The Journal of Open Source Software

FDApy: a Python package for functional data

Steven Golovkine[1]

1 MACSI, Department of Mathematics and Statistics, University of Limerick, Limerick, Ireland

DOI: 10.21105/joss.07526

Software

- Review
- Repository
- Archive

Editor: Marcel Stimberg

Reviewers:

- @quantgirluk
- @vnmabus

Submitted: 28 October 2024
Published: 28 February 2025

Summary

Functional data analysis (FDA) is a statistical methodology for analyzing data that can be characterized as functions. These functions could represent measurements taken over time, space, frequency, probability, etc. The goal of FDA is to extract meaningful information from these functions and to model their behavior. See, e.g., J. O. Ramsay & Silverman (2005),Horváth & Kokoszka (2012), and Kokoszka & Reimherr (2017) for some references on FDA. FDA has been successfully applied in different contexts, such as identifying patterns of movements in sport biomechanics (Warmenhoven et al., 2019), analyzing changes in brain activity in neuroscience (Song & Kim, 2022), fault detection of batch processes (Wang & Yao, 2015) or in autonomous driving (Golovkine et al., 2022). In this paper, we introduce FDApy, a library developed for the FDA community and Python users, designed to facilitate the manipulation and processing of (multivariate) functional data.

Statement of need

In order to apply FDA to real datasets, there is a need for appropriate softwares with up-to-date methodological implementation and easy addition of new theoretical developments. The seminal R package for FDA is fda (J. Ramsay et al., 2023), based on work cited in J. O. Ramsay & Silverman (2005) and J. Ramsay et al. (2009). Most of the R packages that implement FDA methods are highly specialized and are built upon fda. For example, one may cite FDboost (Brockhaus et al., 2020) and refund (Goldsmith et al., 2023) for regression and classification, funFEM (Bouveyron, 2021) and funLBM (Bouveyron & Schmutz, 2022) for clustering or fdasrvf (Tucker & Stamm, 2023) for functional data registration. For most packages, the functional data are however restricted to univariate functional data that are well described by their coefficients in a given basis of functions. The funData package (Happ-Kurz, 2020) aims to provide a unified framework to handle univariate and multivariate functional data defined on different dimensional domains. Sparse functional data are also considered. The MFPCA (Happ-Kurz, 2022) package, built on top of the funData package, implements multivariate functional principal components analysis (MFPCA) for data defined on different dimensional domains (Happ & Greven, 2018). Consider looking at the CRAN webpage[1] on functional data to have a complete overview of the R packages.

Concerning the Python community, there are only few packages that are related to FDA. One may cite sktime (Löning et al., 2022) and tslearn (Tavenard et al., 2020) that provide tools for the analysis of time series as a scikit-learn compatible API. They implement specific time series methods such as DTW-based ones or shapelets learning. The only one that develops specific methods for FDA is scikit-fda (Ramos-Carreño et al., 2024). In particular, it implements diverse registration techniques as well as statistical data depths for functional data. However, most of the methods are for one-dimensional data and, in most cases, they only accept multivariate functional data defined on the same domain.

[1]https://cran.r-project.org/web/views/FunctionalData.html

FDApy supports the analysis of diverse types of functional data (densely or irregularly sampled, multivariate and multidimensional), represented over a grid of points or using a basis of functions. It implements dimension reduction techniques and smoothing functionalities. A large simulation toolbox, based on basis decomposition, is provided. By providing a flexible and robust toolset for functional data analysis, it aims to support researchers and practitioners in uncovering insights from complex functional datasets.

FDApy was used in Golovkine et al. (2022), Yoshida et al. (2022), Golovkine et al. (2023) and Nguyen (2024) and is also presented in the author's doctoral dissertation.

Code Quality and Documentation

FDApy is hosted on GitHub[2]. Examples and API documentation are available on the platform Read the Docs[3]. We provide installation guides, algorithm introductions, and examples of using the package. The package is available on Linux, macOS and Windows for Python $3.9 - 3.11$. It can be installed with `pip install FDApy`.

To ensure high code quality, all implementations adhere to the PEP8 code style (van Rossum et al., 2001), enforced by `flake8`, the code formatter `black` and the static analyzer `prospector`. The documentation is provided through docstrings using the NumPy conventions and build using Sphinx. The code is accompanied by unit tests covering 94% of the lines that are automatically executed in a continuous integration workflow upon commits.

Acknowledgements

Steven Golovkine wishes to thank Groupe Renault and the ANRT (French National Association for Research and Technology) for their financial support via the CIFRE convention No. 2017/1116. Steven Golovkine is partially supported by Science Foundation Ireland under Grant No. 19/FFP/7002 and co-funded under the European Regional Development Fund.

