SEPTEMBER 2025
VOL. 32 NO. 3

INTERNATIONAL SOCIETY FOR BAYESIAN ANALYSIS

THE ISBA BULLETIN

OFFICIAL BULLETIN OF THE INTERNATIONAL SOCIETY FOR BAYESIAN ANALYSIS

SOFTWARE HIGHLIGHT

Luca Porta Mana
pgl@portamana.org

INFERNO: INFERENCE IN R WITH BAYESIAN NONPARAMETRICS

Population inference and Bayesian nonparametrics

A very important kind of inference in research fields such as medicine is *population inference*, often also called "density inference" or "density regression". Its general goal is to infer the frequency distribution of some variates in a population. This is different, for instance, from *functional regression*, where the goal is to infer the functional relationship – assumed to exist – between a set of predictor variates and a target or "predictand" variate. In population inference the existence of a functional relation cannot be assumed. In fact there may not even be a clear distinction between predictor and predictand variates. A typical goal is the inference of frequency distributions within particular sub-populations or sub-groups, thus all kinds of conditional probabilities are required. A clinician may be interested in the statistics and probability of a medical condition given a symptom, but also that of a symptom given a medical condition; and maybe only within subjects of a given sex or age. De Finetti's theorem (see e.g. Bernardo and Smith, 2000, §§ 4.2, 4.3, 4.6) lies at the heart of population-inference methods; a particularly brilliant discussion is given by Lindley & Novick (1981).

Sadly many researchers still approach population-inference problems by means of p-values or other frequentist practices, which only give limited, coarse, and not seldom misleading results about a population's frequency distribution. Some researchers adopt Bayesian methods but limit themselves to *parametric* ones, which make very restrictive and possibly unrealistic assumptions about the population's distribution; as opposed to *nonparametric* ones, which don't.

Until a couple decades ago the use of parametric methods, and maybe even of frequentist practices, was justified by pragmatic reasons. Better methods were computationally too costly or unfeasible. Population-inference problems were low-dimensional; one could *visually* check whether the assumptions were appropriate and the results reasonable. But today these reasons cannot earnestly be given (Walker, 2010). Bayesian nonparametric methods have become computationally feasible for many kinds of inference. Many inference problems today involve from tens to thousands of variates; it is impossible to visually check in such high-dimensional spaces whether frequentist practices or parametric assumptions are acceptable, or by how much they err. Results may be affected by large parametric-modelling errors (Draper, 1995).

But one reason still exist today for why Bayesian nonparametric methods are avoided: *lack of user-friendly software*. Many clinical researchers might like to try a Bayesian nonparametric analysis, but cannot: they would need to study Markov-chain Monte Carlo techniques, programming languages to implement the latter, and a read about plethora of debated practices to "assess convergence". Most clinicians do not have time to learn all this even if they wanted to. Also, available packages for Bayesian nonparametrics are not quite suited to population inference. Some of them focus on functional regression, which as discussed above is not an appropriate assumption. Some make a priori distinctions between predictor and predictand variates, limiting the range of useful inferences. Most still require Monte Carlo programming expertise.

The R-package **inferno**[1] was built to try to remedy the lack of user-friendly software of this kind.

[1] https://pglpm.github.io/inferno/

Use and features

The application of the package is simple. The user first provides a data sample S of variates from a population, for instance age, sex, symptom, disease, and kind of treatment of a number of patients. The package can work with any combination of continuous, discrete ordinal, nominal, and binary variates. Continuous variates can be defined in bounded intervals, and can also have boundary values with finite probability mass, as it may happen with censoring. The package cannot handle yet periodic variates or variates with complex topology, such as images. The data sample and the variate characteristics (type, domain, possible censoring) are provided by the user as two CSV files to an R function called `learn()`.

The package then runs a Markov-chain Monte Carlo computation, using parallel CPUs if available, to find the probability distribution for all possible joint frequency distributions of the variates. The result is saved in an R object called `learnt`. The crucial point here is that this computation is automatic and does not require any further control from the user, who is simply informed at regular intervals about the expected end time of the computation.

Once the computation has finished, the user can inquire multiple times about any of the following:

- For a new unit of the population, say a new patient, the conditional probability (density) for the values of *any* set of variates $A{=}a, B{=}b, \dots$ given *any* other set $C{=}c, D{=}d, \dots$:

$$\mathrm{P}(A{=}a, B{=}b, \dots \mid C{=}c, D{=}d, \dots, S) \tag{1}$$

  Such a probability could for instance be used in medical decision making (Sox et al., 2024; Hunink et al., 2014). The conditional can be empty, and tail probabilities (e.g. $A \geq a$, $C \leq c$) can also be requested.

