OXFORD (GIGA)n SCIENCE

GigaScience, 2024, **13**, 1–9
DOI: 10.1093/gigascience/giad111
TECH NOTE

Machine Learning Made Easy (MLme): a comprehensive toolkit for machine learning–driven data analysis

Akshay Akshay [1,2,†], Mitali Katoch [3,†], Navid Shekarchizadeh [4,5], Masoud Abedi [4], Ankush Sharma [6,7], Fiona C. Burkhard [1,8], Rosalyn M. Adam [9,10,11], Katia Monastyrskaya [1,8], and Ali Hashemi Gheinani [1,8,9,10,11,*]

[1]Functional Urology Research Group, Department for BioMedical Research DBMR, University of Bern, 3008 Bern, Switzerland
[2]Graduate School for Cellular and Biomedical Sciences, University of Bern, 3012 Bern, Switzerland
[3]Institute of Neuropathology, Universitätsklinikum Erlangen, Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), 91054 Erlangen, Germany
[4]Department of Medical Data Science, Leipzig University Medical Centre, 04107 Leipzig, Germany
[5]Center for Scalable Data Analytics and Artificial Intelligence (ScaDS.AI) Dresden/Leipzig, 04105 Leipzig, Germany
[6]KG Jebsen Centre for B-cell Malignancies, Institute for Clinical Medicine, University of Oslo, 0318 Oslo, Norway
[7]Department of Cancer Immunology, Institute for Cancer Research, Oslo University Hospital, 0310 Oslo, Norway
[8]Department of Urology, Inselspital University Hospital, 3010 Bern, Switzerland
[9]Urological Diseases Research Center, Boston Children's Hospital, 02115 Boston, MA, USA
[10]Department of Surgery, Harvard Medical School, 02115 Boston, MA, USA
[11]Broad Institute of MIT and Harvard, Cambridge, 02142 MA, USA
*Correspondence address: Ali Hashemi Gheinani, Urological Diseases Research Center, Boston Children's Hospital, Harvard Medical School and Broad Institute of MIT and Harvard, Cambridge, MA, USA. E-mail: Ali.HashemiGheinani@childrens.harvard.edu
†Contributed equally.

Abstract

Background: Machine learning (ML) has emerged as a vital asset for researchers to analyze and extract valuable information from complex datasets. However, developing an effective and robust ML pipeline can present a real challenge, demanding considerable time and effort, thereby impeding research progress. Existing tools in this landscape require a profound understanding of ML principles and programming skills. Furthermore, users are required to engage in the comprehensive configuration of their ML pipeline to obtain optimal performance.

Results: To address these challenges, we have developed a novel tool called Machine Learning Made Easy (MLme) that streamlines the use of ML in research, specifically focusing on classification problems at present. By integrating 4 essential functionalities—namely, Data Exploration, AutoML, CustomML, and Visualization—MLme fulfills the diverse requirements of researchers while eliminating the need for extensive coding efforts. To demonstrate the applicability of MLme, we conducted rigorous testing on 6 distinct datasets, each presenting unique characteristics and challenges. Our results consistently showed promising performance across different datasets, reaffirming the versatility and effectiveness of the tool. Additionally, by utilizing MLme's feature selection functionality, we successfully identified significant markers for $CD8^+$ naive (BACH2), $CD16^+$ (CD16), and $CD14^+$ (VCAN) cell populations.

Conclusion: MLme serves as a valuable resource for leveraging ML to facilitate insightful data analysis and enhance research outcomes, while alleviating concerns related to complex coding scripts. The source code and a detailed tutorial for MLme are available at https://github.com/FunctionalUrology/MLme.

Keywords: machine learning, classification problems, data analysis, AutoML, visualization

Key points

- MLme is a novel tool that simplifies machine learning (ML) for researchers by integrating Data Exploration, AutoML, CustomML, and Visualization functionalities.
- MLme improves efficiency and productivity by streamlining the ML workflow and eliminating the need for extensive coding efforts.
- Rigorous testing on diverse datasets demonstrates MLme's promising performance in classification problems.
- MLme provides intuitive interfaces for data exploration, automated ML, customizable ML pipelines, and result visualization.
- Future developments aim to expand MLme's capabilities to include support for unsupervised learning, regression, hyperparameter tuning, and integration of user-defined algorithms.

Introduction

In the realm of research, machine learning (ML) has emerged as a vital resource for analyzing intricate datasets that conventional statistical approaches struggle to interpret [1–5]. However, the integration of ML into research presents a multitude of challenges. Foremost, the construction and execution of an effective ML pipeline can be daunting, requiring deep domain expertise, ex-

Received: July 4, 2023. **Revised:** September 20, 2023. **Accepted:** December 8, 2023

tensive technical knowledge, and proficient programming skills. In addition, the utilization of ML techniques demands a comprehensive understanding of the underlying principles to ensure that the trained models are unbiased and transparent.

Multiple tools have been developed to streamline the process of building and executing ML pipelines (Supplementary Table S1) [6–16]. These tools often require a significant level of coding proficiency and extensive configuration to achieve optimal effectiveness. Additionally, many of these tools serve as algorithm recommenders, functioning by running multiple ML algorithms on user-provided data and providing model performance metrics. However, this approach can limit user input and guidance, as the tools tend to prioritize automated decision-making rather than allowing users to actively participate in the process. As a result, tailoring the ML models to specific research needs and ensuring that the models align with domain knowledge and expertise can be challenging. This lack of flexibility and limited user control potentially hinder the accuracy and applicability of the research outcomes.

Machine Learning Made Easy (MLme) is a comprehensive solution aimed at bridging the gap between researchers and the inherent technical complexities of ML. It facilitates the adoption of ML techniques by simplifying the ML workflow and minimizing the typically steep learning curve associated with ML. Through its intuitive interfaces, MLme enhances accessibility and usability for researchers of varying levels of technical expertise (Fig. 1).

