

# iTensor: An R package for independent component analysis-based matrix/tensor decomposition

Koki Tsuyuzaki<sup>1, 2</sup>

<sup>1</sup> Department of Artificial Intelligence Medicine, Graduate School of Medicine, Chiba University, Japan <sup>2</sup> Laboratory for Bioinformatics Research, RIKEN Center for Biosystems Dynamics Research, Japan

DOI: [DOIunavailable](#)

## Software

- [Review](#) ↗
- [Repository](#) ↗
- [Archive](#) ↗

Editor: [Pending Editor](#) ↗

## Reviewers:

- [@Pending Reviewers](#)

Submitted: N/A

Published: N/A

## License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).

## Summary

Independent Component Analysis (ICA) is a widely used algorithm to extract a small number of mutually independent source signals in high-dimensional data. There are many applications of ICA in signal processing (Calhoun, 2006; Hyvärinen, 2000), neuroscience (Calhoun, 2006; Hyvärinen, 2000), bioinformatics (Trapnell, 2014), and causal discovery (Shimizu, 2006). ICA has been applied to matrix data but there is a growing demand to apply ICA to more heterogeneous data such as multiple matrices and tensors (high-dimensional arrays), which are higher-order data structures than matrices (Akaho, 1999; Calhoun, 2009; Pfister, 2018; Vasilescu, 2005). To meet these requirements, I originally developed iTensor, which is an R/CRAN package to perform some ICA-based matrix/tensor decomposition algorithms (<https://cran.r-project.org/web/packages/iTensor/index.html>).

## Statement of need

Currently, the most comprehensive implementation for ICA-related algorithms is the Group ICA of fMRI Toolbox (GIFT, <http://mialab.mrn.org/software/gift>), but it is not freely available because it is implemented in MATLAB. Also, some open-source software is implemented in R and Python but those only focus on fewer algorithms. To fill this gap, I originally implemented some ICA-based matrix/tensor decomposition algorithms in R.

iTensor provides the ICA-based matrix/tensor decomposition functions as follows:

- ICA: ICA (3 classic models including InfoMax (Amari, 1995; Bell, 1995), ExtInfoMax (Lee, 1999), and FastICA (Hyvärinen, 1999))
- ICA2: ICA (9 modern models including JADE (Cardoso, 1993), AuxICA1/2 (Ono, 2010), SIMBEC (Cruces, 2001), AMUSE (Tong, 1991), SOBI (Belouchrani, 1997), FOBI (Cardoso, 1989), ProDenICA (Hastie, 2002), and RICA (Le, 2011))
- MICA: Multimodal ICA (Akaho, 1999)
- GroupICA: Group ICA (Calhoun, 2009; Pfister, 2018)
- MultilinearICA: Multilinear ICA (Vasilescu, 2005)

I also implemented CorrIndex (Sobhani, 2022), which is a performance index to evaluate ICA results.

## Example

ICA and plots in [Figure 1](#) can be easily reproduced on any machine where R is pre-installed by using the following commands in R:

```
# Install package required (one per computer)
install.packages("BiocManager")
```

```
BiocManager::install(c("mixOmics", "iTensor"))

# Load required package (once per R instance)
library("iTensor")

# Load Toy data
data1 <- toyModel("ICA_Type1")

# Perform ICA
set.seed(1234)
out.JADE <- ICA2(X=data1$X_observed, J=3, algorithm="JADE")

# Source Signal extracted by ICA (If it becomes an upright square,
# the calculation is successful)
pairs(data1$X_observed)
pairs(Re(out.JADE$S))

# CorrIndex (0.2211509, the closer to 0, the better the performance)
CorrIndex(cor(data1$S, Re(out.JADE$S)))
```

