The Journal of Open Source Software

TMLE.jl: Targeted Minimum Loss-Based Estimation In Julia.

Olivier Labayle[1], **Chris P. Ponting**[2], **Mark J. van der Laan**[5], **Ava Khamseh**[2,3,5], **and Sjoerd Viktor Beentjes**[2,4,5]

1 Institute for Regeneration and Repair, University of Edinburgh, Edinburgh EH16 4UU, United Kingdom **2** MRC Human Genetics Unit, Institute of Genetics and Cancer, University of Edinburgh, Edinburgh EH4 2XU, United Kingdom. **3** School of Informatics, University of Edinburgh, Edinburgh EH8 9AB, United Kingdom **4** School of Mathematics and Maxwell Institute for Mathematical Sciences, University of Edinburgh, Edinburgh EH9 3FD, United Kingdom **5** Division of Biostatistics, University of California, Berkeley, CA, USA

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor: Open Journals

Reviewers:

- @openjournals

Submitted: 01 January 1970
Published: unpublished

Summary

TMLE.jl is a Julia (Bezanson et al., 2017) package implementing targeted minimum loss-based estimation (TMLE), a general framework for causal effect estimation that unites modern machine learning with the theoretical guarantees of semiparametric statistics. TMLE yields doubly robust and semiparametrically efficient estimators, meaning it remains consistent if either the outcome model or the treatment assignment model is correctly specified, and it achieves the smallest possible asymptotic variance under standard regularity conditions. The package integrates with the broader Julia machine learning ecosystem and can be used in both observational and experimental settings. It is particularly well-suited for high-dimensional problems where robust inference is essential.

Background

Causal inference is essential for understanding the effects of interventions in real-world settings, such as determining whether a treatment improves health outcomes or whether a genetic variant contributes to disease risk. Traditional approaches often begin by positing a specific parametric model, such as a linear-Gaussian or logistic regression model, and then estimating its parameters using efficient likelihood-based methods. This strategy has two main drawbacks. First, the quantity being estimated is often dictated by the parametric form of the model rather than the scientific question of interest (e.g., additive effects under a linear model, odds ratios under a logistic model). Second, when the model is misspecified, particularly in high-dimensional settings with complex interactions, parametric estimators can be severely biased.

Over the past two decades, new approaches have emerged that combine causal inference, machine learning (ML), and semiparametric theory to address these limitations. These methods begin with the explicit definition of a target parameter, often a causal estimand such as the average treatment effect, derived from a causal model. They then replace restrictive parametric models with flexible ML algorithms (e.g., gradient boosting, neural networks) to estimate nuisance components such as the outcome regression and treatment mechanism (called nuisance functions). However, since most ML methods are optimised for prediction rather than unbiased estimation, ML-based estimates are still biased when plugged directly into the parameter formula. To remove this bias, modern approaches use a debiasing step based on a key mathematical object, the efficient influence function (EIF). The result is an estimator that is asymptotically unbiased, efficient, and valid under much weaker assumptions than traditional parametric models.

Several estimators share this debiasing principle. The one-step estimator (OSE) (Kennedy, 2024; Pfanzagl & Wefelmeyer, 1985) is a general methodology which proceeds by estimating and removing the bias resulting from machine-learning fitting via the EIF. The widely used augmented inverse probability of treatment weighting (Glynn & Quinn, 2010), is a special case of the OSE for the average treatment effect. The double machine learning (DML) framework (Chernozhukov et al., 2018) extends the OSE by introducing cross-fitting, which allows the use of highly flexible ML algorithms without restrictive empirical process conditions (e.g., Donsker assumptions). A limitation of both OSE and DML is that they perform debiasing in the parameter space, which can lead to estimates outside the natural range of the target causal estimand (e.g., probabilities below 0 or above 1). In contrast, the targeted maximum likelihood estimator (TMLE) (Van der Laan et al., 2011; Van der Laan & Rose, 2018) performs the debiasing step in function space by iteratively updating the initial ML fit so that the resulting estimate both removes bias and respects the parameter's natural bounds. While more involved to implement, TMLE retains all the theoretical guarantees of OSE/DML while ensuring parameter validity.

TMLE.jl is a Julia package that implements both TMLE and the OSE, with or without cross-fitting, for the estimation of causal parameters in observational and experimental studies. It enables researchers to estimate average treatment effects and other causal parameters while leveraging modern ML algorithms. TMLE.jl is applicable across a broad range of disciplines—including epidemiology, biostatistics, econometrics, and genomics—where valid causal effect estimation from high-dimensional or observational data is essential.

Statement of Need

The main entry point to the DML methodology, in both Python and R, is the DoubleML package (Bach et al., 2022; Bach, Kurz, et al., 2024; Bach, Chernozhukov, et al., 2024). Further Python packages focused on the estimation of the conditional average treatment effect include EconML (Keith Battocchi, 2019) and CausalML (Chen et al., 2020).