MLme offers 4 important components: Data Exploration, AutoML, CustomML, and Visualization, each serving a specific purpose in understanding and extracting meaningful information from the data within the ML workflow. Through the intuitive Data Exploration feature, users easily examine their datasets and gain preliminary understanding using an interactive interface. For advanced users, the CustomML interface within MLme provides a flexible platform to design and develop tailor-made ML pipelines that align with their specific research requirements. Furthermore, it facilitates effortless interpretation and analysis of results with rich visualization capabilities.

Key Features of MLme

MLme is a multifaceted toolkit that equips researchers with the functionalities necessary to effectively utilize ML in their research. It consists of 4 distinct web interfaces, each tailored to address specific research needs, ensuring a versatile and comprehensive experience for users.

Data Exploration

The Data Exploration feature of MLme allows users to upload their datasets and explore them using a range of statistical visualizations, such as density plots, scatter matrix plots, area plots, and class distribution plots (Supplementary Fig. S1A). These visualizations and statistical summaries enable users to gain a comprehensive understanding of their data, including patterns and trends within the data, data distribution, and potential outliers. A density plot, for instance, can reveal how data are distributed, while a scatter matrix plot can identify potential correlations. Class distribution plots are particularly useful for comprehending the balance of target classes within the dataset, which can be crucial when designing a machine learning model.

Overall, the Data Exploration feature enables users to efficiently explore their datasets and acquire initial insights into their data. This knowledge can inform subsequent modeling decisions, ensuring that users are using the most appropriate modeling techniques for their specific dataset.

AutoML

The AutoML feature in MLme enables users to effortlessly extract meaningful information from their datasets using ML, even without extensive technical expertise (Supplementary Fig. S1B). With a preconfigured ML pipeline (Fig. 2), the AutoML handles essential preprocessing steps such as data resampling, scaling, and feature selection [17]. These steps ensure that the input data are properly prepared for ML algorithms, enhancing the performance and reliability of subsequent trained models. The AutoML conducts training and evaluation of multiple classification models, including a dummy classifier. By employing diverse models, users gain a comprehensive understanding of their data and can identify the most effective algorithms for their specific dataset.

After the pipeline is completed, the AutoML offers users various options for examining and interpreting the results. These options include intuitive and interactive plots, which help users gain a deeper understanding of the performance characteristics of the models. Additionally, users have the flexibility to download the results and explore them further using the Visualization interface at their convenience.

CustomML

The CustomML feature of MLme empowers users with moderate to advanced knowledge of the ML domain to design and customize an ML pipeline that caters to their specific research needs (Supplementary Fig. S2A). With its user-friendly and intuitive interface, users can easily include or exclude steps and algorithms using a simple toggle button. This eliminates the worry about writing complex programming scripts and allows focusing on selecting the most suitable steps and algorithms for the dataset.

CustomML offers an extensive range of preprocessing options, including 7 algorithms for data resampling, 19 algorithms for scaling, and a diverse array of feature selection algorithms to select relevant features from the dataset. Moreover, with 16 classification algorithms available, users can refine their pipeline to align with their research requirements. To provide a comprehensive understanding of the trained model's performance, CustomML supports 10 different evaluation methods and 14 evaluation metrics.

The customization options of CustomML are enhanced by allowing users to select the parameters value for all the provided algorithms, giving them greater control over the behavior of their developed pipeline. Once the pipeline is designed, it can be conveniently downloaded and executed either locally or on a cluster, offering flexibility in computing resources. The CustomML-generated ML pipeline produces a pickle file (.pkl) as an output upon completion, which contains all the results from the pipeline. This file can be uploaded to the Visualization interface, enabling users to interpret these results using various plots.

Visualization

The Visualization feature in MLme allows users to effortlessly interpret their results without the need for advanced programming skills or expertise in data visualization (Supplementary Fig. S2B). It provides a comprehensive range of plots and tables, covering fundamental as well as advanced options such as bar plots, heatmaps, and spider plots. These diverse visualization tools facilitate effective comparison of trained model performance.

Furthermore, this feature allows users to customize the appearance of their plots by selecting from over 50 different color

