The Journal of Open Source Software

PyMC-Marketing: Bayesian Marketing Mix Models and Customer Analytics in Python

William Dean[1], **Juan Orduz**[1], **Colt Allen**[1], **Carlos Trujillo**[1], **Ricardo Vieira**[1], **Benjamin T. Vincent**[1], **Thomas V. Wiecki**[1], and **Nathaniel Forde**[1]

1 PyMC Labs

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor: Open Journals

Reviewers:

- @openjournals

Submitted: 01 January 1970
Published: unpublished

Summary

PyMC-Marketing is a comprehensive Python library implementing Bayesian marketing analytics, built on PyMC (Salvatier et al., 2016). Commercial marketing analytics tools typically provide limited transparency into their models, while open-source alternatives like Meta's Robyn and Google's Meridian focus primarily on media mix modeling (Google Inc., 2023; Meta Platforms Inc., 2022). PyMC-Marketing provides a unified framework spanning multiple marketing domains, including: Media Mix Modeling, Customer Lifetime Value analysis, Bass Diffusion Models, and Customer Choice Models. All outputs include full posterior distributions rather than point estimates, enabling explicit risk assessment in business decisions.

Statement of Need

Marketing organizations struggle to attribute sales outcomes to specific marketing activities across multiple touchpoints and delayed conversion effects. Existing solutions suffer from: (1) black-box proprietary models with limited customization; (2) oversimplified approaches failing to capture marketing dynamics; and (3) lack of uncertainty quantification for high-stakes decisions.

PyMC-Marketing addresses these gaps by bridging marketing science research and practical applications. It operationalizes advanced Bayesian methods—hierarchical modeling, experimental calibration, and uncertainty quantification—within a user-friendly, scikit-learn compatible API. Key innovations include time-varying coefficients using modern Gaussian process approximations optimized for marketing contexts, and a novel experimental calibration framework that integrates lift test results directly into model likelihood. While frequentist approaches like Robyn provide bootstrap-based intervals, all PyMC-Marketing outputs include full Bayesian posterior distributions, enabling decision-makers to assess risk explicitly.

State of the Field

Existing marketing mix modeling tools include Meta's Robyn (Meta Platforms Inc., 2022) and Google's Meridian (Google Inc., 2023), which focus primarily on MMM with limited Bayesian inference capabilities. While Robyn provides bootstrap-based uncertainty intervals, it lacks full posterior distributions. Meridian offers Bayesian inference but is limited to media mix modeling without extending to customer lifetime value, choice analysis, or product diffusion modeling.

PyMC-Marketing fills this gap by providing a unified Bayesian framework across multiple marketing domains (MMM, CLV, Bass diffusion, choice modeling) with full uncertainty quantification. Rather than contributing to existing tools, we created a standalone library to integrate advanced

Bayesian methods (hierarchical modeling, experimental calibration, time-varying parameters via modern GP approximations) within a scikit-learn compatible API. This design enables both methodological research and production applications while maintaining computational efficiency.

Software Design

PyMC-Marketing follows a modular component architecture built on the PyMC probabilistic programming framework (Salvatier et al., 2016). The design prioritizes flexibility and extensibility through pluggable transformation components (adstock, saturation functions) and a builder pattern for model construction.

Key architectural decisions include: (1) separation of data transformation from model specification enabling custom function implementation; (2) scikit-learn compatibility for seamless integration with existing ML pipelines; (3) PyMC backend providing automatic differentiation and multiple MCMC samplers; (4) standardized serialization for production deployment via MLflow (Zaharia et al., 2018). This architecture enables both methodological research and production applications while maintaining computational efficiency through GPU acceleration and modern sampling algorithms including NumPyro (Bingham et al., 2019) and Nutpie (Seyboldt & PyMC Developers, n.d.).

Installation and Dependencies

PyMC-Marketing is available on conda-forge and PyPI. Core dependencies include PyMC (5.0 or higher), NumPy (Harris et al., 2020), Pandas (team, 2020), ArviZ (Kumar et al., 2019), and scikit-learn (Pedregosa et al., 2011). Optional dependencies enable GPU acceleration (JAX), advanced samplers (NumPyro, Nutpie), and production deployment (MLflow (Zaharia et al., 2018), Docker).

Key Features

PyMC-Marketing provides multiple comprehensive modules addressing various marketing analytics domains:

1. Media Mix Modeling (MMM): Multiple adstock functions, saturation curves, time-varying parameters via HSGP (Solin & Särkkä, 2020), experimental calibration for causal inference, budget optimization with business constraints, time-slice cross-validation, and marginal effects analysis (Arel-Bundock et al., 2024).

2. Customer Lifetime Value (CLV): BTYD models (Fader & Hardie, 2020) including BG/NBD, Pareto/NBD, and Gamma-Gamma frameworks with hierarchical extensions and individual-level uncertainty.

3. Bass Diffusion Models: Product adoption forecasting (Bass, 1969) with flexible parameterization for innovation and imitation effects across multiple products.

4. Customer Choice Models: Discrete choice analysis (Train, 2009) based on random utility theory, including multinomial logit and multivariate interrupted time series models.

Production Ready: All modules feature MLflow (Zaharia et al., 2018) integration, Docker containerization, multiple MCMC backends (NumPyro (Bingham et al., 2019), Nutpie (Seyboldt & PyMC Developers, n.d.)), variational inference (ADVI), MAP estimation, data connectors (e.g., Fivetran), and comprehensive diagnostics via ArviZ.