- The probability distribution for the conditional *frequencies*, in the whole unsampled population, of the values of any set of variates given any other set (possibly empty). If we denote the frequency distribution by F and a specific value by f, this probability can be written

$$\mathrm{P}\bigl[F(A{=}a, \dots \mid C{=}c, \dots) = f \bigm| S\bigr] \,\mathrm{d}f \,. \tag{2}$$

  This probability distribution is represented by Monte Carlo samples f_i drawn from it.

- The mutual information (MacKay, 2005, Ch. 8) between any two sets of quantities.

Probabilities (1) and (2) are connected by a variant of de Finetti's theorem:

$$\mathrm{P}(A{=}a, \dots \mid C{=}c, \dots, S) = \int f \,\mathrm{P}\bigl[F(A{=}a, \dots \mid C{=}c, \dots) = f \bigm| S\bigr] \,\mathrm{d}f \,. \tag{3}$$

In a manner of speaking, the probability distribution (2) expresses how much the probability value (1) could change, if it were updated by sampling the whole population. It expresses the uncertainty in the statistical results owing to finite sample size.

The first two calculations are made using the R function `Pr(Y, X, learnt)`. The user provides a list `Y` of predictand variates and values of interest; an optional, analogous list `X` of predictors; and the `learnt` object produced by the `learn()` function. The calculation of mutual information is made by the R function `mutualinfo(Y1names, Y2names, X)`, where the first two arguments are the variate sets of interest, and the optional third argument is a set of variate values to conditionalize upon.

The package allows the user to visualize the probabilities above when just one predictand and one predictor variates are involved. If the user saves the results of the `Pr()` function in some object `probs`, say, then the visualization is done by simply calling `plot(probs)`.

Figure 1 shows two examples from a study (Porta Mana et al., 2023) about conversion from Mild Cognitive Impairment (MCI) to Alzheimer's Disease (AD); they can be used to further illustrate the probability distributions discussed above.

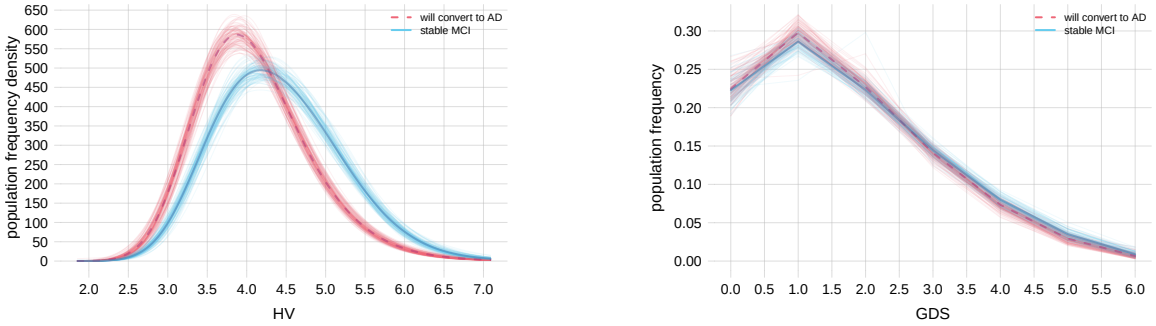

Figure 1

In the plot on the left, the predictand variate is Hippocampal Volume (HV); the predictor is the binary, yes/no variate 'will convert to AD' (cAD). The thick solid blue line and thick dashed red line are the conditional probability distributions (omitting the sample-data dependence)

$$\mathrm{P}(HV \mid cAD{=}\mathrm{N}) \,, \qquad \mathrm{P}(HV \mid cAD{=}\mathrm{Y}) \,.$$

The cloud of thinner blue lines that surrounds the first distribution above represents the probability distribution of possible frequency distributions

$$\mathrm{P}\big[F(HV \mid cAD{=}\mathrm{N})\big]$$

each thin line is a sample from this distribution.

Looking at this plot, a clinician can immediately see that the frequency distributions of hippocampal volume in the sub-populations of patients that will convert to Alzheimer's, and in those who won't, are different. Such a difference is almost certain even considering the uncertainty from the finite sample size.