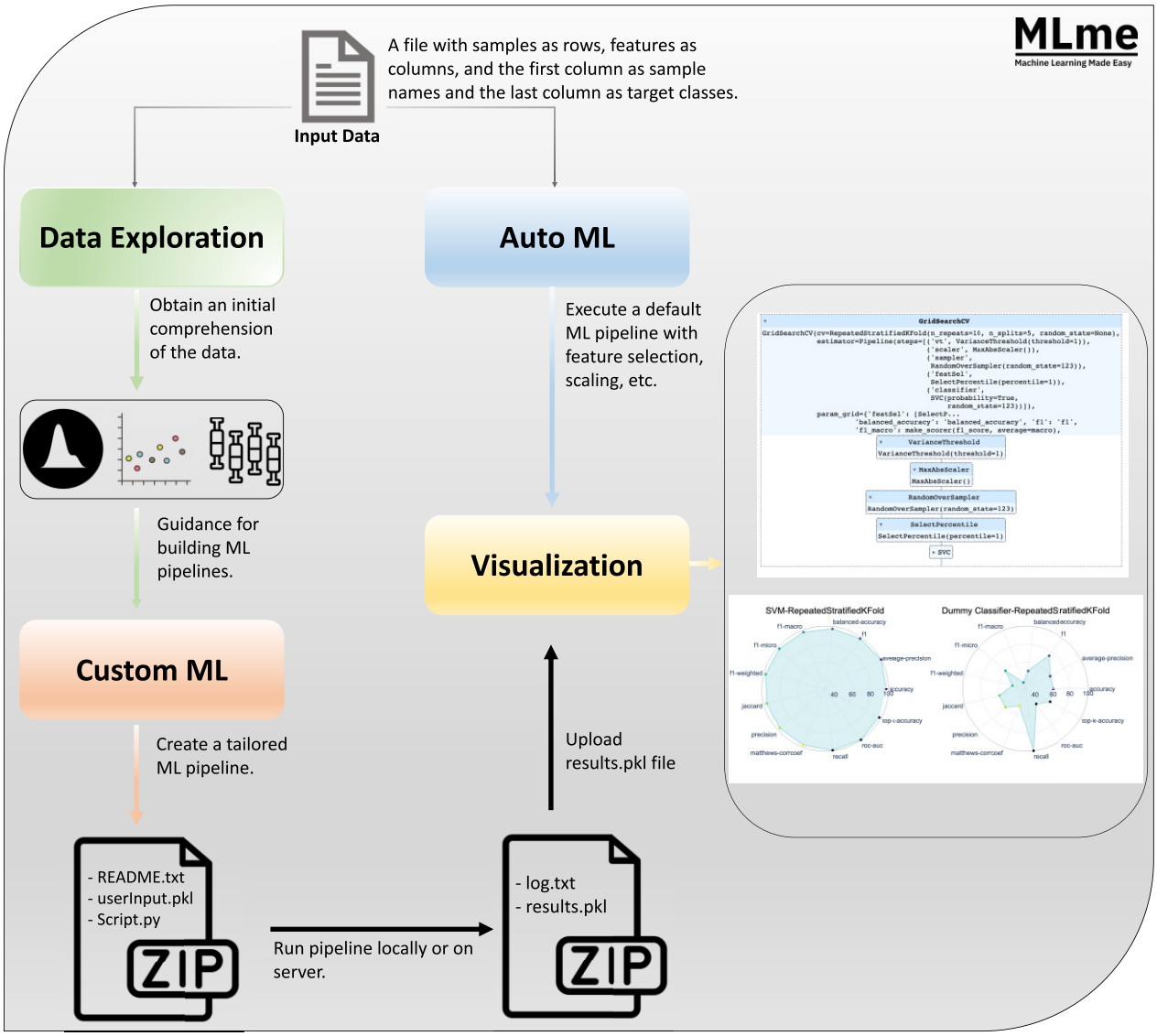

Figure 1: Graphical abstract. The input data for Machine Learning Made Easy (MLme) is a file with samples as rows and features as columns, with sample names in the first column and target classes in the last column. MLme provides various features to enhance usability. The data exploration feature enables users to explore the data and gain initial insights. For advanced users, the custom ML feature allows the creation of custom ML pipelines. Upon execution, MLme generates a compressed zip file containing inputParameter.pkl, script.py, and README.txt. Alternatively, users can opt for the AutoML feature, which applies a default ML pipeline to the input file. Both CustomML and AutoML produce a results.pkl file, which can be further analyzed using the visualization feature.

palettes. Additionally, all generated plots are of high quality and are downloadable in high resolution, ensuring they are suitable for publication purposes. Supplementary Fig. S11 showcases the available list of algorithms and diverse plot types within MLme for various machine learning stages.

Use Cases

Dataset selection criteria

The MLme application is evaluated using 7 distinct datasets (Supplementary Table S2) that are carefully chosen to ensure robustness. Factors such as sample size, diversity, class imbalance, and dimensionality are considered during the selection process. The selected datasets vary in sample size and diversity, providing a comprehensive assessment of the MLme application's performance across different data scales.

This includes datasets of varying sizes, from small (chronic lymphocytic leukemia [CLL] and cervical cancer study) to large (invasive breast carcinoma and body signal datasets), which test the application's scalability and efficiency. Imbalanced datasets, like invasive breast carcinoma (BRCA), are included to evaluate the MLme application's handling of class imbalance and prediction accuracy, which is particularly relevant in real-world scenarios, such as biological research. The datasets also address the challenge of high-dimensional features and low sample sizes, known as the curse of dimensionality. By including such datasets, MLme's ability to handle challenges is thoroughly assessed.

Furthermore, the glass identification dataset was selected as a nonbiological example, offering variation and enabling testing across diverse domains. This dataset, with multiple target classes,