References

Bouveyron, C. (2021). *funFEM: Clustering in the discriminative functional subspace*. https://doi.org/10.32614/cran.package.funfem

Bouveyron, C., & Schmutz, J. J. and A. (2022). *funLBM: Model-based co-clustering of functional data*. https://doi.org/10.32614/cran.package.funlbm

Brockhaus, S., Rügamer, D., & Greven, S. (2020). Boosting functional regression models with FDboost. *Journal of Statistical Software*, *94*, 1–50. https://doi.org/10.18637/jss.v094.i10

Goldsmith, J., Scheipl, F., Huang, L., Wrobel, J., Di, C., Gellar, J., Harezlak, J., McLean, M. W., Swihart, B., Xiao, L., Crainiceanu, C., Reiss, P. T., Chen, Y., Greven, S., Huo, L., Kundu, M. G., Park, S. Y., Miller, D. L., Staicu, A.-M., … Li, Z. (2023). *Refund: Regression with functional data*. https://doi.org/10.32614/cran.package.refund

Golovkine, S., Gunning, E., Simpkin, A. J., & Bargary, N. (2023). *On the use of the Gram matrix for multivariate functional principal components analysis*. arXiv. https://doi.org/10.48550/arXiv.2306.12949

Golovkine, S., Klutchnikoff, N., & Patilea, V. (2022). Clustering multivariate functional data using unsupervised binary trees. *Computational Statistics & Data Analysis*, *168*, 107376. https://doi.org/10.1016/j.csda.2021.107376

[2] https://github.com/StevenGolovkine/FDApy
[3] https://fdapy.readthedocs.io

Happ, C., & Greven, S. (2018). Multivariate functional principal component analysis for data observed on different (dimensional) domains. *Journal of the American Statistical Association*, *113*(522), 649–659. https://doi.org/10.1080/01621459.2016.1273115

Happ-Kurz, C. (2020). Object-oriented software for functional data. *Journal of Statistical Software*, *93*, 1–38. https://doi.org/10.18637/jss.v093.i05

Happ-Kurz, C. (2022). *MFPCA: Multivariate functional principal component analysis for data observed on different dimensional domains*. https://doi.org/10.32614/cran.package.mfpca

Horváth, L., & Kokoszka, P. (2012). *Inference for functional data with applications* (Vol. 200). Springer. https://doi.org/10.1007/978-1-4614-3655-3

Kokoszka, P., & Reimherr, M. (2017). *Introduction to functional data analysis*. Chapman and Hall/CRC. https://doi.org/10.1201/9781315117416

Löning, M., Király, F., Bagnall, T., Middlehurst, M., Ganesh, S., Oastler, G., Lines, J., Walter, M., ViktorKaz, Mentel, L., chrisholder, Tsaprounis, L., RNKuhns, Parker, M., Owoseni, T., Rockenschaub, P., danbartl, jesellier, eenticott-shell, … rice, B. (2022). *Sktime/sktime: V0.13.4*. Zenodo. https://doi.org/10.5281/zenodo.7117735

Nguyen, C. M. A. (2024). *Learning domain-specific cameras*. https://purl.stanford.edu/zm136ny2176

Ramos-Carreño, C., Torrecilla, J. L., Carbajo-Berrocal, M., Marcos, P., & Suárez, A. (2024). Scikit-fda: A Python package for functional data analysis. *Journal of Statistical Software*, *109*, 1–37. https://doi.org/10.18637/jss.v109.i02

Ramsay, J. O., & Silverman, B. W. (2005). *Functional data analysis*. Springer. https://doi.org/10.1007/b98888

Ramsay, J., Hooker, G., & Graves, S. (2009). *Functional data analysis with R and MATLAB*. Springer. https://doi.org/10.1007/978-0-387-98185-7

Ramsay, J., Hooker, G., & Graves, S. (2023). *Fda: Functional data analysis*. https://doi.org/10.32614/cran.package.fda

Song, J., & Kim, K. (2022). Sparse multivariate functional principal component analysis. *Stat*, *11*(1), e435. https://doi.org/10.1002/sta4.435

Tavenard, R., Faouzi, J., Vandewiele, G., Divo, F., Androz, G., Holtz, C., Payne, M., Yurchak, R., Rußwurm, M., Kolar, K., & Woods, E. (2020). Tslearn, a machine learning toolkit for time series data. *Journal of Machine Learning Research*, *21*(118), 1–6. http://jmlr.org/papers/v21/20-091.html

Tucker, J. D., & Stamm, A. (2023). *Fdasrvf: Elastic functional data analysis*. https://doi.org/10.32614/cran.package.fdasrvf

van Rossum, G., Warsaw, B., & Coghlan, A. (2001). *PEP8 - style guide for Python code* (PEP No. 8). https://peps.python.org/pep-0008/

Wang, H., & Yao, M. (2015). Fault detection of batch processes based on multivariate functional kernel principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, *149*, 78–89. https://doi.org/10.1016/j.chemolab.2015.09.018

Warmenhoven, J., Cobley, S., Draper, C., Harrison, A., Bargary, N., & Smith, R. (2019). Bivariate functional principal components analysis: Considerations for use with multivariate movement signatures in sports biomechanics. *Sports Biomechanics*, *18*(1), 10–27. https://doi.org/10.1080/14763141.2017.1384050

Yoshida, K., Commandeur, D., Hundza, S., & Klimstra, M. (2022). Detecting differences in gait initiation between older adult fallers and non-fallers through multivariate functional principal component analysis. *Journal of Biomechanics*, *144*, 111342. https://doi.org/10.

1016/j.jbiomech.2022.111342