The Journal of Open Source Software

PyCCEA: A Python package of cooperative co-evolutionary algorithms for feature selection in high-dimensional data

Pedro Vinícius A. B. Venâncio [1,3¶] **and Lucas S. Batista** [2,3]

1 Graduate Program in Electrical Engineering, Universidade Federal de Minas Gerais, Brazil **2** Departament of Electrical Engineering, Universidade Federal de Minas Gerais, Brazil **3** Operations Research and Complex Systems Laboratory, Universidade Federal de Minas Gerais, Brazil **¶** Corresponding author

DOI: 10.21105/joss.08348

Software

- Review
- Repository
- Archive

Editor: Blake Rayfield

Reviewers:

- @geoffreyp
- @maximtrp

Submitted: 13 April 2025
Published: 10 August 2025

Summary

Feature selection is a critical preprocessing step in many machine learning pipelines, particularly when dealing with high-dimensional datasets commonly found in domains such as genomics, text mining, and image analysis. However, many feature selection techniques, such as heuristic search methods and evolutionary algorithms, struggle to maintain predictive performance and interpretability as the dimensionality of data increases (Theng & Bhoyar, 2024).

Cooperative co-evolutionary algorithms (CCEAs) offer a promising approach for tackling this challenge by dividing the high-dimensional space into multiple tractable low-dimensional subcomponents. Each subcomponent is evolved independently using a subpopulation, and candidate solutions are evaluated based on their collaboration with representatives from other subpopulations (Ma et al., 2018).

`PyCCEA` is a dedicated and extensible package for implementing CCEAs aimed at feature selection in high-dimensional data (i.e., more than 1000 features). As illustrated in Figure 1, it is structured around modular components that encapsulate key elements of the CCEA paradigm, including decomposition strategies, cooperation schemes, fitness evaluation methods, and evolutionary optimizers. This design allows researchers and practitioners to easily replicate existing approaches or prototype novel algorithms by composing interchangeable modules (Venâncio & Batista, 2025).

The package currently adopts a wrapper-based evaluation approach, where subsets of features are assessed using machine learning models to guide the search process. However, this design choice is not restrictive. `PyCCEA` has been structured to accommodate future extensions supporting other evaluation paradigms, such as filter-based, embedded, or hybrid methods. These alternatives are under consideration for upcoming releases to broaden the range of applicable scenarios and research directions.

In addition to its evolutionary components, `PyCCEA` integrates machine learning and data processing functionalities built on top of widely adopted libraries, including `scikit-learn` (Pedregosa et al., 2011) and `Pandas` (McKinney, 2010). In view of this, users can take advantage of built-in support for models (e.g., KNN, SVM, Random Forest), metrics (e.g., accuracy, F_1-score), and data operations (e.g., normalization, cross-validation, splitting, preprocessing). This tight integration ensures compatibility with existing machine learning workflows while maintaining flexibility and ease of use. For comprehensive details on installation, usage, and API, please refer to the official PyCCEA documentation.

To further support empirical studies, `PyCCEA` includes a benchmarking module composed of well-known datasets (e.g., DLBCL, lung cancer) (Kelly et al., 2017) and baseline CCEAs (e.g., CCEAFS (A. B. Rashid et al., 2020), CCFSRFG1 (A. Rashid et al., 2020)). This enables standardized evaluation and comparison of new strategies in a reproducible and transparent manner. To the best of our knowledge, PyCCEA is the first open-source package specifically designed for cooperative co-evolution in the context of feature selection, providing a solid foundation for research and practical application in high-dimensional machine learning problems. In contrast, other popular packages such as scikit-learn (Pedregosa et al., 2011), MLXtend (Raschka, 2018), and Feature-engine (Galli, 2021) rely on more traditional approaches, including statistical filters (e.g., ANOVA, correlation, mutual information), model-based embedded methods (e.g., tree importance, model coefficients), and deterministic or greedy searches (e.g., sequential or exhaustive search, recursive elimination). These methods, while effective in low to moderate dimensions, often struggle with scalability and performance in high-dimensional spaces, where complex feature interactions and search space combinatorics demand more flexible and exploratory algorithms such as CCEAs.

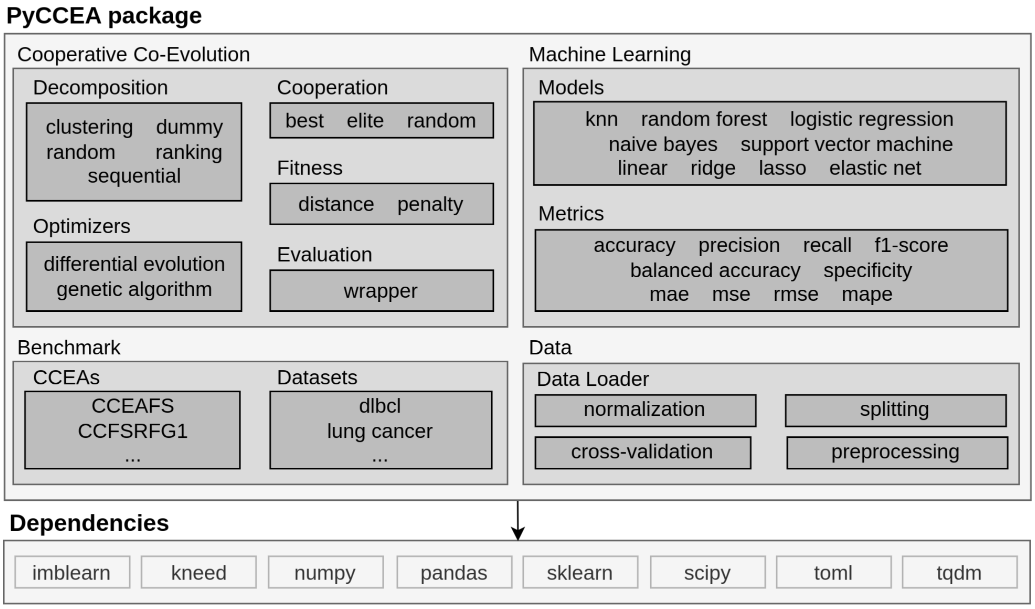

Figure 1: An overview of the PyCCEA package, its modules and underlying Python dependencies.