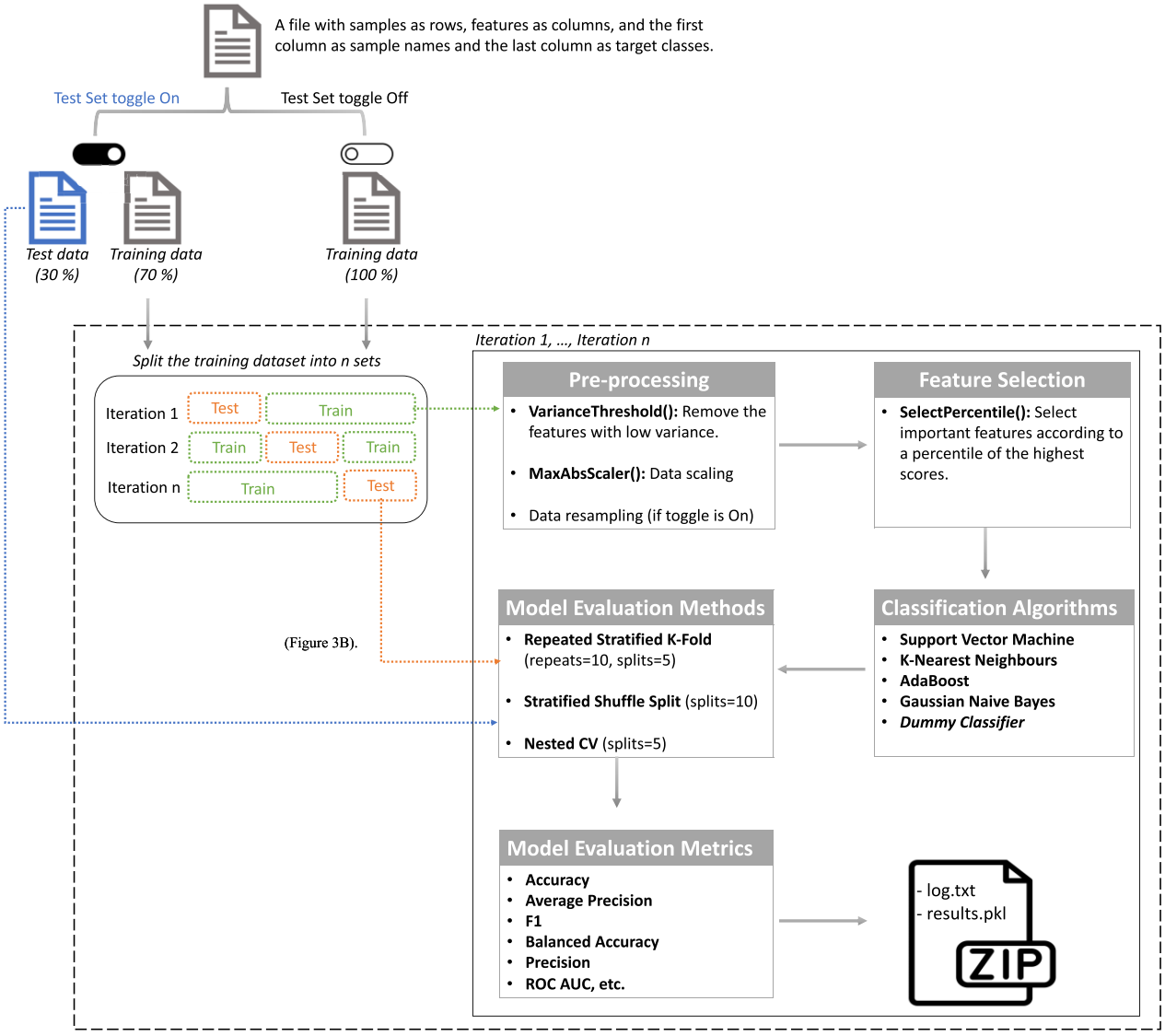

Figure 2: Default ML Pipeline for AutoML. The default ML pipeline can be represented as a flowchart that starts by splitting the input dataset into training and independent test sets, provided the user has activated the test set option. Otherwise, the entire dataset is used for training. In the subsequent step, the training dataset is divided into *n* bins of equal size through stratified sampling. From these bins, k – 1 are designated as training sets while the remainder becomes the test set. In the preprocessing step, low variance features are removed first, followed by data scaling and resampling. Subsequently, the SelectPercentile univariate feature selection method is applied to select important features, and 5 ML classification algorithms are trained. Model performance is assessed on the test set using 3 different methods, and multiple performance metrics are computed. This entire process is repeated for each unique bin in the k-fold corss validation (CV) method. The pipeline outputs a zip file comprising the log .txt and the results.pkl files. The user can examine the results by visualizing the contents of the pickle file using MLme.

allows evaluation of the MLme application's performance in multiclass classification problems.

Dataset descriptions

The first dataset comprised messenger RNA (mRNA) patient data (n = 136) obtained from a study on CLL, which measured their transcriptome profiles [18]. Our objective was to build a model that could classify male and female patients based on their transcriptomic profiles, using the top 5,000 most variable mRNAs (excluding Y chromosome genes). The second dataset was collected from a cervical cancer study that analyzed the expression levels of 714 microRNAs (miRNAs) in human samples (n = 58) [19].

The third and fourth datasets were obtained from The Cancer Genome Atlas (TCGA), consisting of mRNA (n = 1,219) and miRNA (n = 1,207) sequencing data from patients with invasive BRCA, which were retrieved using the TCGAbiolinks package [20] in R. For the BRCA mRNA dataset, we focused only on differentially expressed genes from edgeR (False discovery rate (FDR) ≤0.001 and log fold change >±2) [21]. Our goal was to train a model capable of distinguishing normal and tumor samples for both cervical cancer and TCGA–BRCA datasets.

The fifth dataset consists of single-cell RNA (scRNA) sequencing data obtained from peripheral blood mononuclear cells (PBMCs) that were sequenced using 10× chromium technology [22]. Among all the cell populations described in this study, we specifically utilized the scRNA datasets of $CD8^+$ naive, $CD14^+$, and $CD16^+$ monocytes (n = 1,500) with the goal of identifying distinct markers for each of these cell populations.

The sixth dataset utilized in this study was the widely recognized glass identification dataset ($n = 214$) obtained from the University of California, Irvine ML repository [23]. This dataset comprises 10 distinct features that represent oxide content of glass samples. The primary objective of this dataset is to classify different types of glass based on their oxide content.

The seventh dataset in our study comprises body signal data collected from 100,000 individuals through the National Health Insurance Service in Korea [24]. This dataset includes 21 essential biological signals related to health, such as measurements of systolic blood pressure and total cholesterol levels. Our main goal with this dataset was to determine whether individuals consume alcohol based on the available biological signal information.

Results

To perform a thorough assessment of the MLme functionality, we utilized its CustomML feature to construct distinct ML pipelines for CLL, cervical cancer, body signal, and TCGA datasets. These pipelines entailed various processing steps, including data scaling and resampling using different algorithms, multiple ML classifiers, diverse evaluation methods, and metrics. Additionally, we employed the AutoML feature of MLme to train multiple models for both the PBMC and glass datasets. The top-performing models consistently achieved scores exceeding 90% for all computed metrics across all evaluated datasets except glass identification and body signal datasets. As anticipated, the dummy classifiers performed the worst among all the datasets (Supplementary Figs. S3–S9). Additionally, we conducted a comparative analysis to assess the performance of MLme in comparison to Tree-based Pipeline Optimization Tool (TPOT) and hyperopt-sklearn on these datasets. The fact that all 3 tools demonstrated similar performance (Supplementary Fig. S10) for all datasets, except the glass dataset, underscores the reliability and consistency of the results produced by MLme. For hyperopt-sklearn, we configured it to comprehensively explore all classification algorithms and data transformations within the library while utilizing the tree-structured Parzen estimator algorithm for hyperparameter search. For TPOT, we employed a 5-minute runtime limit, a population size of 50, 5 generations, and default values for all other parameters.

To further demonstrate the applicability of MLme, we utilized its feature selection functionality from AutoML to identify the most important genes for classifying $CD8^+$ naive, $CD14^+$, and $CD16^+$ monocyte cell populations from the PBMC dataset. By selecting the top 10% of the original input of 500 highly variable genes, MLme provided a list of 50 genes that are sufficient for classifying these cell types (Fig. 3A). These 50 genes exhibited a strong correspondence with their respective cell populations, except for 13 ribosomal genes (RPS and RPL) that showed similar expression levels across all 3 cell types.

