

¹ MatrixLM: A Julia package to obtain closed-form least squares estimates for matrix linear models.

³ **Gregory Farage**  ^{1*}, **Jane W Liang**  ^{2*}, **Chenhao Zhao**  ¹, and **Šaunak Sen** 

⁵ 1 Division of Biostatistics, Department of Preventive Medicine, University of Tennessee Health Science Center, United States  ⁶ 2 Division of Research, Kaiser Permanente Northern California, United States  ⁷ * These authors contributed equally.

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

Software

- [Review](#) 
- [Repository](#) 
- [Archive](#) 

Editor:

Submitted: 05 February 2026

Published: unpublished

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).
¹⁶ ¹⁷ ¹⁸ ¹⁹ ²⁰

⁸ Summary

⁹ MatrixLM is an open-source Julia package for fitting matrix linear models, which extend ¹⁰ classical linear regression to a bilinear framework for matrix-valued responses. It is designed ¹¹ for analyzing high-throughput assays in which both rows and columns of the data matrix have ¹² associated covariates, such as in metabolomics, proteomics, or chemical genetic screens.

¹³ In a matrix linear model, the entries of a response matrix are modeled as a joint function ¹⁴ of sample-level covariates (e.g., treatment group, demographic factors) and feature-level ¹⁵ covariates (e.g., molecular or anatomical annotations, biological groups, pathways). MatrixLM ¹⁶ implements efficient estimation and inference for this class of models using efficient matrix ¹⁷ operations whenever possible allowing users to fit large numbers of models while retaining ¹⁸ an explicit linear model interpretation. The inputs include a response matrix and two design ¹⁹ matrices encoding the row and column covariates, and the main outputs include estimated ²⁰ coefficients, standard errors, and test statistics for user-specified contrasts.

²¹ Compared with workflows built from many separate univariate models, MatrixLM provides a ²² unified interface for specifying, fitting, and summarizing matrix linear models. This reduces code ²³ duplication, improves reproducibility, and makes it easier to express hypotheses that naturally ²⁴ involve both sample- and feature-level information (for example, testing for differential effects ²⁵ across feature groups or experimental conditions). By providing an efficient implementation ²⁶ in Julia, MatrixLM enables researchers to perform interpretable analyses of large structured ²⁷ matrix-valued data.

²⁸ Statement of need

²⁹ High-throughput studies in biology and medicine often produce matrix-shaped data where ³⁰ each row corresponds to a sample (e.g., a patient, mutant strain, or experimental unit) and ³¹ each column represents a molecular measurement (e.g., metabolite, gene, or phenotype). In ³² many applications, both the samples and the measured features have associated metadata ³³ that should be incorporated into the analysis. However, existing tools either ignore these ³⁴ annotations or handle them in a fragmented, two-step fashion.

³⁵ Standard approaches often involve fitting a separate model to each feature (e.g., using t-tests ³⁶ or linear models) and then performing a second-stage enrichment or grouping analysis. This ³⁷ approach is limited in two key ways: (1) it does not handle overlapping or quantitative ³⁸ annotations well, and (2) it fails to exploit shared structure among features or among samples. ³⁹ Dimension-reduction and machine learning methods address some of these issues, but often ⁴⁰ sacrifice interpretability and do not provide familiar statistical outputs like effect sizes or

⁴¹ confidence intervals.

⁴² MatrixLM addresses this gap by implementing matrix linear models (MLMs), a class of
⁴³ bilinear models that allow researchers to directly model associations between sample-level and
⁴⁴ feature-level characteristics. MLMs naturally accommodate both categorical and continuous
⁴⁵ annotations, support hypothesis testing, and enable users to assess the effect of covariates
⁴⁶ while adjusting for confounding structure. Compared to standard univariate workflows, MLMs
⁴⁷ offer better interpretability and power, especially when annotations overlap or when feature
⁴⁸ relationships are complex, as demonstrated in both chemical genetic screens (Liang et al.,
⁴⁹ 2019) and metabolomics applications (Farage et al., 2025).

⁵⁰ Despite their utility, matrix linear models have not been widely available in reusable, general-
⁵¹ purpose software. MatrixLM provides a fast, open-source Julia implementation with a user-
⁵²friendly formula interface, making it easier for applied researchers to fit, interpret, and extend
⁵³ these models in large-scale studies.

Key Features

⁵⁴ MatrixLM leverages the speed and expressiveness of the Julia programming language to provide:

- ⁵⁵ **Closed-form Estimation:** Uses efficient matrix operations to solve least squares problems without the need for iterative solvers or massive Kronecker products in memory.
- ⁵⁶ **Formula Interface:** Includes a @formula macro that allows users to specify models for both row (X) and column (Z) covariates using standard statistical model formula syntax (e.g., \sim Treatment + Age).
- ⁵⁷ **Statistical Inference:** Provides standard errors, t-statistics, and p-values for estimated coefficients.
- ⁵⁸ **Permutation Testing:** Built-in support for permutation tests to control false positive rates making minimal assumptions about the error distribution.

