Additive Bayesian Networks

Matteo Delucchi [1,2], **Jonas I. Liechti** [3], **Georg R. Spinner** [2], and **Reinhard Furrer** [1¶]

1 Department of Mathematical Modeling and Machine Learning, University of Zurich, Zürich, Switzerland **2** Centre for Computational Health, Institute of Computational Life Sciences, Zurich University of Applied Sciences (ZHAW), Wädenswil, Switzerland **3** www.T4D.ch, T4D GmbH, Zurich, Switzerland ¶ Corresponding author

DOI: 10.21105/joss.06822

Software

- Review
- Repository
- Archive

Editor: Chris Vernon

Reviewers:

- @dhvalden
- @abhishektiwari

Submitted: 20 May 2024
Published: 30 September 2024

Summary

The R package abn is a comprehensive tool for Bayesian Network (BN) analysis, a form of probabilistic graphical model. BNs are a type of statistical model that leverages the principles of Bayesian statistics and graph theory to provide a framework for representing complex multivariate data. They can derive a directed acyclic graph from empirical data to describe the dependency structure between random variables.

Additive Bayesian Network (ABN) models extend the concept of generalized linear models, typically used for predicting a single outcome, to scenarios with multiple dependent variables (e.g., Kratzer et al. (2023)). This makes them a powerful tool for understanding complex, multivariate datasets. This package provides routines for structure learning and parameter estimation of ABN models.

Statement of need

The increasing complexity of data in various fields, ranging from healthcare research to environmental science and ecology, has resulted in a need for a tool like abn. Researchers often face multivariate, tabular data where the relationships between variables are not straightforward. BN analysis becomes essential when traditional statistical methods fail to analyze multivariate data with intricate relationships, as it models these relationships graphically for more straightforward data interpretation.

Commonly used implementations of BN models, such as `bnlearn` (Scutari, 2010), `bnstruct` (Franzin et al., 2017), `deal` (Boettcher & Dethlefsen, 2003), `gRain` (Højsgaard, 2012), `pcalg` (Kalisch et al., 2012) and `pchc` (Tsagris, 2021), limit variable types, often allowing discrete variables to have only discrete parent variables, where a parent starts a directed edge in the graph. This limitation can pose challenges when dealing with continuous or mixed-type data (i.e., data that includes both continuous and discrete variables) or when attempting to model complex relationships that do not fit these restricted categories. For a comprehensive overview of structure learning algorithms, including those applicable to mixed-type data, we refer the reader to the works of Kitson et al. (2023) and Zanga et al. (2022). In the context of patient data, the study from Delucchi et al. (2022) has discussed further details and strategies for handling this scenario, particularly in relation to the abn package and the widely used `bnlearn` package (Scutari, 2010).

The abn package overcomes this limitation through its additive model formulation, which generalizes the usual (Bayesian) multivariable regression to accommodate multiple dependent variables. Additionally, the abn package offers a comprehensive suite of features for model selection, structure learning, and parameter estimation. It includes exact and greedy search algorithms for structure learning and allows for integrating prior expert knowledge into the model selection process by specifying structural constraints. For model selection, a Bayesian and an information-theoretic model scoring approach are available, allowing users to choose between a Bayesian and frequentist paradigm. To our knowledge, this feature is not available in other software. Furthermore, it supports mixed-effect models to control one-layer clustering, making it suitable, e.g., for handling data from different sources.

Previous versions of the abn package have been successfully used in various fields, including epidemiology Kratzer & Furrer (2018) and health Delucchi et al. (2022). Despite its promise, the abn package encountered historical obstacles. Sporadic maintenance and an incomplete codebase hindered its full potential. Recognizing the need for enhancement, we undertook a substantial upgrade and meticulously addressed legacy issues, revamped the codebase, and introduced significant improvements. The latest version 3 of abn is now a robust and reliable tool for BN analysis. Applying the latest standards for open-source software, we guarantee active maintenance of abn. Future updates are planned to enhance its functionality and user experience further. We highly value feedback from the user community, which will guide our ongoing developments.

In summary, abn sets itself apart by emphasizing ABNs and its exhaustive features for model selection and structure learning. Its unique contribution is the implementation of mixed-effect BN models, thereby extending its applicability to a broader range of complex, multivariate datasets of mixed, continuous, and discrete data.