Among the remaining 37 genes, we discovered classic markers for $CD8^+$ naive cells (TCF7 [25, 26], LEF1 [25], BACH2 [27], BCL11B [28], and THEMIS [29]), which have been previously described in the literature (Fig. 3B). The list also included markers for the $CD16^+$ cell population, such as FCGR3A (CD16), TCF7L2, MS4A7, IFITM3, MTSS1, LST1, and WARS (Fig. 3C), which have been associated with CD16+ cells in previous studies [30, 31]. Furthermore, our marker list encompassed known $CD14^+$ specific genes, including VCAN, a marker of monocytic lineage [32]; CSF3R, previously described in the $CD14^+$ population [33]; and NEAT1 (Fig. 3D). These findings validate the biological relevance of the selected genes and highlight the utility of the MLme tool in biomedical research.

Implementation

The MLme is developed using the *Dash* library [34] in the *Python* [35] programming language. Plots are generated using *Plotly* [36], *matplotlib* [37], and *bokeh* [38] libraries. *Pandas* [39] and *NumPy* [40] libraries are used to handle data storage and processing. The development of the ML pipeline is facilitated by employing the *Scikit-Learn* [41] and *Imbalanced Learn* [42] libraries.

Limitations

Currently, MLme focuses on classification problems since a substantial portion of research questions and available datasets are aligned with the domain of classification. This limitation hinders MLme's applicability to regression or unsupervised learning tasks. Additionally, the tool lacks built-in hyperparameter tuning capabilities. This absence of a key feature may hinder users in fine-tuning their models.

Overall, despite its limitations in handling regression and unsupervised ML problems, the current version of MLme is well equipped to develop pipelines for classification tasks. It is worth noting that users have the flexibility to choose values for all the parameters of a given algorithm through the user interface, to some extent mitigating the impact of the lack of built-in hyperparameter tuning.

Conclusion

Our article introduces a user-friendly tool called MLme, which offers a wide range of functionalities for ML analysis. Its primary goal is to make ML accessible to users of all skill levels by removing technical barriers. With the Data Exploration feature, users can efficiently explore datasets and gain initial insights into their data. The AutoML feature simplifies ML usage, allowing them to leverage ML capabilities without dealing with complex technicalities. Moreover, the CustomML functionality assists in creating personalized pipelines using an intuitive graphical user interface that caters to specific requirements, eliminating the need for coding complexities. Additionally, the Visualization features enable users to interactively explore and understand model performance, without extensive data visualization or coding expertise. In summary, MLme is a powerful and user-friendly tool that empowers researchers to enhance their research outcomes through ML.

However, it is crucial to emphasize that, despite their impressive capabilities, automated ML tools should never be regarded as a replacement for domain expertise. Users of MLme must maintain a strong awareness of the invaluable role that domain knowledge plays when using this software to address real-world problems. Consequently, expertise in the specific field remains irreplaceable, and MLme should be viewed as a complementary tool to augment, rather than replace, human understanding and insights.

Outlook

Despite the limitations mentioned above, there are several promising directions for future development of the MLme. Our primary objective is to expand the capabilities of MLme to include support for unsupervised learning and regression problems. This expansion will greatly enhance the tool's utility and enable its application in a broader range of ML tasks.