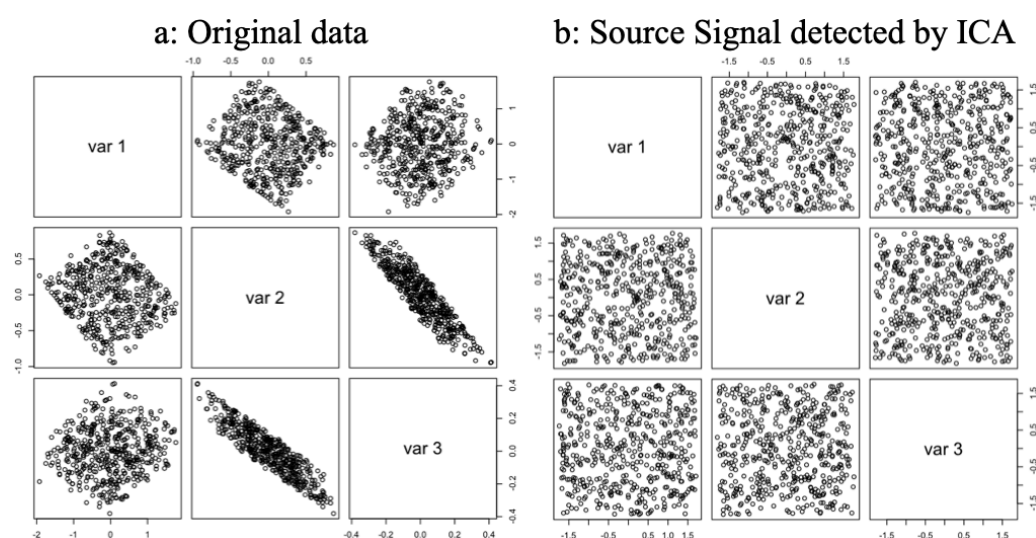


Figure 1: ICA with time-independent sub-gaussian data.

## Related work

There are some packages to perform ICA for matrix, matrices, and tensor but such packages focus on only a few algorithms. *iTensor* is the most comprehensive and unified package to perform ICA-based matrix/tensor decomposition as follows.

Table 1: Existing ICA-related packages

Name (function or package)	Language	ICA for matrix	ICA for matrices	ICA for tensor	Reference
scikit-learn	Python	1	-	-	Pedregosa (2011)
MNE	Python	1	-	-	Gramfort (2013)

Name (function or package)	Language	ICA for matrix	ICA for matrices	ICA for tensor	Reference
rica	MATLAB	1	-	-	Le (2011)
fastICA	R	1	-	-	Hyvarinen (1999)
fICA	R	1	-	-	Hyvarinen (1999)
JADE	R	1	-	-	Cardoso (1993)
ProDenICA	R	1	-	-	Hastie (2002)
ica	R	3	-	-	Calhoun (2006); Hyvärinen (2000)
groupICA	R	-	1	-	Pfister (2018)
coroICA	R/Python/MATLAB	-	2	-	Pfister (2019)
BrainVoyager	MATLAB	1	-	-	Goebel (2006); Formisano (2006)
FMRLAB	MATLAB	1	-	-	Perlbarg (2007)
GIFT	MATLAB	14	1	-	Wei (2022)
tensorBSS	R	-	-	6	Virta (2016)
iTensor	R	12	2	1	This paper

For MICA (Akaho, 1999) and Multilinear ICA (Vasilescu, 2005), there is no package without iTensor to perform them.

## References

- Akaho, S. et al. (1999). MICA: Multimodal independent component analysis. *IJCNN'99*, 2, 927–932. <https://doi.org/10.1109/ijcnn.1999.831077>
- Amari, S. et al. (1995). A new learning algorithm for blind signal separation. *NIPS 1995*.
- Bell, A. J. et al. (1995). An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7(6), 1129–1159. <https://doi.org/10.7551/mitpress/7011.003.0009>
- Belouchrani, A. et al. (1997). A blind source separation technique using second-order statistics. *IEEE Transactions on Signal Processing*, 45(2), 434–444.
- Calhoun, V. D. et al. (2006). Unmixing fMRI with independent component analysis. *IEEE Eng Med Biol Mag*, 25(2), 79–90. <https://doi.org/10.1109/memb.2006.1607672>
- Calhoun, V. D. et al. (2009). A review of group ICA for fMRI data and ICA for joint inference of imaging, genetic, and ERP data. *Neuroimage*, 45(1 Suppl), S163–S172. <https://doi.org/10.1016/j.neuroimage.2008.10.057>
- Cardoso, J.-F. (1989). Source separation using higher order moments. *International Conference on Acoustics, Speech, and Signal Processing*, 4, 2109–2112. <https://doi.org/10.1109/icassp.1989.266878>
- Cardoso, J.-F. (1993). Blind beamforming for non-gaussian signals. *IEEE Proceedings F*, 140(6), 362–370. <https://doi.org/10.1049/ip-f-2.1993.0054>
- Cruces, S. et al. (2001). Criteria for the simultaneous blind extraction of arbitrary groups of sources. *International Conference on ICA and BSS*, 740–745.

