Boosting Independent Component Analysis

YunPeng Li   ZhaoHui Ye

Abstract—Independent component analysis is intended to recover the mutually independent components from their linear mixtures. This technique has been widely used in many fields, such as data analysis, signal processing, and machine learning. To alleviate the dependency on prior knowledge concerning unknown sources, many nonparametric methods have been proposed. In this paper, we present a novel boosting-based algorithm for independent component analysis. Our algorithm consists of maximizing likelihood estimation via boosting and seeking unmixing matrix by the fixed-point method. A variety of experiments validate its performance compared with many of the presently known algorithms.

Index Terms—Independent component analysis, boosting, nonparametric maximum likelihood estimation, fixed-point method.

I. INTRODUCTION

INDEPENDENT Component Analysis (ICA) has received much attention in recent years due to its effective methodology for various problems, such as feature extraction, blind source separation, and exploratory data analysis. In the ICA model, a m dimensional random vector $\mathbf{x} = (x_1, \cdots, x_m)^T$ is observed as a linear mixture of the m independent sources $\mathbf{s} = (s_1, \cdots, s_m)^T$,

$$\mathbf{x} = \mathbf{A}\mathbf{s} \tag{1}$$

where $\mathbf{s}$ contains the mutually independent components, and $\mathbf{A}$ is called the mixing matrix. An unmixing matrix $\mathbf{W}$ is used to determine the estimated sources $\mathbf{y} = (y_1, \cdots, y_m)^T$ from the observed mixtures $\mathbf{x}$, where the source's estimation $\mathbf{y}$ is the scaling and permutation of the original source $\mathbf{s}$. The recovering process can be modeled as

$$\mathbf{y} = \mathbf{W}\mathbf{x} \tag{2}$$

Without loss of generality, several assumptions are clear in ICA : (i) $\mathrm{E}(\mathbf{s}) = \mathbf{0}$ and $\mathrm{Cov}(\mathbf{s}) = \mathbf{I}$; (ii) each s_i is independent distributed and at most one of the sources is Gaussian [1]; (iii) both $\mathbf{A}$ and $\mathbf{W}$ are invertible matrices. Since centering and whitening are often performed as preprocessing steps in the context of linear ICA, $\mathbf{W}$ is restricted to be an orthonormal matrix $\mathbf{W}\mathbf{W}^T = \mathbf{I}$.

ICA was firstly introduced to the neural network domain in the 1980s [2]. It was not until 1994 [1] that the theory of ICA was established. The most popular method is to optimize some contrast functions to achieve source separation. These contrast functions were usually chosen to represent the measure of independence or non-Gaussianity, for example, the mutual information [3], [4], the maximum entropy (or negentropy) [5]–[7], and the nonlinear decorrelation [8], [9].

YunPeng Li (corresponding author) is with the D-ITET, ETH Zurich, Zurich, Switzerland e-mail: yunpli@isi.ee.ethz.ch

ZhaoHui Ye is with the Department of Automation, Tsinghua University, Beijing, China e-mail: yezhaohui@mail.tsinghua.edu.cn

In addition, higher-order moments methods [10], [11] were also designed to estimate the unknown sources. It has been pointed out that these contrast functions are related to the sources' density distributions in the parametric maximum likelihood estimation [12]–[15]. These parametric methods' performance are dependent on the choices of contrast functions or prior assumptions on the unknown sources' distributions. Although it [5] has been shown that small misspecifications (for parametric methods) in the source densities do not affect local consistency of the estimation concerning the unmixing matrix $\mathbf{W}$, Amari [16] demonstrated that ICA methods have to estimate the sources' densities (nonparametric methods) to achieve the information bound concerning $\mathbf{W}$.

Nonparametric ICA's in-depth analysis and asymptotic efficiency were firstly available in [17]. There are currently two kinds of nonparametric ICA methods: the restriction methods and the regularization methods. In restriction methods, each source s_i belongs to certain density family or owns special structure, such as Gaussian mixtures models [18], [19], kernel density distributions [20], [21], and log-concave family [22]. The regularization methods determine the unknown sources via maximizing a penalized likelihood [23]. For other recent ICA approaches, see also [24]–[29].