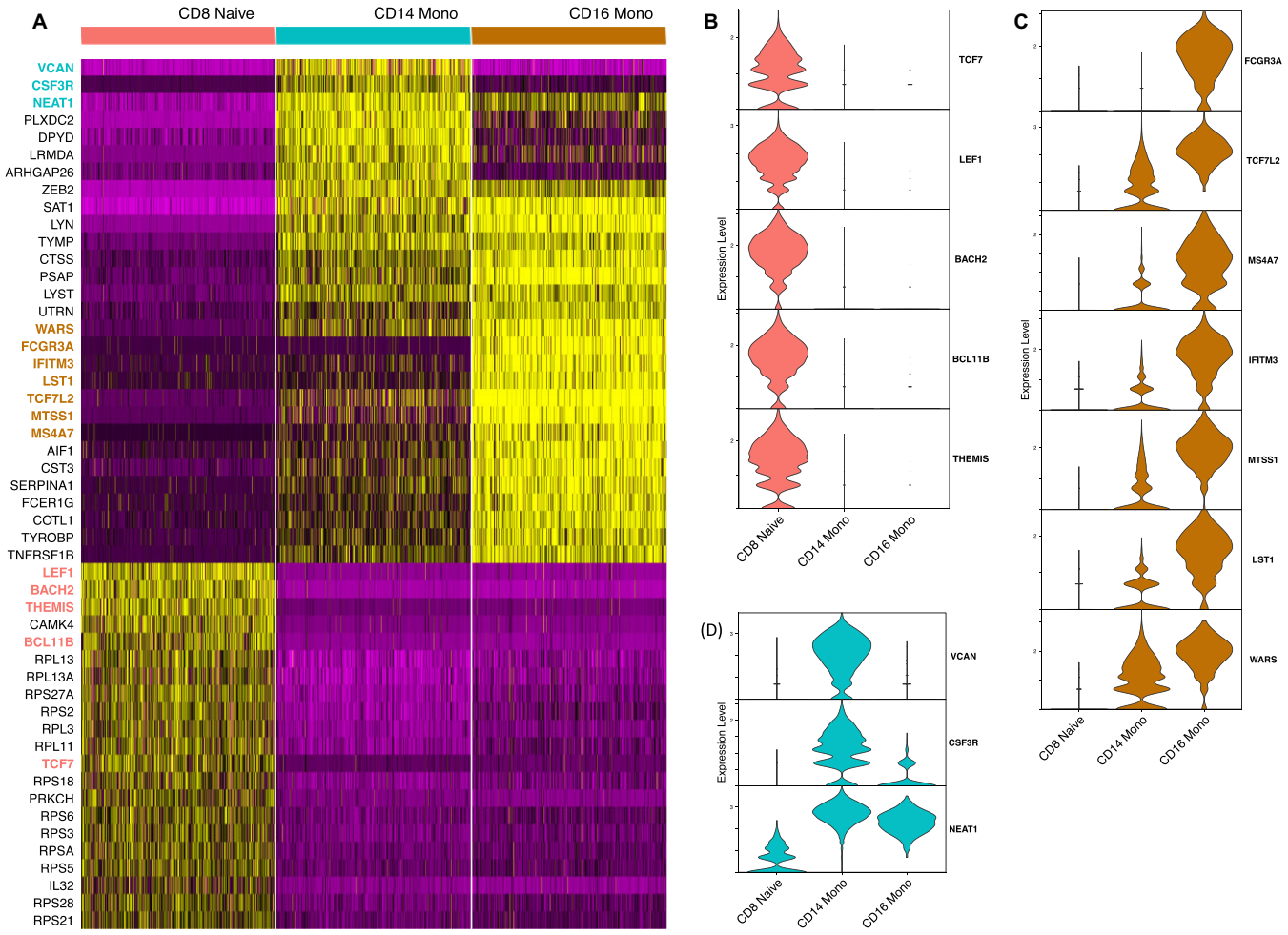

Figure 3: Identification of potential markers for CD8$^+$ naive, CD16$^+$, and CD14$^+$ cell populations in the PBMC dataset. (**A**) Heatmap visualization showing the expression patterns of 50 genes selected by MLme. (**B**–**D**) Expression levels of key markers specific to CD8$^+$ naive, CD16$^+$, and CD14$^+$ cell populations, respectively, within each cell type.

Recognizing the importance of hyperparameter tuning in optimizing models, we plan to incorporate hyperparameter tuning capabilities into the tool. This addition will enable users to fine-tune their models and improve overall performance, thereby increasing the effectiveness and reliability of MLme. Additionally, we intend to introduce a feature that allows users to upload and integrate their own algorithms into the pipeline. This feature will enable users to use their preferred algorithms, even if they are not currently available within the tool, thereby expanding its applicability and customization options.

By drawing inspiration from other similar tools like MLbox [11], TransmogrifAI [9], STREAMLINE [10], AutoSklearn [16], and Weka [6], we aim to integrate advanced features into MLme. These include automated data cleaning, robust feature engineering, and efficient data imputation. These future developments aim to overcome the current limitations of MLme and enhance its functionality and adaptability. By addressing these limitations, we firmly believe that the MLme will evolve into a more comprehensive and valuable resource for ML practitioners.

Availability of Supporting Source Code and Requirements

Project name: Machine Learning Made Easy (MLme)

Project homepage: https://github.com/FunctionalUrology/MLme

Operating system(s): Platform independent

Programming language: Python (version 3.9)

Other requirements: Docker or Python

License: GNU GPL

BioTool ID: MLme

SciCrunch ID: MLme (RRID: SCR_024439)

Additional Files

Supplementary Fig. S1. Key features of Machine Learning Made Easy (MLme). (A) Data Exploration. (B) AutoML.

Supplementary Fig. S2. Key features of Machine Learning Made Easy (MLme). (A) CustomML. (B) Visualization.

Supplementary Fig. S3. Projection of metrics scores on 2-dimensional polar coordinates. The plots illustrate the performance scores of the top and worst 5 machine learning (ML) algorithms trained on the chronic lymphocytic leukemia (CLL) dataset, both during training (A) and testing (B). Each ML model is represented by a circle, and each vertex represents a specific performance metric. A circle with a larger shaded area indicates better performance.

Supplementary Fig. S4. Projection of metrics scores on 2-dimensional polar coordinates. The plots illustrate the performance scores of the top and worst 5 machine learning (ML) algorithms trained on the cervical cancer dataset, both during training (A) and testing (B). Each ML model is represented by a circle, and each vertex represents a specific performance metric. A circle with a larger shaded area indicates better performance.

Supplementary Fig. S5. Projection of metrics scores on 2-dimensional polar coordinates. The plots illustrate the performance scores of the top and worst 5 machine learning (ML) algorithms trained on the TCGA mRNA dataset, both during training (A) and testing (B). Each ML model is represented by a circle, and each vertex represents a specific performance metric. A circle with a larger shaded area indicates better performance.

Supplementary Fig. S6. Projection of metrics scores on 2-dimensional polar coordinates. The plots illustrate the performance scores of the top and worst 5 machine learning (ML) algorithms trained on the TCGA miRNA dataset, both during training (A) and testing (B). Each ML model is represented by a circle, and each vertex represents a specific performance metric. A circle with a larger shaded area indicates better performance.

Supplementary Fig. S7. Projection of metrics scores on 2-dimensional polar coordinates. The plots illustrate the performance scores of the top and worst 5 machine learning (ML) algorithms trained on the peripheral blood mononuclear cell (PBMC) dataset, both during training (A) and testing (B). Each ML model is represented by a circle, and each vertex represents a specific performance metric. A circle with a larger shaded area indicates better performance.

Supplementary Fig. S8. Projection of metrics scores on 2-dimensional polar coordinates. The plots illustrate the performance scores of the top and worst 5 machine learning (ML) algorithms trained on the glass identification dataset, both during training (A) and testing (B). Each ML model is represented by a circle, and each vertex represents a specific performance metric. A circle with a larger shaded area indicates better performance.

Supplementary Fig. S9. Projection of metrics scores on 2-dimensional polar coordinates. The plots illustrate the performance scores of the machine learning (ML) algorithms trained on the body signal dataset, both during training (A) and testing (B). Each ML model is represented by a circle, and each vertex represents a specific performance metric. A circle with a larger shaded area indicates better performance.

Supplementary Fig. S10. Performance comparison of MLme, TPOT, and hyperopt-sklearn across multiple datasets. Each bar in (A), (B), and (C) represents the F1, accuracy, and recall scores, respectively, on the test data for each dataset.

Supplementary Fig. S11. MLme's list of algorithms and diverse plot types for different machine learning stages. MLme offers an array of diverse plots suitable for both exploratory data analysis (EDA) (A) and visualizing outcomes derived from either AutoML or CustomML (B). MLme provides users with the flexibility to design their own machine learning pipelines. (C) The potential pipeline steps alongside corresponding algorithm choices and (D) the steps and corresponding algorithms included in MLme's default AutoML pipeline.