Statement of need

Despite the growing interest in cooperative co-evolutionary algorithms for feature selection and their promising results (Song et al., 2020), there is currently no publicly available software package that consolidates these techniques into a reusable and extensible tool. Existing research typically relies on custom (Firouznia et al., 2023; A. Rashid et al., 2020; A. B. Rashid et al., 2020) or unpublished implementations (Song et al., 2020; Zhou et al., 2024), which makes it difficult to reproduce results, compare methods, or build upon previous work.

PyCCEA addresses this gap by providing a well-organized, research-focused implementation of cooperative co-evolutionary algorithms tailored to feature selection. It incorporates widely used strategies from the literature and encourages standardization in experimental design and evaluation. By enabling consistent benchmarking and facilitating the development of new strategies, PyCCEA supports both methodological innovation and practical application in high-dimensional machine learning problems. Its release lowers the barrier to entry for researchers and practitioners, accelerating progress in the field of feature selection using evolutionary computation.

Recent research has already leveraged PyCCEA as a foundational framework for implementing and evaluating cooperative co-evolutionary strategies in high-dimensional classification problems (Venâncio & Batista, 2025), demonstrating its effectiveness in supporting reproducible and extensible scientific research. To expand the framework's usability beyond classification, we have recently introduced support for regression tasks, further broadening its applicability to diverse problem domains. These ongoing developments also seek to promote greater community engagement by making the framework increasingly adaptable, accessible, and conducive to collaborative research and development.

Acknowledgements

This work was supported by Brazilian agencies FAPEMIG (Research Support Foundation of the State of Minas Gerais), CNPq (The National Council for Scientific and Technological Development), CAPES (Coordination for the Improvement of Higher Education Personnel) – Finance Code 001, and the Operations Research and Complex Systems Laboratory (ORCS Lab./UFMG). Professor Lucas S. Batista is a FAPEMIG-CNPQ scholarship holder (process APQ-06716-24).

References

Firouznia, M., Ruiu, P., & Trunfio, G. A. (2023). Adaptive cooperative coevolutionary differential evolution for parallel feature selection in high-dimensional datasets. *The Journal of Supercomputing*, 1–30. https://doi.org/10.1007/s11227-023-05226-y

Galli, S. (2021). Feature-engine: A python package for feature engineering for machine learning. *The Journal of Open Source Software*, *6*(65), 3642. https://doi.org/10.21105/joss.03642

Kelly, M., Longjohn, R., & Nottingham, K. (2017). *The UCI machine learning repository*. University of California, Irvine, School of Information; Computer Sciences. https://doi.org/10.17616/R3T91Q

Ma, X., Li, X., Zhang, Q., Tang, K., Liang, Z., Xie, W., & Zhu, Z. (2018). A survey on cooperative co-evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, *23*(3), 421–441. https://doi.org/10.1109/TEVC.2018.2868770

McKinney, W. (2010). Data structures for statistical computing in Python. In Stéfan van der Walt & Jarrod Millman (Eds.), *Proceedings of the 9th Python in Science Conference* (pp. 51–56). https://doi.org/10.25080/Majora-92bf1922-00a

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830. https://doi.org/10.48550/arXiv.1201.0490

Raschka, S. (2018). MLxtend: Providing machine learning and data science utilities and extensions to python's scientific computing stack. *The Journal of Open Source Software*, *3*(24), 638. https://doi.org/10.21105/joss.00638

Rashid, A. B., Ahmed, M., Sikos, L. F., & Haskell-Dowland, P. (2020). A novel penalty-based wrapper objective function for feature selection in big data using cooperative co-evolution. *IEEE Access*, *8*, 150113–150129. https://doi.org/10.1109/ACCESS.2020.3016679

Rashid, A., Ahmed, M., Sikos, L. F., & Haskell-Dowland, P. (2020). Cooperative co-evolution for feature selection in big data with random feature grouping. *Journal of Big Data*, *7*(1), 1–42. https://doi.org/10.1186/s40537-020-00381-y

Song, X.-F., Zhang, Y., Guo, Y.-N., Sun, X.-Y., & Wang, Y.-L. (2020). Variable-size cooperative coevolutionary particle swarm optimization for feature selection on high-dimensional data. *IEEE Transactions on Evolutionary Computation*, *24*(5), 882–895. https://doi.org/10.1109/TEVC.2020.2968743

Theng, D., & Bhoyar, K. K. (2024). Feature selection techniques for machine learning: A survey of more than two decades of research. *Knowledge and Information Systems*, *66*(3), 1575–1637. https://doi.org/10.1007/s10115-023-02010-5

Venâncio, P. V. A. B., & Batista, L. S. (2025). A self-tuning decomposition strategy in cooperative co-evolutionary algorithms for high-dimensional feature selection. *Knowledge-Based Systems*, *316*, 113327. https://doi.org/10.1016/j.knosys.2025.113327

Zhou, J., Zheng, H., Li, S., Hao, Q., Zhang, H., Gao, W., & Wang, X. (2024). A knowledge-guided competitive co-evolutionary algorithm for feature selection. *Applied Sciences*, *14*(11), 4501. https://doi.org/10.3390/app14114501