Despite its theoretical and practical advantages, targeted maximum likelihood estimation (TMLE) remains largely implemented in R, with limited support in other languages. The original tmle package (Gruber & van der Laan, 2012) provides a single estimator for various causal parameters and relies on the SuperLearner package for flexible nuisance estimation. The more recent tmle3 package (Coyle, 2021) offers a unified, object-oriented interface that explicitly represents the key mathematical components of TMLE. It supports a broad range of parameters, integrates cross-validated TMLE (CV-TMLE), the TMLE analogue of cross-fitting in DML, and is part of the broader tlverse ecosystem. Specialised extensions in this ecosystem target particular estimands, such as the mean outcome under the optimal treatment rule (tmle3mopttx) (Malenica et al., 2022) and stochastic intervention parameters (tmle3shift) (Hejazi et al., 2021). Longitudinal TMLE for time-varying exposures is supported by the separate ltmle package (Lendle et al., 2017). To improve the robustness of estimators, for instance in the presence of practical violations or model instability, the collaborative targeted maximum likelihood estimation (C-TMLE) framework has been proposed, with several implementations available via the ctmle package (Ju et al., 2017).

For practitioners and developers who prefer a performant, composable, and type-safe environment such as Julia, no native TMLE implementation previously existed. As causal inference methods gain traction in computational biology, health sciences, economics, and other data-intensive disciplines, the absence of robust, well-integrated TMLE tooling in modern scientific programming languages has become increasingly limiting. TMLE.jl addresses this gap by providing the first native Julia implementation of TMLE. It supports the estimation of a variety of causal estimands, including the counterfactual mean, average treatment effect, and average interaction effects of arbitrary order. Any differentiable transformation of these estimands (e.g., risk ratio, odds ratio) can be obtained via automatic differentiation. Both TMLE and one-step estimators (OSE) are available in canonical and cross-fitting variants through a unified interface, and selected C-TMLE instantiations (greedy and scalable) are also implemented. The package accommodates more than binary treatments, allowing for any number of categorical treatments, an important feature for studying combinatorial intervention effects.

TMLE.jl is fully integrated into the broader Julia ecosystem. Machine learning models, including ensemble learners, can be specified via the MLJ toolbox (Blaom et al., 2020); datasets are represented as DataFrames (Bouchet-Valat & Kamiński, 2023); and automatic differentiation is supported through any backend via DifferentiationInterface.jl (Dalle & Hill, 2024). Simulation datasets from CausalTables.jl (Balkus & Hejazi, 2025) are also directly supported.

In doing so, TMLE.jl fills an important niche for causal inference in Julia, expanding the reach of TMLE beyond R and contributing to a growing ecosystem of open-source tools for rigorous, scalable, and reproducible statistical modelling.

Applications to Population Genetics

While TMLE.jl is applicable to a broad range of scientific problems, it was developed with population genetics in mind, which informed several aspects of its design. Its performance has been benchmarked in large-scale population genetics simulations and applied to UK Biobank data (Labayle et al., 2025). In such settings, the number of estimands can reach millions, posing a challenge for semiparametric estimators that rely on computationally intensive machine learning procedures. To address this, TMLE.jl implements automatic caching of intermediate results, enabling substantial computational savings. For example, when estimating the effect of a single treatment variable across multiple traits, the same propensity score can be reused across all analyses without recomputation. This mechanism has already been applied to the discovery of genetic variants affecting human traits via differential binding (submitted), and is currently being used in studies of genetic variants associated with myalgic encephalomyelitis.

Acknowledgements

Olivier Labayle was supported by the United Kingdom Research and Innovation (grant EP/S02431X/1), UKRI Centre for Doctoral Training in Biomedical AI at the University of Edinburgh, School of Informatics. Mark van der Laan is supported by NIH grant R01AI074345. Chris P. Ponting was funded by the MRC (MC_UU_00007/15). Ava Khamseh was supported by the XDF Programme from the University of Edinburgh and Medical Research Council (MC_UU_00009/2), and by a Langmuir Talent Development Fellowship from the Institute of Genetics and Cancer, and a philanthropic donation from Hugh and Josseline Langmuir.