Implementation

As outlined in Kratzer et al. (2023), the package's comprehensive framework integrates the mixed-effects model for clustered data, considering data heterogeneity and grouping effects. However, this was confined to a Bayesian context and was only a preliminary implementation. With the release of abn major version 3, this was completed with an implementation under the information-theoretic (`method = "mle"`) setting.

Analyzing hierarchical or grouped data, i.e., observations nested within higher-level units, requires statistical models with group-varying parameters (e.g., mixed-effect models). The abn package facilitates single-layer clustering, where observations are grouped. These clusters are assumed to be independent, but intra-cluster observations may exhibit correlation (e.g., students within schools, patient-specific measurements over time, etc.). The ABN model is fitted independently as a varying intercept model, where the intercept can vary while the slope is assumed constant across all group levels.

Under the frequentist paradigm (`method = "mle"`), abn employs the `lme4` package (Bates et al., 2015) to fit generalized linear mixed models for each of the Binomial, Poisson, and Gaussian distributed variables. For multinomial distributed variables, abn fits a multinomial baseline category logit model with random effects using the `mclogit` package (Elff, 2022). Currently, only single-layer clustering is supported (e.g., for `method = "mle"`, this corresponds to a random intercept model).

With a Bayesian approach (`method = "bayes"`), abn utilizes its own implementation of the Laplace approximation as well as the `INLA` package (Martins et al., 2013) to fit a single-level hierarchical model for Binomial, Poisson, and Gaussian distributed variables.

Furthermore, the code base has been enhanced to be more efficient, reliable, and user-friendly through code optimization, regular reviews, and continuous integration practices. We have adhered to the latest open-source software standards, including active maintenance of abn. Future updates to augment its functionality are planned via a flexible roadmap. User feedback is valued through open communication channels, which will steer our ongoing developments. Consequently, the latest version of abn is now a robust and reliable tool for BN analysis.

Validation and Testing

A comprehensive set of documented case studies has been published to validate the abn package (see the abn website). The numerical accuracy and quality assurance exercises were demonstrated in Kratzer et al. (2023). A rigorous testing framework is implemented using the testthat package (Wickham, 2011), which is executed as part of an extensive continuous integration pipeline designed explicitly for non-standard R packages that rely on Rcpp (Eddelbuettel et al., 2023) and JAGS (Plummer, 2003). Additional documentation and resources are available on the abn website for further reference and guidance.

Availability

The latest version of the abn package along with additional information on the installaiton process for various operating sytsems can be found on GitHub.

Acknowledgments

The development of the abn package would not have been possible without the significant contributions of the former developers whose efforts have been instrumental in shaping this project. We acknowledge the contributions of Fraser Iain Lewis, Marta Pittavino, Gilles Kratzer, and Kalina Cherneva, in particular. We extend our gratitude to the faculty staff at the Department of Mathematical Modeling and Machine Learning, University of Zurich (UZH), and the Department of Mathematics, UZH, who maintain the research and teaching infrastructure. Our appreciation also goes to the UZH and the ZHAW for their financial support. We want to highlight the funding from both the Zurich University of Applied Sciences (ZHAW) and the Digitalization Initiative of the Zurich Higher Education Institutions (DIZH), which were instrumental in realizing this project, particularly within the context of the "Modeling of multicentric and dynamic stroke health data" and "Stroke DynamiX" projects, respectively. This work was conducted as part of M.D.'s PhD project, co-supervised by Prof. Dr. Sven Hirsch (ZHAW) and Prof. Dr. Reinhard Furrer (UZH).

References

Bates, D., Mächler, M., Bolker, B., & Walker, S. (2015). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, *67*(1), 1–48. https://doi.org/10.18637/jss.v067.i01

Boettcher, S. G., & Dethlefsen, C. (2003). deal: A package for learning Bayesian networks. *Journal of Statistical Software*, *8*, 1–40. https://doi.org/10.18637/jss.v008.i20

Delucchi, M., Spinner, G. R., Scutari, M., Bijlenga, P., Morel, S., Friedrich, C. M., Furrer, R., & Hirsch, S. (2022). Bayesian network analysis reveals the interplay of intracranial aneurysm rupture risk factors. *Computers in Biology and Medicine*, *147*, 105740. https://doi.org/10.1016/j.compbiomed.2022.105740