Recently, we have successfully applied boosting to the nonparametric maximum likelihood estimation (boosting NPMLE) [30]. In this paper, we further propose a novel ICA method called $BoostingICA$ based on our previous work. The proposed approach adaptively includes only those basis functions that contribute significantly to the sources' estimation. Real data experiments validate its competitive performance with other popular or recent ICA methods.

II. NONPARAMETRIC MAXIMUM LIKELIHOOD ESTIMATION IN ICA

In this section, nonparametric maximum likelihood estimation is applied to solve the ICA problem. Maximizing the likelihood can be viewed as a joint maximization over the unmixing matrix $\mathbf{W}$ and the estimated sources' densities, fixing one argument and maximizing over the other.

Let $p_i(y_i)$ be the estimated density distribution for single component y_i. Owing to the independence among y_i, the estimated sources' density distribution $p_{\mathbf{y}}(\mathbf{y})$ can be written as

$$p_{\mathbf{y}}(\mathbf{y}) = \prod_{i=1}^{m} p_i(y_i) \tag{3}$$

Since there is a linear transform in equation (2), the joint density of the observed mixtures $\mathbf{x}$ is

$$p_{\mathbf{x}}(\mathbf{x}; \mathbf{W}, \mathbf{y}) = |\mathbf{det}(\mathbf{W})| p_{\mathbf{y}}(\mathbf{W}\mathbf{x}) = \prod_{i=1}^{m} p_i(\mathbf{w}_i^T \mathbf{x}) = \prod_{i=1}^{m} p_i(y_i) \tag{4}$$

where $\mathbf{w}_i^T$ is the i_{th} row in the orthonormal matrix $\mathbf{W}$ and $|\det(\mathbf{W}))| = 1$.

For parametric ICA, We have to determine the density distribution $p_i(y_i)$ based on the prior assumptions beforehand, which might result in the unwanted density mismatching. To keep the density distribution of y_i unspecified, nonparametric methods are demanded. In the proposed method, we choose to model the estimated source y_i with Gibbs distribution in equation (5).

$$p_i(y_i) = \frac{e^{f_i(y_i)}}{\int e^{f_i(y_i)}dy_i} \qquad (5)$$

where $f_i(y_i)$ is assumed to be a smooth function in $\mathbb{R}$, and the denominator of equation (5) is the partition function. Given N independent identically distributed samples $\{\mathbf{x}_j\}_1^N$, we can estimate the source's sample $y_i^j = \mathbf{w}_i^T \mathbf{x}_j$.

To simplify the integral in equation (5), we construct a grid of L values y_i^{*l} (ascending order) with Δ_i step , and let the corresponding frequency q_i^l be

$$q_i^l = \sum_{j=1}^N \mathbb{I}(y_i^j \in (y_i^{*l} - \Delta_i/2, y_i^{*l} + \Delta_i/2])/N \qquad (6)$$

where $\mathbb{I}(.)$ is the indicator function. The support of $p_i(y_i)$ is restricted in $[y_i^{*1} - \Delta_i/2, y_i^{*L} + \Delta_i/2]$, and $p_i(y_i)$ is then transformed as

$$p_i(y_i) = e^{f_i(y_i)} / \sum_{l=1}^L \Delta_i e^{f_i(y_i^{*l})} \qquad (7)$$

It [30], [31] has been shown that the log-likelihood $\mathcal{L}(\mathbf{w}_i, f_i)$ concerning y_i can be simplified to the following modified form

$$\mathcal{L}(\mathbf{w}_i, f_i) = \sum_{l=1}^L q_i^l \ln p_i(y_i^{*l}) = \sum_{l=1}^L q_i^l f_i(y_i^{*l}) - \Delta_i e^{f_i(y_i^{*l})} \qquad (8)$$

and the maximum log-likelihood is obtained when the partition function $\sum_{l=1}^L \Delta_i e^{f_i(y_i^{*l})} = 1$. Thus, the total modified log-likelihood in our method is

$$\mathcal{L}(\{\mathbf{w}_i, f_i\}_{i=1}^m) = \sum_{i=1}^m \mathcal{L}(\mathbf{w}_i, f_i) \qquad (9)$$

The key of $BoostingICA$ is to optimize equation (9) until convergence by joint maximization [22], [23], [32], and the two iterative stages are illustrated in Algorithm 1. Similar joint maximization has been used in past researches concerning both projection pursuit [33], [34] and ICA [22], [23].