The plot on the right is analogous but for the predictand variate 'Geriatric Depression Scale' (GDS) in stead of hippocampal volume. In this case the frequency distributions for the two sub-populations cannot be distinguished within the finite-sample uncertainty.

It must be remarked that it is exactly this kind of differences and uncertainties that clinical researchers often try to clumsily capture by using p-values. In private communications, several practitioners expressed contentment at the possibility of visualizing the estimates of different sub-population statistics and even the uncertainties they carry because of finite sample size.

The quick analysis above is mostly qualitative, but concrete numbers, such as quantiles and expected values, can be easily produced. This becomes necessary when many variates are considered jointly and visualization is impossible. In such high-dimensional cases the package allows the user to compute any kind of distance between two frequency distributions (such as Hellinger or Kantorovich or Shannon-Jansen distance, or relative entropy) as well as its credibility intervals. The computation of the mutual information between any two sets of variates gives moreover a quantitative measure of their association that does not depend on assumptions such as linearity or gaussianity.

The ability to swap the "predictor" and "predictand" roles of any variates is illustrated in the plots of fig. 2, parallel to those of fig. 1. The thicker lines show the probabilities

$$\mathrm{P}(cAD{=}\mathrm{Y} \mid HV) \,, \qquad \mathrm{P}(cAD{=}\mathrm{Y} \mid HV)$$

for various values of the HV variate, which is now a predictor. In the plot on the left we see for example that among individuals having hippocampal volume around 5.0, between 30% and 40%

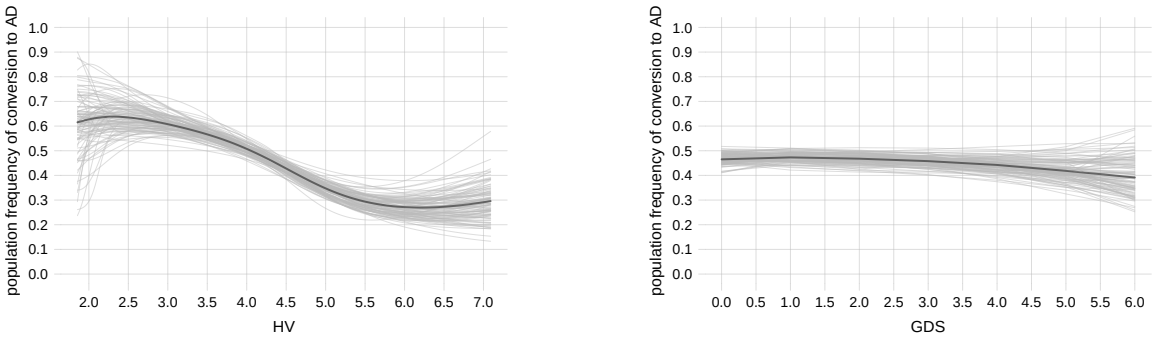

Figure 2

will convert to Alzheimer's. Among those with volume around 2.0, we can only say that between roughly 45% and 80% will convert; in this case the finite-sample size (few samples with this HV value) leads to a much larger uncertainty of the frequency estimates.

The possibility of swapping predictor and predictand roles makes it also possible to implement, on the fly, correction for base rates (Lindley and Novick, 1981, § 4)

Plots and calculations like the ones above are of course nothing new in Bayesian nonparametrics. The point here is that the user can produce them just by first inputting the population sample in the package, waiting for the Monte Carlo computation to finish, and then simply asking about variates of interest and plotting the results. The sequence of commands could be as simple as

```
learnt <- learn(data = 'data.csv', metadata = 'metadata.csv')
## Information about MCMC and expected end time...

probs <- Pr(Y = list(A = seq(0, 8, 0.1)), X = list(C = c('yes', 'no')), learnt)

plot(probs)
```

More extensive use examples are given in the package's main vignette[2]. Readers of the ISBA Bulletin are probably interested in the internals of the package, which now be briefly discussed.

Internals

In Bayesian nonparametric population inference the posterior probability distribution is over the set of all possible frequency distributions. The mathematical representation of this set is therefore crucial. The package uses the ingenious representation of a distribution as a mixture of product kernels discussed by Dunson & Bhattacharya 2011. For instance, for variates A and B the frequency distribution is written as

$$F(A,B) = \sum_i w_i \, K_A(A \mid \alpha_i) \, K_B(B \mid \beta_i) \tag{4}$$

where w_i are positive and normalized weights, K_A is a distribution for variate A depending on parameters α_i, and similarly for K_B. The product is easily generalized to any number of variates. In principle the sum should be countably infinite, but as discussed in Ishwaran & Zarepour 2002a; 2002b and in Dunson & Bhattacharya 2011, it is possible to truncate it to a finite number of values if an appropriate prior distribution is used for the weights w_i. Thus a frequency distribution F is effectively represented – non-uniquely – by a large but finite set of parameters (w_i, α_i, β_i).

