The Journal of Open Source Software

Automated Statistical and Machine Learning Platform for Chemical Biology Research

Rimmo Loyi Lego [1,2], **Denver Jn. Baptiste** [1,2], **and Samantha Gauthier**[1,2]

1 Department of Biomedical Engineering, Charles V. Schaefer, Jr. School of Engineering and Science, Stevens Institute of Technology, Hoboken, NJ 07030, USA **2** Department of Chemistry and Chemical Biology, Charles V. Schaefer, Jr. School of Engineering and Science, Stevens Institute of Technology, Hoboken, NJ 07030, USA

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor: Open Journals

Reviewers:

- @openjournals

Submitted: 01 January 1970
Published: unpublished

Summary

This software provides a browser-based platform combining machine learning and statistical analysis for chemical biology research. Researchers can upload CSV data, train Random Forest classification models with automated hyperparameter tuning, and perform statistical tests (t-tests, ANOVA, correlation analysis) through a unified interface requiring no programming or installation. The platform addresses the common workflow challenge of using multiple disconnected tools by integrating data preprocessing, model training, feature importance analysis, and interactive visualization in a single web application. Built with React and TypeScript, it runs entirely in the browser while handling typical research datasets efficiently. This accessibility enables experimental scientists to apply computational methods without extensive technical training.

Statement of Need

Chemical and biomedical researchers routinely need to apply machine learning and statistics to experimental data, but existing tools create significant barriers. Powerful frameworks like scikit-learn (Pedregosa et al., 2011) and R (R Core Team, 2023) require programming expertise that many experimental scientists lack. Tools operate in isolation—researchers must manually transfer data between separate programs for statistical testing, machine learning, and visualization, reducing efficiency and introducing errors (Baker, 2016).

This software addresses these gaps by providing a zero-installation web application that combines Random Forest classification (Breiman, 2001) with standard statistical tests (t-tests, ANOVA, correlation) in one interface. Unlike desktop software or Jupyter notebooks (Kluyver et al., 2016), it requires no installation or coding knowledge. Unlike visual tools like Orange (Demšar et al., 2013), it includes comprehensive statistical testing alongside machine learning. The platform enables complete workflows—upload data, train models, test hypotheses, generate visualizations—without switching applications or writing code.

Key Features

Implementation

Built with React 18.3 and TypeScript, the application uses Vite for fast development and optimized production builds. Core functionality relies on `ml-random-forest` (v2.1) for machine learning, `papaparse` (v5.5) for CSV parsing, and `recharts` (v2.15) for interactive visualizations.

The modular component architecture separates data processing, model training, statistical analysis, and visualization into independent units.

Data Upload and Processing

Users upload CSV files via drag-and-drop or file browser. The system automatically detects file structure, displays preview tables with the first 100 rows, and computes summary statistics (mean, standard deviation, quartiles). Data validation checks for missing values with options for row deletion or imputation. Numerical columns can be normalized using z-score or min-max scaling, while categorical variables are automatically integer-encoded.

Machine Learning

The platform implements Random Forest classification (Breiman, 2001), proven effective for chemical property prediction (Svetnik et al., 2003). Users configure three hyperparameters: number of trees (default: 100), maximum depth (default: unlimited), and minimum samples per split (default: 2). Training runs asynchronously with progress indicators.

Data splits 80/20 for training/testing with stratified sampling to preserve class proportions. The system reports accuracy, precision, recall, and confusion matrices. Feature importance scores identify which variables most influence predictions, helping researchers understand which chemical descriptors drive classification.

Statistical Testing

The platform includes parametric and non-parametric tests for hypothesis testing. Welch's t-test (Welch, 1947) compares means between two groups without assuming equal variances. Mann-Whitney U test provides a non-parametric alternative. For relationships between variables, Pearson's (Pearson, 1895) and Spearman's correlation coefficients quantify linear and monotonic associations respectively. All tests report p-values, effect sizes, and confidence intervals with guidance on assumption checking and test selection.

Visualization

Interactive SVG-based charts include scatter plots, histograms, box plots, feature importance bars, confusion matrices, and correlation heatmaps. All plots support hover tooltips, zoom/pan, and legend toggling. Charts export as PNG for manuscripts and automatically resize for different screen sizes.