A. Estimating the source's density via boosting

To apply the boosting principle (forward-stagewise regression) [35], [36] to nonparametric maximum likelihood estimation [30], we regard $f_i(y_i)$ in equation (8) as a combination of weak learners $b_i(y_i; \gamma_{i,k})$,

$$f_i(y_i) = \sum_{k=1}^M b_i(y_i; \gamma_{i,k}) \qquad (10)$$

where M is the number of boosting iterations and k is the index. Each single weak learner $b_i(y_i; \gamma_{i,k})$ is characterized

---

Algorithm 1 BoostingICA algorithm

1: Initialization
2: maximum iterations $maxit$
3: grid value L
4: boosting iterations M
5: degree of freedom df
6: **for** $it = 1$ to $maxit$ **do**
7:     **for** $i = 1$ to m **do**
8:         fixing $\mathbf{W}$, each f_i is estimated by boosting NPMLE (Algorithm 2).
9:     **end for**
10:     given f_i, $\mathbf{W}$ is restricted to be orthonormal and calculated via the fixed-point method (Algorithm 3).
11: **end for**
12: Output
13: unmixing matrix $\mathbf{W}$
14: estimated densities $\{f_i\}_{i=1}^m$

---

by a set of parameters $\gamma_{i,k}$ and is trained on the weighted data at the k_{th} iteration. We define the f_i at the k_{th} iteration as f_i^k

$$f_i^k(y_i) = \sum_{r=1}^k b(y_i; \gamma_{i,r}) = f_i^{k-1}(x) + b(y_i; \gamma_{i,k}) \qquad (11)$$

and the subproblem based on the former f_i^{k-1} becomes

$$\max_{\gamma_{i,k}} \quad \mathcal{L}(f_i^{k-1}(y_i) + b(y_i; \gamma_{i,k})) \qquad (12)$$

We maximize equation (12) via second-order approximation around f_i^{k-1}, and the corresponding iterative reweighted least squares (IRLS) [37] (Line 14) and updating strategy for the next iteration (Line 10-11) are shown in Algorithm 2. A penalty term $\lambda J(b_i(y_i; \gamma_{i,k}))$ (Line 14) is added to the original least squares to restrict the model complexity of $b_i(y_i; \gamma_{i,k})$, where λ is the Lagrange multiplier and $J(b_i(y_i; \gamma_{i,k}))$ is a nonnegative function. ω_l^k, Y_l^k (Line 10-11) are the weight and response of y_i^{*l} at the k_{th} iteration, and they are simultaneously updated in the next iteration. Once all the weak learners have been trained, $f_i(y_i)$ is the combination of whole M weak learners.

Inspired by the past researches [23], [30], smooth spline is chosen as the weak learner for two reasons: Line 14 in Algorithm 2 can be efficiently computed in $\mathcal{O}(L)$ time [23], and then the first and second derivatives of smooth spline (concerning y_i) are immediately available [23], [38]. The penalty term is defined as $J(b_i(y_i; \gamma_{i,k})) = \int \left[b_i''(y_i; \gamma_{i,k}) \right]^2 dy_i$ (roughness penalty penalizes the curvature of function), and the degree of freedom [39] df (corresponding to the prechosen λ) is defined implicitly by the trace of the linear smoother in Algorithm 2 (Line 14). With df's increase from 2 to L (corresponding to λ's decrease from ∞ to 0), $b_i(y_i; \gamma_{i,k})$ changes from a simple line fit to the ordinary least squares fit. For single weak learner, we only need to determine its hyper-parameter df beforehand, simply choosing the small value (slightly greater than 2, and our method's default value is 3) for df to restrict the model complexity (slighter more complex than the simple line fit).