Mathematical framework

⁶⁵ Matrix linear models extend ordinary linear regression to situations where the outcome is a whole matrix rather than a single response vector. We arrange the data as follows:

- ⁶⁶ Y is an $n \times m$ matrix of high-throughput measurements (rows = samples, columns = features).
- ⁶⁷ X is an $n \times p$ matrix of sample-level covariates (e.g., treatment group, sex, clinical variables).
- ⁶⁸ Z is an $m \times q$ matrix of feature-level covariates (e.g., metabolite class, pathway, or other annotations).
- ⁶⁹ B is a $p \times q$ matrix of regression coefficients linking the sample and feature covariates.

⁷⁰ The matrix linear model assumes

$$Y = XBZ^T + E,$$

⁷⁵ where E is an $n \times m$ matrix of residuals. In element-wise form, each entry y_{ij} is written as

$$y_{ij} = \sum_{k=1}^p \sum_{\ell=1}^q x_{ik} z_{j\ell} b_{k\ell} + e_{ij}.$$

⁷⁷ As the equation above shows, the elements of B may be interpreted as interactions between the columns of X and the columns of Z .

⁷⁹ We treat X and Z as known and estimate B by least squares, choosing \hat{B} to minimize the Frobenius norm of the residuals,

$$\hat{B} = \arg \min_B \|Y - XBZ^T\|_F^2.$$

81 This optimization problem has a closed-form solution:

$$\hat{B} = (X^T X)^{-1} X^T Y Z (Z^T Z)^{-1},$$

82 when $X^T X$ and $Z^T Z$ are invertible. From \hat{B} , the software can construct fitted values
83 $\hat{Y} = X \hat{B} Z^T$ and standard errors and test statistics for entries of \hat{B} or for user-specified linear
84 contrasts, directly analogous to classical linear models.

85 When there is only a single feature (one column of Y) and no feature-level design matrix Z ,
86 this framework reduces to ordinary linear regression with design matrix X . The MatrixLM
87 package therefore generalizes familiar linear modeling ideas to matrix-valued outcomes with
88 structured annotations on both rows and columns.

89 Figures

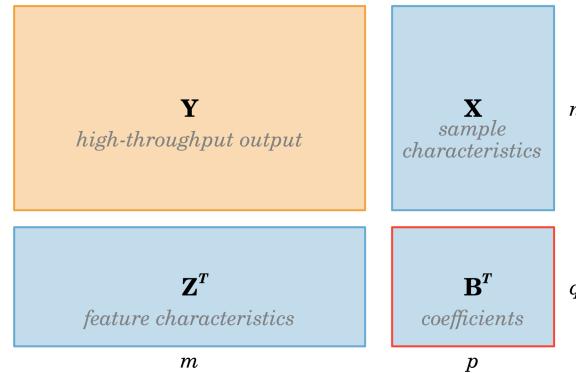


Figure 1: A visualization of the response ($\mathbf{Y} : n \times m$), sample covariates ($\mathbf{X} : n \times p$), feature covariates ($\mathbf{Z}^T : m \times q$), and coefficients ($\mathbf{B}^T : p \times q$) matrices for a matrix linear model. The dimensions in the model correspond to n samples/individuals, m features/measurements, p sample covariates, and q feature covariates; the matrix \mathbf{B} is to be estimated.

90 Conclusion and future directions

91 MatrixLM provides a practical implementation of matrix linear models for encoding relationships
92 and groupings high-throughput, matrix-shaped data with annotations on both samples and
93 features. By combining a flexible formula interface with fast, closed-form least-squares
94 estimation, the package makes it straightforward for applied researchers to encode biological
95 or experimental structure directly into their models.

96 In ongoing work, we are extending this framework to penalized matrix linear models for high-
97 dimensional settings: a companion Julia package, MatrixLMnet, that implements elastic-net
98 and related penalties on the coefficient matrix to enable variable selection and regularization
99 in matrix linear models.

100 AI usage disclosure

101 The authors used GitHub Copilot to assist in coding testing functions. All Copilot-suggested
102 code was reviewed, tested, and validated by the human authors to ensure correctness. The
103 authors take full responsibility for the content of this manuscript.

104 Acknowledgements

105 This work started when JWL was a summer intern at UCSF, and continued when she was a
106 scientific programmer at the University of Tennessee Health Science Center (UTHSC). We
107 thank both UCSF and UTHSC for funding, and a supportive environment. This work was
108 partly supported by National Institutes of Health grants GM-070683 (SS), GM-078338 (SS),
109 GM-123489 (SS, GF), ES-022841 (SS), and DA-044223 (SS).

110 References

- 111 Farage, G., Zhao, C., Choi, H. Y., Garrett, T. J., Elam, M. B., Kechris, K., & Sen, Š. (2025).
112 Matrix linear models for connecting metabolite composition to individual characteristics.
113 *Metabolites*, 15(2). <https://doi.org/10.3390/metabo15020140>
- 114 Liang, J. W., Nichols, R. J., & Sen, Š. (2019). Matrix Linear Models for High-Throughput
115 Chemical Genetic Screens. *Genetics*, 4(212), 1063–1073. <https://doi.org/10.1534/genetics.119.302299>