[2] https://pglpm.github.io/inferno/articles/inferno_start.html

The above representation was somewhat deprecated in more recent works (e.g. Wade et al., 2014b,a) which, however, consider inference problems where variates have clear predictor or predictand roles. As previously discussed, in many research fields there are no such a-priori roles and any variate could be both. A representation that can easily swap the two roles and does not overemphasizing either is therefore most appropriate. The representation (4) leads to very simple and symmetric analytical expressions for the conditional of A given B and vice versa, as well as any marginal:

$$F(A \mid B) = \sum_i \frac{w_i \, K_A(A \mid \alpha_i) \, K_B(B \mid \beta_i)}{\sum_j w_j K_B(B \mid \beta_j)} \qquad F(B \mid A) = \sum_i \frac{w_i \, K_B(B \mid \beta_i) \, K_A(A \mid \alpha_i)}{\sum_j w_j K_A(A \mid \alpha_j)}$$

$$F(A) = \sum_i w_i \, K_A(A \mid \alpha_i) \; .$$

The prior over the parameters representing the frequency distribution is a Dirichlet-process mixture. It allows for distributions having multiple peaks, also sharp ones (with a lower probability). For information about the kernels K used for the different kinds of variates, and the prior hyperdistribution see the technical manual[3].

The Markov-chain Monte Carlo computation is based on Gibbs sampling; several chains are run in parallel. The stopping rule implements some variations of the methods discussed in Vehtari et al. (2021). The package takes the following stand:

• For this kind of inference problems there is no need for tens of thousands or more independent Monte Carlo samples. The default total number of putatively-independent samples is just 3600. The mean of such samples is the probability (1), and the error in this mean is thus around $\sqrt{3600} = 60$ times smaller than the "variability" of this probability, given by eq. (2).

• It is acceptable that the sampling has not fully converged. Consider that machine-learning algorithms such as neural networks are effectively Bayesian nonparametric functional regressors, and their training is essentially an *unconverged* Monte Carlo sampling (MacKay, 1992; Gal, 2016; Mandt et al., 2017; Huszár, 2017). This "sampling" stops at a *local* maximum of the posterior, and all samples are discarded. Thanks to this lack of full convergence the computation is fast, and despite of this lack the inferences can still be impressive. The same stance can be adopted in Bayesian nonparametrics: despite a possible lack of full convergence, the results can still be impressive and informative; and the uncertainty in the results can moreover be assessed, whereas it is discarded in neural-network training.

[[in progress]]

References

J.-M. Bernardo and A. F. Smith. *Bayesian Theory*. Wiley series in probability and mathematical statistics. Wiley, New York, repr. edition, 2000. doi:10.1002/9780470316870. First publ. 1994.

J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. Dawid, D. Heckerman, A. F. M. Smith, and M. West, editors. *Bayesian Statistics 9*. Oxford University Press, Oxford, 2011. doi:10.1093/acprof:oso/9780199694587.001.0001.

D. Draper. Assessment and propagation of model uncertainty. *J. R. Stat. Soc. B*, 57(1):45–70, 1995. doi:10.1111/j.2517-6161.1995.tb02015.x. See also discussion and reply in Spiegelhalter et al. (1995).

D. B. Dunson and A. Bhattacharya. Nonparametric Bayes regression and classification through mixtures of product kernels. In *Bernardo et al. (2011)*, pages 145–158. 2011. doi:10.1093/acprof:oso/9780199694587.003.0005.

Y. Gal. *Uncertainty in Deep Learning*. PhD thesis, University of Cambridge, Cambridge, 2016. https://www.cs.ox.ac.uk/people/yarin.gal/website/thesis/thesis.pdf.

[3] https://github.com/pglpm/inferno/raw/main/development/manual/optimal_predictor_machine.pdf

N. L. Hjort, C. Holmes, P. Müller, and S. G. Walker, editors. *Bayesian Nonparametrics*. Cambridge series in statistical and probabilistic mathematics. Cambridge University Press, Cambridge, 2010. doi:10.1017/CBO9780511802478.