Algorithm 2 Estimating the density distribution of y_i by boosting NPMLE

1: Initialization
2: $b_i(y_i; \gamma_{i,0}) \leftarrow -\frac{1}{2}y_i^2 - \frac{1}{2}\ln 2\pi$
3: $f_i^0(y_i) \leftarrow b_i(y_i; \gamma_{i,0})$
4: $f_i^{0'}(y_i) \leftarrow b_i'(y_i; \gamma_{i,0})$
5: $f_i^{0''}(y_i) \leftarrow b_i''(y_i; \gamma_{i,0})$
6: compute the L grid values $\{y_i^{*l}\}_{l=1}^L$ and Δ_i from $\{y_i^j\}_{j=1}^N$.
7: $\omega_l^0 \leftarrow \Delta_i \frac{1}{\sqrt{2\pi}} e^{-\frac{y_i^{*l^2}}{2}}, 1 \leq l \leq L$
8: **for** $k = 1$ to M **do**
9:    **for** $l = 1$ to L **do**
10:       $\omega_l^k \leftarrow \omega_l^{k-1} e^{b_i(y_i^{*l}; \gamma_{i,k-1})}$
11:       $Y_l^k \leftarrow \frac{q_l - \omega_l^k}{\omega_l^k}$
12:    **end for**
13:    solve the reweighted least squares with respect to $\gamma_{i,k}$
14:    $\sum_{l=1}^L \frac{1}{2}\omega_l^k (b_i(y_i^{*l}; \gamma_{i,k}) - Y_l^k)^2 + \lambda J(b_i(y_i; \gamma_{i,k}))$
15:    $f_i^k(y_i) \leftarrow f_i^{k-1}(y_i) + b_i(y_i; \gamma_{i,k})$
16:    $f_i^{k'}(y_i) \leftarrow f_i^{k-1'}(y_i) + b_i'(y_i; \gamma_{i,k})$
17:    $f_i^{k''}(y_i) \leftarrow f_i^{k-1''}(y_i) + b_i''(y_i; \gamma_{i,k})$
18: **end for**
19: Output
20: $f_i(y_i) \leftarrow f_i^M(y_i)$
21: $f_i'(y_i) \leftarrow f_i^{M'}(y_i)$
22: $f_i''(y_i) \leftarrow f_i^{M''}(y_i)$

In fact, when df is fixed with the default value, the number of boosting iterations M is the only hyper-parameter in the proposed method. Fortunately, it [40] has been shown that even when the training error is close to zero (log-likelihood is close to the upper-bound in our algorithm), boosting provably tends to increase the performance on the testing data with the increase of M. We later provide robust experiment to validate this surprising phenomenon.

Figure 1 shows the estimated log-likelihood, and the true rotate angles (concerning the mixing matrix $\mathbf{A}$) of mixtures $\mathbf{x}$ are plotted for comparison. Smooth spline surprisingly performs well when $M = 1$, and it successfully finds the ground-truth in two cases as the increase of boosting iterations M.

B. Fixed-point method for estimating the separation matrix

Given the fixed f_i, partition function becomes constant. Since there exits a whitening preprocessing stage for the observation x and the unmixing matrix $\mathbf{W}$ is orthonormal, fixed-point method developed in $FastICA$ [5] can be used in our algorithm. Then, the log-likelihood is maximized by an approximative Newton method with quadratic convergence (see Algorithm 3).

C. Discussions of BoostingICA

Since there is a trade-off between df and M in boosting, simple hyper-parameters tuning strategies are available:
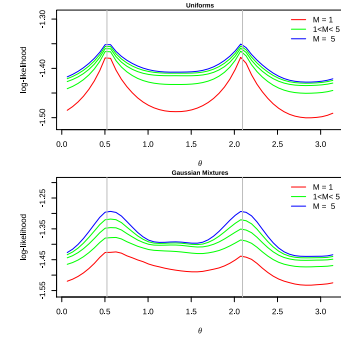

Fig. 1. The log-likelihood estimations (smooth spline as weak learner, $df = 3$) in two dimension using different boosting iterations M. The independent components are uniforms in the top figure and Gaussian mixtures in the bottom figure. In each figure, the ordinate is the log-likelihood estimations and the abscissa represents the rotate angles θ of $\mathbf{W}$. The vertical lines are shown to indicate true rotate angles.