References

Bach, P., Chernozhukov, V., Klaassen, S., Kurz, M. S., & Spindler, M. (2024). *DoubleML – Double machine learning in Python* (latest). https://github.com/DoubleML/doubleml-for-py

Bach, P., Chernozhukov, V., Kurz, M. S., & Spindler, M. (2022). DoubleML – An object-oriented implementation of double machine learning in Python. *Journal of Machine Learning Research*, *23*(53), 1–6. http://jmlr.org/papers/v23/21-0862.html

Bach, P., Kurz, M. S., Chernozhukov, V., Spindler, M., & Klaassen, S. (2024). DoubleML: An object-oriented implementation of double machine learning in R. *Journal of Statistical Software*, *108*(3), 1–56. https://doi.org/10.18637/jss.v108.i03

Balkus, S. V., & Hejazi, N. S. (2025). CausalTables.jl: Simulating and storing data for statistical causal inference in Julia. *Journal of Open Source Software*, *10*(106), 7580. https://doi.org/10.21105/joss.07580

Bezanson, J., Edelman, A., Karpinski, S., & Shah, V. B. (2017). Julia: A fresh approach to numerical computing. *SIAM Review*, *59*(1), 65–98. https://doi.org/10.1137/141000671

Blaom, A. D., Kiraly, F., Lienart, T., Simillides, Y., Arenas, D., & Vollmer, S. J. (2020). MLJ: A Julia package for composable machine learning. *arXiv Preprint arXiv:2007.12285*. https://doi.org/10.21105/joss.02704

Bouchet-Valat, M., & Kamiński, B. (2023). Dataframes.jl: Flexible and fast tabular data in Julia. *Journal of Statistical Software*, *107*, 1–32. https://doi.org/10.18637/jss.v107.i04

Chen, H., Harinen, T., Lee, J.-Y., Yung, M., & Zhao, Z. (2020). *CausalML: Python package for causal machine learning*. https://arxiv.org/abs/2002.11631

Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W., & Robins, J. (2018). *Double/debiased machine learning for treatment and structural parameters*. Oxford University Press Oxford, UK. https://doi.org/10.1111/ectj.12097

Coyle, J. R. (2021). *tmle3: The extensible TMLE framework*. https://github.com/tlverse/tmle3. https://doi.org/10.5281/zenodo.4603358

Dalle, G., & Hill, A. (2024). *DifferentiationInterface.jl*. Zenodo. https://doi.org/10.5281/zenodo.11092033

Glynn, A. N., & Quinn, K. M. (2010). An introduction to the augmented inverse propensity weighted estimator. *Political Analysis*, *18*(1), 36–56. https://doi.org/doi:10.1093/pan/mpp036

Gruber, S., & van der Laan, M. J. (2012). tmle: An R package for targeted maximum likelihood estimation. *Journal of Statistical Software*, *51*(13), 1–35. https://doi.org/10.18637/jss.v051.i13

Hejazi, N. S., Coyle, J. R., & van der Laan, M. J. (2021). *tmle3shift: Targeted Learning of the causal effects of stochastic interventions*. https://github.com/tlverse/tmle3shift. https://doi.org/10.5281/zenodo.4603372

Ju, C., Gruber, S., & Laan, M. van der. (2017). *Ctmle: Collaborative targeted maximum likelihood estimation*. https://doi.org/10.32614/cran.package.ctmle

Keith Battocchi, M. H., Eleanor Dillon. (2019). *EconML: A Python Package for ML-Based Heterogeneous Treatment Effects Estimation*. https://github.com/py-why/EconML.

Kennedy, E. H. (2024). Semiparametric doubly robust targeted double machine learning: A review. *Handbook of Statistical Methods for Precision Medicine*, 207–236. https://doi.org/10.48550/arXiv.2203.06469

Labayle, O., Roskams-Hieter, B., Slaughter, J., Tetley-Campbell, K., Laan, M. J. van der, Ponting, C. P., Beentjes, S. V., & Khamseh, A. (2025). Semi-parametric efficient estimation of small genetic effects in large-scale population cohorts. *arXiv Preprint arXiv:2505.14675*. https://doi.org/10.48550/arXiv.2505.14675

Lendle, S. D., Schwab, J., Petersen, M. L., & van der Laan, M. J. (2017). ltmle: An R package implementing targeted minimum loss-based estimation for longitudinal data. *Journal of Statistical Software*, *81*(1), 1–21. https://doi.org/10.18637/jss.v081.i01

Malenica, I., Coyle, J., & van der Laan, M. J. (2022). *tmle3mopttx: Targeted learning and variable importance with optimal individualized categorical treatment*. https://github.com/tlverse/tmle3mopttx

Pfanzagl, J., & Wefelmeyer, W. (1985). Contributions to a general asymptotic statistical theory.

Statistics & Risk Modeling, *3*(3-4), 379–388. https://doi.org/10.1007/978-1-4612-5769-1

Van der Laan, M. J., & Rose, S. (2018). *Targeted learning in data science*. Springer. https://doi.org/10.1007/978-3-319-65304-4

Van der Laan, M. J., Rose, S., & others. (2011). *Targeted learning: Causal inference for observational and experimental data* (Vol. 4). Springer. https://doi.org/10.1007/978-1-4419-9782-1