The Annals of Applied Statistics
2020, Vol. 14, No. 4, 2099–2100
https://doi.org/10.1214/20-AOAS1411
Main article: https://doi.org/10.1214/20-AOAS1410

RESPONSE TO 'FITTING A FOLDED NORMAL DISTRIBUTION WITHOUT EM'

BY SUNGKYU JUNG[1], MARK FOSKEY[2] AND J. S. MARRON[3]

[1]*Department of Statistics, Seoul National University, sungkyu@snu.ac.kr*
[2]*Accuray Incorporated, mark.foskey@gmail.com*
[3]*Department of Statistics and Operations Research, University of North Carolina at Chapel Hill, marron@unc.edu*

We thank the Editor-in-Chief for providing us the opportunity to respond to the letter by Professor Iain MacDonald.

Professor MacDonald pointed out that MLEs of parameters for the univariate folded normal distribution can be efficiently obtained via a ready-made general-purpose optimizer, rather than deriving EM algorithms as we originally proposed. We welcome his comment and take this opportunity to numerically compare these two solutions and to raise a couple of related questions.

We have compared the folded normal MLEs obtained by our EM algorithm (implemented in Matlab) and by the `nlm` function (of R) for the 15 sets of data used in Jung, Foskey and Marron (2011). While the EM estimates are nearly identical to the `nlm`-estimates (up to the fifth digit), the computation time needed for the EM algorithm is nearly one tenth of `nlm`: 0.01 seconds for the EM and 0.13 seconds for the optimizer. This might be due to the fact that most of the data points are far from the boundary, that is, there was almost no folding with $\hat{\mu}/\hat{\sigma} \approx 15$. To mimic the situation where some folding must occur, we have translated the data toward the origin (and taken the absolute values) so that $\hat{\mu}/\hat{\sigma} \approx 1$. The results were similar; both estimates are nearly identical, but the EM algorithm took 0.01 seconds while the optimizer needed 0.11 seconds. (The data and codes for this experiment can be found in the Supplementary Material (Jung, Foskey and Marron (2020)) and also at https://github.com/sungkyujung/EMvsNLM.) While directly applying general-purpose optimizers can be considered as an efficient solution, carefully designed *statistical algorithms* (such as the EM algorithm) can be even more efficient.

We want to point out that a similar problem of fitting a folded normal was also considered in Eltzner, Huckemann and Mardia (2018). There, the authors considered fitting a transformed folded normal distribution; writing $f_{\mu,\sigma}(x)$ for the folded normal density, the problem of Eltzner, Huckermann and Mardia is to obtain MLE of (μ, σ) from the sample $X_i \sim g(x) = c(\mu, \sigma) \sin^d(x) f_{\mu,\sigma}(x)$. Here, d is a known natural number and $c(\mu, \sigma)$ is the normalizing constant (with no closed form expression available). Eltzner et al. did not attempt to construct an EM algorithm. A potential reason for that is the fact that an M-step does not have a closed-form solution, at least not without an ingenious formulation. For such problems, a numerical optimizer indeed provides satisfactory solutions for the likelihood problem, as Professor MacDonald suggested.

The problems considered in Jung, Foskey and Marron (2011) and Eltzner, Huckemann and Mardia (2018) are motivated by preventing overfitting of the least-square circles and spheres to directional data on the unit hypersphere. The circles and spheres on the hypersphere correspond to certain classes of curves and surfaces in $\mathbb{R}^d$, including straight lines and hyperplanes. The folded normal distribution was used to decide whether nonlinear dimension reduction is more advantageous than linear dimension reduction. The use of the folded normal distribution for such purposes is superseded by other proposals in Jung, Dryden and Marron (2012)

Received October 2020.

and Kim, Schulz and Jung (2020). We believe there still is room and need for improvement in the problem of "linear vs. nonlinear" dimension reduction, and cordially invite researchers to tackle this problem.

Finally, we note that folded *multivariate* distributions have recently gained some attention in the context of compositional data analysis. For example, Tsagris and Stewart (2020) and Scealy and Welsh (2011) considered fitting a folded multivariate normal distribution and a folded Kent distribution, respectively. Will a universal numerical optimization give performance similar to Tsagris and Stewart's (carefully conceived) EM algorithm in terms of computation times? The same question applies to fitting folded or truncated Kent distributions.

SUPPLEMENTARY MATERIAL

Data and codes for computer experiments (DOI: 10.1214/20-AOAS1411SUPP; .zip). We provide R and Matlab codes and data comparing the accuracy and the computation speed of algorithms.

REFERENCES

ELTZNER, B., HUCKEMANN, S. and MARDIA, K. V. (2018). Torus principal component analysis with applications to RNA structure. *Ann. Appl. Stat.* **12** 1332–1359. MR3834306 https://doi.org/10.1214/17-AOAS1115

JUNG, S., DRYDEN, I. L. and MARRON, J. S. (2012). Analysis of principal nested spheres. *Biometrika* **99** 551–568. MR2966769 https://doi.org/10.1093/biomet/ass022

JUNG, S., FOSKEY, M. and MARRON, J. S. (2011). Principal arc analysis on direct product manifolds. *Ann. Appl. Stat.* **5** 578–603. MR2810410 https://doi.org/10.1214/10-AOAS370

JUNG, S., FOSKEY, M. and MARRON, J. S. (2020). Supplement to "Response to 'Fitting a folded normal distribution without EM'." https://doi.org/10.1214/20-AOAS1411SUPP.

KIM, B., SCHULZ, J. and JUNG, S. (2020). Kurtosis test of modality for rotationally symmetric distributions on hyperspheres. *J. Multivariate Anal.* **178** 104603, 17. MR4076342 https://doi.org/10.1016/j.jmva.2020.104603

SCEALY, J. L. and WELSH, A. H. (2011). Regression for compositional data by using distributions defined on the hypersphere. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* **73** 351–375. MR2815780 https://doi.org/10.1111/j.1467-9868.2010.00766.x

TSAGRIS, M. and STEWART, C. (2020). A folded model for compositional data analysis. *Aust. N. Z. J. Stat.* **62** 249–277.