Algorithm 3 Estimating the unmixing matrix $\mathbf{W}$ by fixed-point method

---

1: **for** $i = 1$ to m **do**
2:    $\mathbf{w}_i \leftarrow \mathrm{E}\{\mathbf{x}f_i'(\mathbf{w}_i^T\mathbf{x})\} - \mathrm{E}\{f_i''(\mathbf{w}_i^T\mathbf{x})\}\mathbf{w}_i$
3: **end for**
4: $\mathbf{W} \leftarrow (\mathbf{w}_1, \cdots, \mathbf{w}_m)^T$
5: symmetric decorrelation
6: $\mathbf{W} \leftarrow (\mathbf{W}\mathbf{W}^T)^{-\frac{1}{2}}\mathbf{W}$

- decreasing the value of boosting iterations M might lead to the reduce of elapsed time, at the cost of weakening the density estimation for unknown sources.
- slightly increasing the weak learner's model complexity df might benefit the density estimation without sacrificing time efficiency.
- we can level up the separation performance by slightly increasing df, and reduce the elapsed time via cutting down M.

Owing that s_i is zero mean and unit variance, the most uncertain (or circumspect) initialization (Line 2) in Algorithm 2 is the logarithm of the standard Gaussian, which is different from the zero initialization in the original boosting NPMLE [30].

III. EXPERIMENTS AND RESULTS

A. Implementation details

Our experiments were conducted on the R development platform with Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz, and we intend to illustrate the proposed method's comparable performance with other popular or recent ICA algorithms.

ICA algorithms used in our experiments are shown in Table I. $FixNA2$ is the recent blind source separation algorithms based on nonlinear auto-correlation, and $EFICA$ [41] is a statistically efficient version of the $FastICA$. All methods share the same maximum iterations $maxit = 20$, and $EFICA$'s elapsed time (with superscript *) was measured on the Matlab platform while other methods were R implementations.

TABLE I
ICA METHODS USED IN THE EXPERIMENTS.

Methods	Symbols	Parameters
FastICA(G0) [5], [44]	F-G0	$G_0(u) = \frac{1}{4}u^4$
FastICA(G1) [44]	F-G1	$G_1(u) = \ln \cosh(u)$
ProDenICA [23], [44]	PICA	default
FixNA2 [45], [46]	FNA2	default
BoostingICA(smooth spline)	B-SP	$L = 500, \quad df = 3 \quad M = 5$
EFICA [41], [47]	EICA	default

TABLE II
THE AMARI METRICS (MULTIPLIED BY 100), SIR AND CPU ELAPSED
TIME FOR AUDIO SEPARATION TASK (50 REPLICATIONS).

Mean	F-G0	F-G1	PICA	B-SP	FNA2	EICA
Amari metrics	30.51	30.64	**12.41**	13.67	414.86	29.53
SIR	21.98	23.64	**30.14**	29.30	7.02	27.16
Elapsed time (s)	1.73	1.35	**17.54**	20.51	381.15	0.71*

Standard deviation	F-G0	F-G1	PICA	B-SP	FNA2	EICA
Amari metrics	0.00	4.14	0.64	1.01	22.20	4.56
SIR	0.00	0.48	0.35	0.41	0.22	0.71
Elapsed time (s)	0.11	0.10	0.38	0.69	23.69	0.01*

The separation performance is measured by the value of Amari metrics $d(\mathbf{W}, \mathbf{W}_0)$ [42],

$$
\begin{aligned}
&d(\mathbf{W}, \mathbf{W}_0) \\
&= \frac{1}{2m} \sum_{i=1}^{m} \left(\frac{\sum_{j=1}^{m} |r_{ij}|}{\max_j |r_{ij}|} - 1 \right) + \frac{1}{2m} \sum_{j=1}^{m} \left(\frac{\sum_{i=1}^{m} |r_{ij}|}{\max_i |r_{ij}|} - 1 \right)
\end{aligned}
\tag{13}
$$

where $r_{ij} = (\mathbf{W}\mathbf{W}_0^{-1})_{ij}$, $\mathbf{W}_0$ is the known truth. $d(\mathbf{W}, \mathbf{W}_0)$ is equal to zero if and only $\mathbf{W}$ and $\mathbf{W}_0$ are equivalence. Besides Amari metrics, we also used the signal-to-interference ratio SIR (SIR function in *ica* package [43]) as the criterion, where larger SIR indicates better performance in ICA.

B. Audio separation task

Two kinds of sources were used in the audio separation experiment: 6 speech recordings were from male (MJ60_07, MA03_01, MJ57_03) and female (FC14_04,FC18_06, FD19_06) speakers in TSP dataset [48], and 18 different random sources were from the past researches [9], [23]. These 24 sources ($N = 15200$) were mixed by an invertible matrix to produce the high dimensional observed mixtures $\mathbf{x}$, and such procedure was replicated 50 times.