The Journal of Open Source Software

Research Impact Statement

PyMC-Marketing provides uncertainty quantification through full posterior distributions, experimental calibration anchoring observational models to causal ground truth, and flexible budget optimization with business constraints. The scikit-learn compatible API ensures seamless integration into existing data science workflows. The library has been successfully deployed by companies including HelloFresh and Bolt for production marketing analytics, demonstrating real-world impact and scalability.

Novel methodological contributions include: (1) time-varying coefficients using modern Gaussian process approximations specifically optimized for marketing applications; (2) experimental calibration framework integrating lift test results directly into model likelihood—a novel approach in the marketing science literature; (3) comprehensive marginal effects analysis for marketing sensitivity studies (Arel-Bundock et al., 2024). Comprehensive tutorials, example notebooks, and video resources are available in the online documentation at https://www.pymc-marketing.io/en/stable/, with community support from over 70 contributors and translations in Spanish.

AI Usage Disclosure

Generative AI tools were used during paper preparation. OpenCode (v1.0.220) with Claude Opus 4.5 assisted with gathering information from existing documentation and codebase, drafting text, and incorporating peer reviewer feedback. The PyMC-Marketing software itself was developed by human contributors. All paper content was reviewed, edited, and validated by the human authors.

Funding

PyMC-Marketing is a community-driven project built primarily through volunteer contributions from over 70 developers. Some contributions by PyMC Labs affiliates have received partial funding from both consulting engagements on marketing analytics and its internal budget.

Acknowledgments

We acknowledge the PyMC development team for the foundational probabilistic programming framework and the broader PyData ecosystem contributors. Special recognition goes to the marketing science research community for developing the theoretical frameworks that PyMC-Marketing operationalizes (Bass, 1969; Fader et al., 2005; Jin et al., 2017; Train, 2009).

References

Arel-Bundock, V., Greifer, N., & Heiss, A. (2024). How to interpret statistical models using marginaleffects for R and Python. *Journal of Statistical Software*, *111*(9), 1–32. https://doi.org/10.18637/jss.v111.i09

Bass, F. M. (1969). A new product growth model for consumer durables. *Management Science*, *15*(5), 215–227. https://doi.org/10.1287/mnsc.15.5.215

Bingham, E., Chen, J. P., Jankowiak, M., Obermeyer, F., Pradhan, N., Karaletsos, T., Singh, R., Szerlip, P. A., Horsfall, P., & Goodman, N. D. (2019). Pyro: Deep universal probabilistic programming. *Journal of Machine Learning Research*, *20*(28), 1–6. http://jmlr.org/papers/v20/18-403.html

Fader, P. S., & Hardie, B. G. (2020). Customer lifetime value: What's next? *Journal of Interactive Marketing*, *49*, 26–37. https://doi.org/10.1016/j.intmar.2019.08.001

Fader, P. S., Hardie, B. G., & Lee, K. L. (2005). Counting your customers: Who are they and what will they do next? *Management Science*, *51*(9), 1371–1387. https://doi.org/10.1287/mnsc.51.9.1371

Google Inc. (2023). *Meridian: Bayesian media mix modeling at scale*. https://github.com/google/meridian. https://github.com/google/meridian

Harris, C. R., Millman, K. J., Walt, S. J. van der, Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., & others. (2020). Array programming with NumPy. *Nature*, *585*(7825), 357–362. https://doi.org/10.1038/s41586-020-2649-2

Jin, Y., Wang, Y., Sun, Y., Chan, D., & Koehler, J. (2017). *Bayesian methods for media mix modeling with carryover and shape effects*. Google Inc. https://research.google/pubs/pub46001/

Kumar, R., Carroll, C., Hartikainen, A., & Martin, O. A. (2019). ArviZ: A unified library for exploratory analysis of bayesian models in python. *Journal of Open Source Software*, *4*(33), 1143. https://doi.org/10.21105/joss.01143

Meta Platforms Inc. (2022). *Robyn: Marketing mix modeling*. https://github.com/facebookexperimental/Robyn. https://github.com/facebookexperimental/Robyn

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., & others. (2011). Scikit-learn: Machine learning in python. *The Journal of Machine Learning Research*, *12*, 2825–2830.

Salvatier, J., Wiecki, T. V., & Fonnesbeck, C. (2016). Probabilistic programming in python using PyMC3. *PeerJ Computer Science*, *2*, e55. https://doi.org/10.7717/peerj-cs.55

Seyboldt, A., & PyMC Developers. (n.d.). *nutpie*. https://github.com/pymc-devs/nutpie

Solin, A., & Särkkä, S. (2020). Hilbert space methods for reduced-rank gaussian process regression. *Statistics and Computing*, *30*(2), 419–446. https://doi.org/10.1007/s11222-019-09886-w

team, T. pandas development. (2020). *Pandas-dev/pandas: pandas* (latest). Zenodo. https://doi.org/10.5281/zenodo.3509134

Train, K. E. (2009). *Discrete choice methods with simulation*. Cambridge University Press. https://doi.org/10.1017/CBO9780511805271

Zaharia, M. A., Chen, A., Davidson, A., Ghodsi, A., Hong, S. A., Konwinski, A., Murching, S., Nykodym, T., Ogilvie, P., Parkhe, M., Xie, F., & Zumar, C. (2018). Accelerating the Machine Learning Lifecycle with MLflow. *IEEE Data Eng. Bull.*, *41*, 39–45. https://api.semanticscholar.org/CorpusID:83459546