- Formisano, E. et al. (2006). Fundamentals of data analysis methods in fMRI. *Advanced Image Processing in Magnetic Resonance Imaging*.
- Goebel, R. et al. (2006). Analysis of FIAC data with BrainVoyager QX: From single-subject to cortically aligned group GLM analysis and self-organizing group ICA. *Human Brain Mapping*, 27(5), 392–401.
- Gramfort, A. et al. (2013). MEG and EEG data analysis with MNE-python. *Frontiers in Neuroscience*, 7(267), 1–13. <https://doi.org/10.3389/fnins.2013.00267>
- Hastie, T. et al. (2002). Independent components analysis through product density estimation. *NIPS 2002*.
- Hyvarinen, A. (1999). Fast and robust fixed-point algorithms for independent component analysis. *IEEE Transactions on Neural Networks*, 10(3), 626–634. <https://doi.org/10.1109/72.761722>
- Hyvärinen, A. et al. (2000). Independent component analysis: Algorithms and applications. *Neural Network*, 13(4-5), 411–430.
- Le, Q. et al. (2011). ICA with reconstruction cost for efficient overcomplete feature learning. *NIPS 2011*.
- Lee, et al., T. W. (1999). Independent component analysis using an extended infomax algorithm for mixed subgaussian and supergaussian sources. *Neural Computation*, 11(2), 417–441. <https://doi.org/10.1162/089976699300016719>
- Ono, N. et al. (2010). Auxiliary-function-based independent component analysis for supergaussian sources. *Lecture Notes in Computer Science*, 6365, 165–172. [https://doi.org/10.1007/978-3-642-15995-4\\_21](https://doi.org/10.1007/978-3-642-15995-4_21)
- Pedregosa, F. et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(85), 2825–2830.
- Perlberg, V. et al. (2007). CORSICA: Correction structured noise in fMRI by automatic identification of ICA components. *Magnetic Resonance Imaging*, 25(1), 35–46.
- Pfister, N. et al. (2018). groupICA: Independent component analysis for grouped data. *arXiv*.
- Pfister, N. et al. (2019). Robustifying independent component analysis by adjusting for group-wise stationary noise. *Journal of Machine Learning Research*, 20(147), 1–50.
- Shimizu, S. et al. (2006). A linear non-gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 7, 2003–2030.
- Sobhani, E. et al. (2022). CorrIndex: A permutation invariant performance index. *Signal Processing*, 195, 108457. <https://doi.org/10.1016/j.sigpro.2022.108457>
- Tong, L. et al. (1991). Indeterminacy and identifiability of blind identification. *IEEE Transactions on Circuits and Systems*, 38(5), 499–509. <https://doi.org/10.1109/31.76486>
- Trapnell, C. et al. (2014). The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nature Biotechnology*, 32(4), 381–386. <https://doi.org/10.1038/nbt.2859>
- Vasilescu, M. A. O. et al. (2005). Multilinear independent component analysis. *IEEE CVPR 2005*.
- Virta, J. et al. (2016). Applying fully tensorial ICA to fMRI data. *Signal Processing in Medicine and Biology Symposium*. <https://doi.org/10.1109/spmb.2016.7846858>
- Wei, P. et al. (2022). Comparing the reliability of different ICA algorithms for fMRI analysis. *PLoS ONE*, 17(6), e0270556. <https://doi.org/10.1371/journal.pone.0270556>