As can be seen from the Table II, B-SP, PICA, EICA acquired the best three separation performances, and $FNA2$ was actually not converged. Although B-SP performed slightly better than PICA in this experiment, we will later illustrate the simple way to improve our methods' separation performance and time efficiency.

C. Natural scene images separation task

We designed an images separation experiment in this subsection, where the three gray-scale images were chosen from the *ICS* [49] package. These images depict a forest

road, cat and sheep, and they have been used in many ICA researches [49]. We vectorized them to arrive into a $130^2 \times 3$ data matrix and we fixed the mixing matrix as $\mathbf{A} = [0.8, 0.3, -0.3; 0.2, -0.8, 0.7; 0.3, 0.2, 0.3]$ (column-wise).

The results are recorded in Table III, where PICA performed better than B-SP ($df = 3, M = 5$) both in Amari/SIR and elapsed time. To improve the performance of B-SP, we have to level up its separation performance and cut down its elapsed time at the same time. Following the instructions in Subsection II-C, we successfully found the appropriate tuning parameters ($df = 8, M = 3$) for B-SP in few attempts, and we show them at the bottom of Table III.

TABLE III
THE AMARI METRICS (MULTIPLIED BY 100), SIR AND CPU ELAPSED
TIME FOR IMAGE SEPARATION TASK.

methods	Amari metrics	SIR	Elapsed time (s)
F-G0	38.61	10.69	0.24
F-G1	54.88	9.04	0.17
PICA	19.04	16.39	1.95
B-SP ($df = 3, M = 5$)	24.45	14.55	2.62
FNA2	32.17	11.84	0.44
EICA	37.87	10.11	0.24*
B-SP ($df = 5, M = 3$)	19.90	15.91	1.65
B-SP ($df = 8, M = 3$)	**18.73**	**16.56**	**1.66**

Once the running time is not our key interest, it might be an appropriate way to improve the separation performance via simply increasing M. The corresponding robust experiments on *ICS* data are shown in Figure 2, and B-SP outperformed PICA when $M \geq 48$.
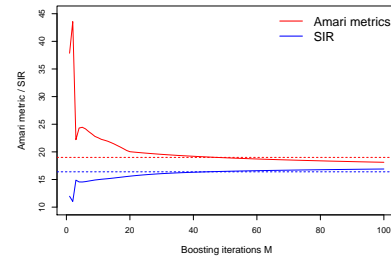

Fig. 2. Separation performances of B-SP on *ICS* images mixtures. The ordinate is the Amari metrics/SIR, and the abscissa represents boosting iterations M. The performance of PICA is shown in the dot line for comparison, and B-SP overwhelmed PICA when $M \geq 48$.

IV. CONCLUSION

In this paper, we introduce boosting to the nonparametric independent component analysis to alleviate the density mismatching between unknown sources and their estimations. The proposed algorithm is based on our earlier research [30], and it is a joint likelihood maximization between boosting NPMLE and fixed-point method. The proposed method $BoostingICA$ is concise and efficient, a list of experiments have illustrated its competitive performance with other popular or recent ICA algorithms.

REFERENCES

[1] P. Comon, "Independent component analysis, a new concept?" *Signal Process.*, vol. 36, pp. 287–314, 1994.

[2] J. Herault and C. Jutten, "Space or time adaptive signal processing by neural network models," in *Neural Networks for Computing*, ser. American Institute of Physics Conference Series, vol. 151, Aug. 1986, pp. 206–211.

[3] A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, no. 6, pp. 1129–1159, 1995.

[4] T. Lee, M. Girolami, and T. Sejnowski, "Independent component analysis using an extended infomax algorithm for mixed subgaussian and supergaussian sources," *Neural computation*, vol. 11, no. 2, p. 417—441, February 1999.

[5] A. Hyvarinen, "Fast and robust fixed-point algorithms for independent component analysis," *IEEE Transactions on Neural Networks*, vol. 10, no. 3, pp. 626–634, 1999.

[6] E. Learned-Miller and J. Fisher III, "Ica using spacings estimates of entropy," *Journal of Machine Learning Research*, vol. 4, pp. 1271–1295, 01 2003.

