sub-matrices from varying input matrices according to principled criteria (e.g., `admmDensestSubmatrix` (Ames & Bombina, 2019), which identifies the densest sub-matrix of an input graph of a specified size; or `FSelector` (Romanski et al., 2023) and `varrank` (Kratzer & Furrer, 2022), which perform attribute subset selection based on various tests and entropy measures to identify the most relevant attributes of a data input), densify adds sensitivity to taxonomic structure and generally more flexibility in parameter settings for the user. The algorithm focuses both on the removal of rows and columns and does not require the size of the sub-matrix to be specified a priori.

The software therefore addresses recurring problems researchers face when working with typological linguistic data, and it was primarily designed to handle such data frames (with rows representing languages and columns representing typological features). However, it will run on any data frame with rows representing any entities with or without taxonomic structure and columns representing variables. It may thus be of use for other applications as well.

Usage

The package densify provides the data from The World Atlas of Language Structures (WALS) (Dryer & Haspelmath, 2013) and the language taxonomy provided by Glottolog v. 5.0 (Hammarström et al., 2024) as example data. The accompanying package vignette features a detailed demonstration of the utility and flexibility of densify to subsample an input matrix according to varying needs, using this data.

Input data must be prepared in a dataframe with rows representing taxa or observations with taxon names specified in a dedicated column, and columns representing variables with variable names as column names. Cells that are empty or contain non-applicable or question mark entries must be coded as NA. If densification should be sensitive to taxonomic structure, a taxonomy must be provided as (i) a phylo object (cf. Paradis & Schliep, 2019), (ii) as an adjacency table (i.e. a data frame containing columns id and parent_id, with each row encoding one parent-child relationship), or (iii) by the glottolog_languoids dataframe provided by the package. Every taxon in the input data frame must be included in the taxonomy (as a tip or node).

Iterative pruning of the input matrix is performed by the densify() function, which can be modulated by several parameters (described in detail in the function documentation and the vignette). It returns a specially formatted tibble (a densify_result object), which describes the result of each densification step alongside some summary statistics. The function rank_results() ranks the densify results accoring to a specifiable, subjectively useful scoring function that recruits these summary statistics. The optimal sub-matrix according to the scoring function receives rank 1 and can be directly retrieved by the function prune(). visualize(), an alias of plot(), visually compares the quality scores between different pruning steps. The default scoring function used by rank_results(), prune() and visualize() maximizes the product of the number of available data points and the overall coding density, but it can be adjusted to include other measures and trade off their relative weight.

Conclusions

The R package densify provides users with a flexible and explicit method to generate sub-matrices from an input matrix in a mathematically principled way. The package documents case examples using a standard sparse linguistic dataset (WALS) and the standard linguistic taxonomy provided by Glottolog.

Examples and further usage details for this software are found in the vignette hosted in the software repository on GitHub.

Acknowledgements

The authors declare that there are no conflicts of interest.

References

Ames, B., & Bombina, P. (2019). *admmDensestSubmatrix: Alternating direction method of multipliers to solve dense dubmatrix problem*. https://doi.org/10.32614/cran.package.admmdensestsubmatrix

Bickel, B., Nichols, J., Zakharko, T., Witzlack-Makarevich, A., Hildebrandt, K., Rießler, M., Bierkandt, L., Zúñiga, F., & Lowe, J. B. (2023). *The AUTOTYP database (v1.1.1)*. https://doi.org/10.5281/zenodo.7976754

Dryer, M. S., & Haspelmath, M. (Eds.). (2013). *WALS online (v2020.3)* [Data set]. Zenodo. https://doi.org/10.5281/zenodo.7385533

Hammarström, H., Forkel, R., Haspelmath, M., & Bank, S. (2024). *Glottolog/glottolog: Glottolog database 5.0* (Version v5.0) [Data set]. Zenodo. https://doi.org/10.5281/zenodo.10804357

Kratzer, G., & Furrer, R. (2022). *Varrank: Heuristics tools based on mutual information for variable ranking*. https://doi.org/10.32614/cran.package.varrank

List, J.-M., Forkel, R., Greenhill, S. J., Rzymski, C., Englisch, J., & Gray, R. D. (2023). *Lexibank analysed* (Version v1.0) [Data set]. Zenodo. https://doi.org/10.5281/zenodo.7836668

Moran, S., & McCloy, D. (2019). *cldf-datasets/phoible: PHOIBLE 2.0.1 as CLDF dataset* (Version v2.0.1) [Data set]. Zenodo. https://doi.org/10.5281/zenodo.2677911

Paradis, E., & Schliep, K. (2019). Ape 5.0: An environment for modern phylogenetics and evolutionary analyses in R. *Bioinformatics*, *35*, 526–528. https://doi.org/10.1093/bioinformatics/bty633

Romanski, P., Kotthoff, L., & Schratz, P. (2023). *FSelector: Selecting attributes*. https://doi.org/10.32614/cran.package.fselector

Skirgård, H., Haynie, H. J., Hammarström, H., Blasi, D. E., Collins, J., Latarche, J., Lesage, J., Weber, T., Witzlack-Makarevich, A., Dunn, M., Reesink, G., Singer, R., Bowern, C., Epps, P., Hill, J., Vesakoski, O., Abbas, N. K., Ananth, S., Auer, D., … Gray, R. D. (2023). *Grambank v1.0* (Version v1.0). Zenodo. https://doi.org/10.5281/zenodo.7740140