Supplementary Table S1. Comparison of features between MLme and other similar machine learning automation tools.

Supplementary Table S2. Example datasets used in this study.

Data Availability

All supporting data, including the input dataset, "inputParameters.pkl," and "results.pkl" files, for all evaluated datasets, are available on Zenodo [43]. The "results.pkl" files can be visualized using the Visualization feature of MLme. An archival copy of the source code and supporting data is also available via the *GigaScience* database, GigaDB [44]. DOME-ML (Data, Optimisation, Model, and Evaluation in Machine Learning) annotation, supporting the current study, is available through DOME Wizard. The link to the DOME annotations for this study is available on GigaDB [44].

Abbreviations

BRCA: breast carcinoma; CLL: chronic lymphocytic leukemia; miRNA: microRNA; ML: machine learning; MLme: Machine Learning Made Easy; mRNA: messenger RNA; PBMC: peripheral blood mononuclear cell; TCGA: The Cancer Genome Atlas.

Competing Interests

The authors declare they have no competing interests.

Authors' Contributions

K.M., A.H.G, A.A., and M.K. conceived the idea for the manuscript. A.A. and M.K. wrote the source code and conducted testing and debugging of the MLme. K.M., F.C.B., R.M.A., A.H.G., and A.S. provided feedback on the biological application of the tool. N.S., M.A., A.H.G., and A.S. provided technical feedback throughout the development phase and participated in testing and debugging. K.M., A.A., and M.K. wrote the manuscript with inputs from all the other authors. All authors contributed to proofreading and revising the manuscript.

Funding

We gratefully acknowledge the financial support of the Swiss National Science Foundation (SNF Grant 310030_175773 to F.C.B. and K.M., 212298 to F.C.B. and A.H.G.) and the Wings for Life Spinal Cord Research Foundation(WFL-AT-06/19 to K.M.). A.H.G. and R.M.A. are supported by R01 DK127673. M.K. is supported by the Else Kröner-Fresenius-Stiftung (EKFS 2021_EKeA.33). The authors acknowledge the financial support from the Federal Ministry of Education and Research of Germany and by the Sächsische Staatsministerium für Wissenschaft Kultur und Tourismus in the program Center of Excellence for AI-research "Center for Scalable Data Analytics and Artificial Intelligence Dresden/Leipzig" (project identification number: ScaDS.AI).

Acknowledgments

We thank Pedro Perreira Amado for his invaluable contribution in testing MLme.