[7] Y. Li, "Second-order approximation of minimum discrimination information in independent component analysis," *IEEE Signal Processing Letters*, vol. 29, pp. 334–338, 2022.

[8] C. Jutten and J. Herault, "Blind separation of sources, part i: An adaptive algorithm based on neuromimetic architecture," *Signal Processing*, vol. 24, no. 1, pp. 1 – 10, 1991.

[9] F. Bach and M. Jordan, "Kernel independent component analysis," *Journal of Machine Learning Research*, vol. 3, pp. 1–48, 03 2003.

[10] J.-F. Cardoso, "Source separation using higher order moments," in *International Conference on Acoustics, Speech, and Signal Processing,*, 1989, pp. 2109–2112 vol.4.

[11] J.-F. Cardoso and A. Souloumiac, "Blind beamforming for non-gaussian signals," *IEE Proceedings F - Radar and Signal Processing*, vol. 140, no. 6, pp. 362–370, 1993.

[12] B. A. Pearlmutter and L. Parra, "A context-sensitive generalization of ICA," in *In International Conference on Neural Information Processing*, 1996, pp. 151–157.

[13] D. J. C. MacKay, "Maximum likelihood and covariant algorithms for independent component analysis," Tech. Rep., 1996.

[14] Dinh Tuan Pham and P. Garat, "Blind separation of mixture of independent sources through a quasi-maximum likelihood approach," *IEEE Transactions on Signal Processing*, vol. 45, no. 7, pp. 1712–1725, 1997.

[15] J.-F. Cardoso, "Infomax and maximum likelihood for blind source separation," *IEEE Signal Processing Letters*, vol. 4, no. 4, pp. 112–114, 1997.

[16] S.-i. Amari, "Independent component analysis (ica) and method of estimating functions," *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 85, no. 3, pp. 540–547, 2002.

[17] A. Chen and P. J. Bickel, "Efficient independent component analysis," *Ann. Statist.*, vol. 34, no. 6, pp. 2825–2855, 12 2006.

[18] E. Moulines, J.-F. Cardoso, and E. Gassiat, "Maximum likelihood for blind separation and deconvolution of noisy signals using mixture models," in *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, 1997, pp. 3617–3620 vol.5.

[19] M. Welling and M. Weber, "A constrained em algorithm for independent component analysis," *Neural Computation*, vol. 13, no. 3, pp. 677–689, 2001.

[20] N. Vlassis and Y. Motomura, "Efficient source adaptivity in independent component analysis," *IEEE Transactions on Neural Networks*, vol. 12, pp. 200–1, 2001.

[21] A. Eloyan and S. K. Ghosh, "A semiparametric approach to source separation using independent component analysis," *Computational Statistics and Data Analysis*, vol. 58, pp. 383–396, 2013, the Third Special Issue on Statistical Signal Extraction and Filtering.

[22] R. Samworth and M. Yuan, "Independent component analysis via nonparametric maximum likelihood estimation," *The Annals of Statistics*, vol. 40, 06 2012.

[23] T. Hastie and R. Tibshirani, "Independent components analysis through product density estimation," in *Advances in Neural Information Processing Systems*, S. Becker, S. Thrun, and K. Obermayer, Eds., vol. 15. MIT Press, 2003, pp. 665–672.

[24] P. Ilmonen and D. Paindaveine, "Semiparametrically efficient inference based on signed ranks in symmetric independent component models," *The Annals of Statistics*, vol. 39, no. 5, pp. 2448 – 2476, 2011.

[25] J. A. Palmer, K. Kreutz-Delgado, and S. Makeig, "Amica: An adaptive mixture of independent component analyzers with shared components," *Swartz Center for Computatonal Neuroscience, University of California San Diego, Tech. Rep*, 2012.

[26] D. S. Matteson and R. S. Tsay, "Independent component analysis via distance covariance," *Journal of the American Statistical Association*, vol. 112, no. 518, pp. 623–637, 2017.

[27] P. Ablin, J.-F. Cardoso, and A. Gramfort, "Faster independent component analysis by preconditioning with hessian approximations," *IEEE Transactions on Signal Processing*, vol. 66, no. 15, pp. 4040–4049, 2018.