M. G. M. Hunink, M. C. Weinstein, E. Wittenberg, M. F. Drummond, J. S. Pliskin, J. B. Wong, and P. P. Glasziou. *Decision Making in Health and Medicine: Integrating Evidence and Values*. Cambridge University Press, Cambridge, 2 edition, 2014. doi:10.1017/CBO9781139506779. First publ. 2001.

F. Huszár. Everything that works works because it's Bayesian: Why deep nets generalize?, 2017. http://www.inference.vc/everything-that-works-works-because-its-bayesian-2.

H. Ishwaran and M. Zarepour. Dirichlet prior sieves in finite normal mixtures. *Stat. Sinica*, 12(3):941–963, 2002a. http://www3.stat.sinica.edu.tw/statistica/J12n3/j12n316/j12n316.htm.

H. Ishwaran and M. Zarepour. Exact and approximate sum representations for the Dirichlet process. *Can. J. Stat.*, 30(2):269–283, 2002b. doi:10.2307/3315951.

D. V. Lindley and M. R. Novick. The role of exchangeability in inference. *Ann. Stat.*, 9(1):45–58, 1981. doi:10.1214/aos/1176345331.

D. J. C. MacKay. Bayesian interpolation. *Neural Comput.*, 4(3):415–447, 1992. https://www.inference.org.uk/mackay/PhD.html, doi:10.1162/neco.1992.4.3.415.

D. J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, Cambridge, version 7.2 (4th pr.) edition, 2005. https://www.inference.org.uk/itila/book.html. First publ. 1995. See also video lectures at https://videolectures.net/events/course_information_theory_pattern_recognition.

S. Mandt, M. D. Hoffman, and D. M. Blei. Stochastic gradient descent as approximate Bayesian inference. *J. Mach. Learn. Res.*, 18:134, 2017. https://www.jmlr.org/papers/v18/17-214.html.

P. G. L. Porta Mana, I. Rye, A. Vik, M. Kociński, A. Lundervold, A. J. Lundervold, and A. S. Lundervold. Personalized prognosis & treatment using an optimal predictor machine: An example study on conversion from mild cognitive impairment to Alzheimer's disease, 2023. Open Science Framework doi:10.31219/osf.io/8nr56.

H. C. Sox, M. C. Higgins, D. K. Owens, and G. S. Schmidler. *Medical Decision Making*. Wiley, New York, 3 edition, 2024. doi:10.1002/9781119627876. First publ. 1988.

D. J. Spiegelhalter, A. P. Grieve, D. V. Lindley, D. V. Lindley, J. B. Copas, A. J. G. Cairns, C. Chatfield, G. Box, D. Cox, J. W. Tukey, A. E. Raftery, M. Aitkin, G. A. Barnard, P. Donnelly, A. S. C. Ehrenberg, A. E. Gelfand, B. K. Mallick, A. Gelman, X.-L. Meng, C. A. Glasbey, G. J. Gibson, R. Glendinning, U. Hjorth, R. E. Kass, L. Wasserman, M. Lavine, D. Madigan, B. M. Pötscher, J. W. Pratt, A. F. de Vos, B. J. Worton, A. Zellner, and D. Draper. Discussion of the paper by Draper. *J. R. Stat. Soc. B*, 57(1):71–97, 1995. doi:10.1111/j.2517-6161.1995.tb02016.x. See Draper (1995).

A. Vehtari, A. Gelman, D. Simpson, B. Carpenter, and P.-C. Bürkner. Rank-normalization, folding, and localization: An improved $\hat{R}$ for assessing convergence of MCMC. *Bayesian Anal.*, 16(2):667–718, 2021. doi:10.1214/20-BA1221. With invited discussion by D. Vats, G. Jones, C. M. Hans, T. Moins, J. Arbel, A. Dutfoy, S. Girard, and rejoinder.

S. Wade, D. B. Dunson, S. Petrone, and L. Trippa. Improving prediction from Dirichlet process mixtures via enrichment. *J. Mach. Learn. Res.*, 15(30):1041–1071, 2014a. http://jmlr.org/papers/v15/wade14a.html.

S. Wade, S. G. Walker, and S. Petrone. A predictive study of Dirichlet process mixture models for curve fitting. *Scand. J. Stat.*, 41(3):580–605, 2014b. doi:10.1111/sjos.12047.

S. G. Walker. Bayesian nonparametric methods: motivation and ideas. In *Hjort et al. (2010)*, chapter 1, pages 22–34. 2010. doi:10.1017/CBO9780511802478.002.