Eddelbuettel, D., Francois, R., Allaire, J., Ushey, K., Kou, Q., Russell, N., Ucar, I., Bates, D., & Chambers, J. (2023). *Rcpp: Seamless R and C++ integration*. https://CRAN.R-project.org/package=Rcpp

Elff, M. (2022). *mclogit: Multinomial logit models, with or without random effects or overdispersion*. https://CRAN.R-project.org/package=mclogit

Franzin, A., Sambo, F., & Di Camillo, B. (2017). bnstruct: An R package for Bayesian network structure learning in the presence of missing data. *Bioinformatics*, *33*(8), 1250–1252. https://doi.org/10.1093/bioinformatics/btw807

Hartnack, S., Odoch, T., Kratzer, G., Furrer, R., Wasteson, Y., L'Abée-Lund, T. M., & Skjerve, E. (2019). Additive Bayesian networks for antimicrobial resistance and potential risk factors in non-typhoidal Salmonella isolates from layer hens in Uganda. *BMC Veterinary Research*, *15*, 212. https://doi.org/10.1186/s12917-019-1965-y

Højsgaard, S. (2012). Graphical independence networks with the gRain package for R. *Journal of Statistical Software*, *46*, 1–26. https://doi.org/10.18637/jss.v046.i10

Kalisch, M., Mächler, M., Colombo, D., Maathuis, M. H., & Bühlmann, P. (2012). Causal inference using graphical models with the R package pcalg. *Journal of Statistical Software*, *47*(1), 1–26. https://doi.org/10.18637/jss.v047.i11

Kitson, N. K., Constantinou, A. C., Guo, Z., Liu, Y., & Chobtham, K. (2023). A survey of Bayesian network structure learning. *Artificial Intelligence Review*, *56*(8), 8721–8814. https://doi.org/10.1007/s10462-022-10351-w

Kratzer, G., & Furrer, R. (2018). Information-theoretic scoring rules to learn additive Bayesian network applied to epidemiology. *arXiv:1808.01126 [Cs, Stat]*. https://doi.org/10.48550/arXiv.1808.01126

Kratzer, G., Lewis, F. I., Comin, A., Pittavino, M., & Furrer, R. (2023). Additive Bayesian network modeling with the R package abn. *Journal of Statistical Software*, *105*, 1–41. https://doi.org/10.18637/jss.v105.i08

Kratzer, G., Lewis, F. I., Willi, B., Meli, M. L., Boretti, F. S., Hofmann-Lehmann, R., Torgerson, P., Furrer, R., & Hartnack, S. (2020). Bayesian network modeling applied to feline calicivirus infection among cats in Switzerland. *Frontiers in Veterinary Science*, *7*. https://doi.org/10.3389/fvets.2020.00073

Martins, T. G., Simpson, D., Lindgren, F., & Rue, H. (2013). Bayesian computing with INLA: New features. *Computational Statistics and Data Analysis*, *67*, 68–83. https://doi.org/10.1016/j.csda.2013.04.014

Pittavino, M., Dreyfus, A., Heuer, C., Benschop, J., Wilson, P., Collins-Emerson, J., Torgerson, P. R., & Furrer, R. (2017). Comparison between generalized linear modelling and additive Bayesian network; identification of factors associated with the incidence of antibodies against Leptospira interrogans sv Pomona in meat workers in New Zealand. *Acta Tropica*, *173*, 191–199. https://doi.org/10.1016/j.actatropica.2017.04.034

Plummer, M. (2003). JAGS: A program for analysis of Bayesian graphical models using Gibbs sampling. *Proceedings of the 3rd International Workshop on Distributed Statistical Computing*, 1–10.

Scutari, M. (2010). Learning Bayesian networks with the bnlearn R package. *Journal of Statistical Software*, *35*, 1–22. https://doi.org/10.18637/jss.v035.i03

Tsagris, M. (2021). A new scalable Bayesian network learning algorithm with applications to economics. *Computational Economics*, *57*(1), 341–367. https://doi.org/10.1007/s10614-020-10065-7

Wickham, H. (2011). testthat: Get started with testing. *The R Journal*, *3*, 5–10. https://doi.org/10.32614/rj-2011-002

Zanga, A., Ozkirimli, E., & Stella, F. (2022). A survey on causal discovery: Theory and practice. *International Journal of Approximate Reasoning*, *151*, 101–129. https://doi.org/10.1016/j.ijar.2022.09.004