[28] P. Spurek, J. Tabor, L. Struski, and M. Smieja, "Fast independent component analysis algorithm with a simple closed-form solution," *Knowledge-Based Systems*, vol. 161, pp. 26–34, 2018.

[29] A. Podosinnikova, A. Perry, A. S. Wein, F. Bach, A. d'Aspremont, and D. Sontag, "Overcomplete independent component analysis via sdp," in *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, vol. 89, 16–18 Apr 2019, pp. 2583–2592.

[30] Y. Li and Z. Ye, "Boosting in univariate nonparametric maximum likelihood estimation," *IEEE Signal Processing Letters*, vol. 28, pp. 623–627, 2021.

[31] B. W. Silverman, "On the estimation of a probability density function by the maximum penalized likelihood method," *Ann. Statist.*, vol. 10, no. 3, pp. 795–810, 09 1982.

[32] Z. Koldovsky, P. Tichavsky, and E. Oja, "Efficient variant of algorithm fastica for independent component analysis attaining the cramÉr-rao lower bound," *IEEE Transactions on Neural Networks*, vol. 17, no. 5, pp. 1265–1277, 2006.

[33] J. Friedman, W. Stuetzle, and A. Schroeder, "Projection pursuit density estimation," *Journal of the American Statistical Association*, vol. 79, no. 387, pp. 599–608, 1984.

[34] J. Friedman, "Exploratory projection pursuit," *Journal of the American Statistical Association*, vol. 82, pp. 249–266, 1987.

[35] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[36] J. Friedman, "Greedy function approximation: A gradient boostingmachine." *The Annals of Statistics*, vol. 29, no. 5, pp. 1189 – 1232, 2001. [Online]. Available: https://doi.org/10.1214/aos/1013203451

[37] R. Wolke and H. Schwetlick, "Iteratively reweighted least squares: Algorithms, convergence analysis, and numerical comparisons," *SIAM Journal on Scientific and Statistical Computing*, vol. 9, no. 5, pp. 907–921, 1988.

[38] C. D. Boor, *A Practical Guide to Splines (Revised Edition)*, ser. Applied Mathematical Sciences 27. Springer, 2001.

[39] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.*, 2nd ed., ser. Springer Series in Statistics. Springer, 2013.

[40] P. Bartlett, Y. Freund, W. S. Lee, and R. E. Schapire, "Boosting the margin: a new explanation for the effectiveness of voting methods," *The Annals of Statistics*, vol. 26, no. 5, pp. 1651 – 1686, 1998.

[41] Z. Koldovsky, P. Tichavsky, and E. Oja, "Efficient variant of algorithm fastica for independent component analysis attaining the cramÉr-rao lower bound," *IEEE Transactions on Neural Networks*, vol. 17, no. 5, pp. 1265–1277, 2006.

[42] S.-i. Amari, A. Cichocki, and H. Yang, "A new learning algorithm for blind signal separation," *Adv. Neural. Inform. Proc. Sys.*, vol. 8, 12 1999.

[43] N. E. Helwig, *ica: Independent Component Analysis*, 2018, r package version 1.0-2. [Online]. Available: https://CRAN.R-project.org/package=ica

[44] T. Hastie and R. Tibshirani, *ProDenICA: Product Density Estimation for ICA using tilted Gaussian density estimates*, 2010, r package version 1.0. [Online]. Available: https://CRAN.R-project.org/package=ProDenICA

[45] M. Matilainen, J. Miettinen, K. Nordhausen, H. Oja, and S. Taskinen, "On independent component analysis with stochastic volatility models," *Austrian Journal of Statistics*, vol. 46, 2017.

[46] K. Nordhausen, M. Matilainen, J. Miettinen, J. Virta, and S. Taskinen, "Dimension reduction for time series in a blind source separation context using R," *Journal of Statistical Software*, vol. 98, no. 15, pp. 1–30, 2021.

[47] Z. Koldovský, "Efica algorithm," https://asap.ite.tul.cz/downloads/the-efica-algorithm/, 2009, version 2.12.

[48] P. Kabal, "Tsp speech database," *McGill University, Database Version*, vol. 1, no. 0, pp. 09–02, 2002.

[49] K. Nordhausen, H. Oja, and D. E. Tyler, "Tools for exploring multivariate data: The package ics," *Journal of Statistical Software*, vol. 28, no. 1, pp. 1–31, 2008.