References

1. Lewis JE, Kemp ML. Integration of machine learning and genome-scale metabolic modeling identifies multi-omics biomarkers for radiation resistance. Nat Commun 2021;12:2700. https://doi.org/10.1038/s41467-021-22989-1.
2. Tollenaar V, Zekollari H, Lhermitte S, et al. Unexplored Antarctic meteorite collection sites revealed through machine learning. Sci Adv 2022;8:eabj8138. https://doi.org/10.1126/sciadv.abj8138.
3. Su Q, Liu Q, Lau RI, et al. Faecal microbiome-based machine learning for multi-class disease diagnosis. Nat Commun 2022;13:6818. https://doi.org/10.1038/s41467-022-34405-3.
4. Martínez BA, Shrotri S, Kingsmore KM, et al. Machine learning reveals distinct gene signature profiles in lesional and nonlesional regions of inflammatory skin diseases. Sci Adv 2022;8:eabn4776. https://doi.org/10.1126/sciadv.abn4776.
5. Chen Z, Ma W, Li Y, et al. Using machine learning to estimate the incidence rate of intimate partner violence. Sci Rep 2023;13:5533. https://doi.org/10.1038/s41598-023-31846-8.
6. Hall M, Frank E, Holmes G, et al. The WEKA data mining software: an update. SIGKDD Explor Newsl 2009;11:10–18. https://doi.org/10.1145/1656274.1656278.
7. Thornton C, Hutter F, Hoos HH, et al. Auto-WEKA: combined selection and hyperparameter optimization of classification algorithms. In: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Association for Computing Machinery; 2013:847–55. https://doi.org/10.1145/2487575.2487629.
8. Frank E, Hall MA, Witten IH. The WEKA Workbench. Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann; 2016.https://www.cs.waikato.ac.nz/ml/weka/book.html
9. Salesforce. Transmogrifai. 2019. https://docs.transmogrif.ai/en/stable/ . Accessed on 20 May 2023.
10. Urbanowicz R, Zhang R, Cui Y, et al. STREAMLINE: a simple, transparent, end-to-end automated machine learning pipeline facilitating data analysis and algorithm comparison. In: Trujillo L, Winkler SM, Silva S, Banzhaf W, eds. Genetic Programming Theory and Practice XIX. Springer Nature, Singapore; 2023:201–31. https://doi.org/10.1007/978-981-19-8460-0_9.
11. Axel. AxeldeRomblay/MLBox. GitHub https://github.com/AxeldeRomblay/MLBox Accessed on 20 May 2023.
12. Jin H, Chollet F, Song Q, et al. AutoKeras: an AutoML library for deep learning. J Mach Learn Res 2023;24:1–6.
13. Komer B, Bergstra J, Eliasmith CH-S. Hyperopt-Sklearn. In: Hutter F, Kotthoff L, Vanschoren J, eds. Automated Machine Learning: Methods, Systems, Challenges. Springer International Publishing, Cham; 2019:97–111. https://doi.org/10.1007/978-3-030-05318-5_5.
14. Le TT, Fu W, Moore JH. Scaling tree-based automated machine learning to biomedical big data with a feature set selector. Bioinformatics 2020;36:250–6. https://doi.org/10.1093/bioinformatics/btz470.
15. La Cava W, Williams H, Fu W, et al. Evaluating recommender systems for AI-driven biomedical informatics. Bioinformatics 2021;37:250–6. https://doi.org/10.1093/bioinformatics/btaa698.
16. Feurer M, Eggensperger K, Falkner S, et al. Auto-sklearn 2.0: hands-free AutoML via meta-learning. J Mach Learn Res 2022;23:261:11936–96.
17. Akshay A. MLme: machine learning made easy. 2023. Accessed on 20 May 2023. https://doi.org/10.48546/WORKFLOWHUB.WORKFLOW.571.1.
18. Dietrich S, Oleś M, Lu J, et al. Drug-perturbation-based stratification of blood cancer. J Clin Invest 2018;128:427–45. https://doi.org/10.1172/JCI93801.
19. Witten D, Tibshirani R, Gu SG, et al. Ultra-high throughput sequencing-based small RNA discovery and discrete statistical biomarker analysis in a collection of cervical tumours and matched controls. BMC Biol 2010;8:58. https://doi.org/10.1186/1741-7007-8-58.
20. Colaprico A, Silva TC, Olsen C, et al. TCGAbiolinks: an R/bioconductor package for integrative analysis of TCGA data. Nucleic Acids Res 2016;44:e71. https://doi.org/10.1093/nar/gkv1507.
21. Robinson MD, Mccarthy DJ, Smyth GK. edgeR: a bioconductor package for differential expression analysis of digital gene expression data. Bioinformatics 2010;26:139–40. https://doi.org/10.1093/bioinformatics/btp616.
22. 10x GenomicsHome page. https://www.10xgenomics.com.Accessed on 20 May 2023.
23. Dua D, Graff C. UCI Machine Learning Repository. 2017.https://archive.ics.uci.edu/
24. Her S. Smoking and drinking dataset with body signal. Kaggle. Accessed on 20 May 2023. https://www.kaggle.com/datasets/sooyoungher/smoking-drinking-dataset
25. Xing S, Li F, Zeng Z, et al. Tcf1 and Lef1 transcription factors establish CD8+ T cell identity through intrinsic HDAC activity. Nat Immunol 2016;17:695–703. https://doi.org/10.1038/ni.3456.
26. Zhang J, Lyu T, Cao Y, et al. Role of TCF-1 in differentiation, exhaustion, and memory of CD8+ T cells: a review. FASEB J 2021;35:e21549.https://doi.org/10.1096/fj.202002566r
27. Roychoudhuri R, Clever D, Li P, et al. BACH2 regulates CD8+ T cell differentiation by controlling access of AP-1 factors to enhancers. Nat Immunol 2016;17:851–60. https://doi.org/10.1038/ni.3441.
28. Helm EY, Zelenka T, Cismasiu VB, et al. Bcl11b sustains multipotency and restricts effector programs of intestinal-resident memory CD8+ T cells. Sci Immunol 2023;8:eabn0484. https://doi.org/10.1126/sciimmunol.abn0484.
29. Tang J, Jia X, Li J, et al. Themis suppresses the effector function of CD8+ T cells in acute viral infection. Cell Mol Immunol 2023;20:512–24. https://doi.org/10.1038/s41423-023-00997-z.
30. Ancuta P, Liu K-Y, Misra V, et al. Transcriptional profiling reveals developmental relationship and distinct biological functions of CD16+ and CD16- monocyte subsets. BMC Genomics 2009;10:403. https://doi.org/10.1186/1471-2164-10-403.
31. Hu Y, Hu Y, Xiao Y, et al. Genetic landscape and autoimmunity of monocytes in developing Vogt–Koyanagi–Harada disease. Proc Natl Acad Sci USA 2020;117:25712–21. https://doi.org/10.1073/pnas.2002476117.
32. Affandi AJ, Olesek K, Grabowska J, et al. CD169 defines activated CD14+ monocytes with enhanced CD8+ T cell activation capacity. Front Immunol 2021;12. https://doi.org/10.3389/fimmu.2021.697840.
33. Combes TW, Orsenigo F, Stewart A, et al. CSF1R defines the mononuclear phagocyte system lineage in human blood in health and COVID-19. Immunother Adv 2021;1:ltab003. https://doi.org/10.1093/immadv/ltab003.
34. Hossain S. Visualization of bioinformatics data with Dash Bio. In: Proceedings of the 18th Python in Science Conference. 2019:126–33. https://doi.org/10.25080/Majora-7ddc1dd1-012.
35. van Rossum G. Python Reference Manual. 1995Department of Computer Science. CWI. ISBN:978-1-4414-1269-0
36. Inc PT. Collaborative data science. 2015. https://plot.ly. Accessed on 20 May 2023.

37. Hunter JD. Matplotlib: a 2D graphics environment. Comput Sci Eng 2007;9:90–5.https://doi.org/10.1109/MCSE.2007.55
38. Bokeh Development Team. Bokeh: Python Library for Interactive Visualization. 2018.https://docs.bokeh.org/
39. McKinney W. Data structures for statistical computing in Python. In: Proceedings of the 9th Python in Science Conference. 2010:56–61. https://doi.org/10.25080/Majora-92bf1922-00a.
40. Harris CR, Millman KJ, Van Der Walt SJ. Array programming with NumPy. Nature 2020;585:357–62. https://doi.org/10.1038/s41586-020-2649-2.
41. Pedregosa F, Varoquaux G, Gramfort A, et al. Scikit-learn: machine learning in Python. J Mach Learn Res 2011;12:2825–30.
42. Lemaıtre G, Nogueira F. Imbalanced-learn: a Python toolbox to tackle the curse of imbalanced datasets in machine learning.2017 Journal of Machine Learning Research 18 17 1–5 https://jmlr.org/papers/v18/16-365.html
43. Akshay A, Katoch M, Shekarchizadeh N, et al. Supporting data for "Machine Learning Made Easy (MLme): A Comprehensive Toolkit for Machine Learning-Driven Data Analysis." Zenodo repository. 2023. https://doi.org/10.5281/zenodo.8073635.
44. Akshay A, Katoch M, Shekarchizadeh N, et al. Supporting data for "Machine Learning Made Easy (MLme): A Comprehensive Toolkit for Machine Learning–Driven Data Analysis." GigaScience Database. 2023. https://doi.org/10.5524/102486.

Received: July 4, 2023. **Revised:** September 20, 2023. **Accepted:** December 8, 2023