User Interface

Tab-based navigation follows the analysis workflow: Data Upload → Model Training → Results → Statistical Analysis. Form validation provides real-time feedback. Models save to browser local storage (5MB) with versioning for comparing multiple configurations. Trained models export as JSON for future predictions.

Research Applications

The platform supports chemical property prediction, bioactivity classification, and exploratory data analysis in chemical biology. Typical applications include QSAR modeling, compound screening, and comparative analysis of experimental conditions. The integrated workflow reduces analysis time and technical barriers for laboratory researchers.

The Journal of Open Source Software

Acknowledgements

The authors acknowledge the Department of Biomedical Engineering and the Department of Chemistry and Chemical Biology at Stevens Institute of Technology for computational resources and institutional support.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., & others. (2016). TensorFlow: A system for large-scale machine learning. *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 265–283.

Baker, M. (2016). 1,500 scientists lift the lid on reproducibility. *Nature*, *533*, 452–454. https://doi.org/10.1038/533452a

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, *13*, 281–305.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297. https://doi.org/10.1007/BF00994018

Darnag, R., Minaoui, B., Glorennec, P. Y., Fakri, A., Zahrae, O., & Mourchid, M. (2010). QSAR studies of HEPT derivatives using support vector machines and neural networks. *QSAR & Combinatorial Science*, *29*(5), 567–577. https://doi.org/10.1002/qsar.200960055

Demšar, J., Curk, T., Erjavec, A., Gorup, Č., Hočevar, T., Milutinovič, M., Možina, M., Polajnar, M., Toplak, M., Starič, A., & others. (2013). Orange: Data mining toolbox in python. *Journal of Machine Learning Research*, *14*, 2349–2353.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, *29*(5), 1189–1232. https://doi.org/10.1214/aos/1013203451

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (2nd ed.). Springer. https://doi.org/10.1007/978-0-387-84858-7

Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B., Bussonnier, M., Frederic, J., Kelley, K., Hamrick, J., Grout, J., Corlay, S., & others. (2016). Jupyter notebooks—a publishing format for reproducible computational workflows. In F. Loizides & B. Schmidt (Eds.), *Positioning and power in academic publishing: Players, agents and agendas* (pp. 87–90). IOS Press. https://doi.org/10.3233/978-1-61499-649-1-87

Mann, H. B., & Whitney, D. R. (1947). On a test of whether one of two random variables is stochastically larger than the other. *Annals of Mathematical Statistics*, *18*(1), 50–60. https://doi.org/10.1214/aoms/1177730491

Murphy, K. P. (2012). *Machine learning: A probabilistic perspective*. MIT Press. ISBN: 978-0262018029

Pearson, K. (1895). Notes on regression and inheritance in the case of two parents. *Proceedings of the Royal Society of London*, *58*, 240–242.

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.

Perkel, J. M. (2021). Ten simple rules for writing and sharing computational analyses in jupyter notebooks. *PLOS Computational Biology*, *17*(7), e1008993. https://doi.org/10.1371/journal.pcbi.1008993

R Core Team. (2023). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. https://www.R-project.org/

Spearman, C. (1904). The proof and measurement of association between two things. *American Journal of Psychology*, *15*(1), 72–101. https://doi.org/10.2307/1412159

Svetnik, V., Liaw, A., Tong, C., Culberson, J. C., Sheridan, R. P., & Feuston, B. P. (2003). Random forest: A classification and regression tool for compound classification and QSAR modeling. *Journal of Chemical Information and Computer Sciences*, *43*(6), 1947–1958. https://doi.org/10.1021/ci034160g

Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, *9*, 2579–2605.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., & others. (2020). SciPy 1.0: Fundamental algorithms for scientific computing in python. *Nature Methods*, *17*, 261–272. https://doi.org/10.1038/s41592-019-0686-2

Welch, B. L. (1947). The generalization of student's problem when several different population variances are involved. *Biometrika*, *34*(1-2), 28–35. https://doi.org/10.1093/biomet/34.1-2.28