A New Algorithm of Non-Gaussian Component Analysis with Radial Kernel Functions

Motoaki Kawanabe (motoaki.kawanabe@first.fraunhofer.de)
Fraunhofer FIRST.IDA, Kekuléstr. 7, 12489 Berlin, Germany

Masashi Sugiyama (sugi@cs.titech.ac.jp)
Department of Computer Science, Tokyo Institute of Technology
2-12-1-W8-74, O-okayama, Meguro-ku, Tokyo, 152-8552, Japan

Gilles Blanchard (blanchar@first.fraunhofer.de)
Fraunhofer FIRST.IDA, Kekuléstr. 7, 12489 Berlin, Germany, and
CNRS, Laboratoire de mathématiques, Université Paris-Sud,
91405 Orsay, France

Klaus-Robert Müller (klaus@first.fraunhofer.de)
Department of Computer Science, Technical University of Berlin
Franklinstr. 28/29, 10587 Berlin, Germany, and
Fraunhofer FIRST.IDA, Kekuléstr. 7, 12489 Berlin, Germany

Abstract

We consider high-dimensional data which contains a linear low-dimensional non-Gaussian structure contaminated with Gaussian noise, and discuss a method to identify this non-Gaussian subspace. For this problem, we provided in our previous work a very general semi-parametric framework called Non-Gaussian Component Analysis (NGCA). NGCA has a uniform probabilistic bound on the error of finding the non-Gaussian components and within this framework, we presented an efficient NGCA algorithm called *Multi-index Projection Pursuit*. The algorithm is justified as an extension of the ordinary projection pursuit (PP) methods and is shown to outperform PP particularly when the data has complicated non-Gaussian structure. However, it turns out that multi-index PP is not optimal in the context of NGCA. In this article, we therefore develop an alternative algorithm called *Iterative Metric Adaptation for radial Kernel functions (IMAK)*, which is theoretically better justifiable within the NGCA framework. We demonstrate that the new algorithm tends to outperform existing methods through numerical examples.

Keywords

linear dimension reduction, non-Gaussian subspace, projection pursuit, semiparametric model Stein's identity

1 Introduction

Suppose that we are given a set of i.i.d. observations $\boldsymbol{X}_i \in \mathbb{R}^d$, $(i = 1, \ldots, n)$ obtained as a sum of a signal $\boldsymbol{S} \in \mathbb{R}^m$ $(m \leq d)$ with an unknown non-Gaussian distribution and a Gaussian noise component $\boldsymbol{N} \in \mathbb{R}^d$:

$$\boldsymbol{X} = A\boldsymbol{S} + \boldsymbol{N}, \tag{1}$$

where A is a $d \times m$ matrix or rank m, $\boldsymbol{N} \sim N(\boldsymbol{0}, \Gamma)$ and $\boldsymbol{S}$ and $\boldsymbol{N}$ are assumed to be independent. The rationale behind this model is that in most real-world applications the 'signal' or 'information' contained in the high-dimensional data is essentially non-Gaussian while the 'rest' can be interpreted as high-dimensional Gaussian noise. Under this modeling assumption, therefore, the tasks are to estimate the relevant non-Gaussian subspace and to recover the low-dimensional non-Gaussian structures by *linear dimension reduction*. Although our goal is dimension reduction, we want to emphasize that we do *not* assume the Gaussian components to be of *smaller* order of magnitude than the signal components. This setting therefore excludes the use of common (nonlinear) dimensionality reduction methods such as PCA, Isomap (Tenenbaum et al., 2000) and LLE (Roweis and Saul, 2000) that are based on the assumption that the data lies, say, on a lower dimensional manifold, up to some small noise distortion.

If the non-Gaussian components S_j $(j = 1, \ldots, m)$ are mutually independent, the model turns out to be the under-complete noisy ICA, and there exist algorithms to extract the independent components in the presence of Gaussian noise (Hyvärinen et al., 2001). However, independence is often a too strict assumption on practical data.

In contrast, Projection Pursuit (PP) (Friedman and Tukey, 1974; Huber, 1985) or FastICA in the deflation mode (Hyvärinen, 1999; Hyvärinen et al., 2001) can also extract dependent non-Gaussian structures. PP iteratively finds directions that maximize a prefixed single non-Gaussianity index. However, it is known that some indices are suitable for finding super-Gaussian components and others are suited for identifying sub-Gaussian components (Hyvärinen et al., 2001). Therefore, PP with a single prefixed non-Gaussianity index tends to give undesired results if the data contains both super- and sub-Gaussian components.

To cope with this problem, two different approaches have been suggested. One is PP with an adaptive single index which learns an efficient index simultaneously from a family of functions. Because it is known that the optimal index in the sense of asymptotic variance depends on the density of each non-Gaussian component (Hyvärinen et al., 2001, Theorem 14.2), even non-parametric density estimators have been integrated in PP. However, due to high computational costs of such algorithms, PP with a fixed index often remains a competitive method. On the other hand, we proposed a very general semiparametric framework called *Non-Gaussian Component Analysis (NGCA)* in our previous research (Blanchard et al., 2006), where the density of the sources is not specified at all. The NGCA framework provides a unified view for combining the results of PPs with *different* indices. Within this framework, we developed an NGCA algorithm called *multi-index PP*. Through numerical examples, we showed that the multi-index PP out-

performs the ordinary single index PP methods in particular when data has complicated non-Gaussian structures.

Although the multi-index extension of PP works better than the original PP, it does not make use of the full potential of the NGCA framework: the NGCA framework does not only provide a unified view for combining the PP results with different indices, but also it naturally defines the optimality criterion for identifying the non-Gaussian components. It turns out that multi-index PP is actually not optimal in the above sense, so there is still room for improvement. In this paper, we thus propose a theoretically more sound NGCA procedure called *Iterative Metric Adaptation for radial Kernel functions (IMAK)*, which can directly optimize the above defined optimality criterion.

The rest of this paper is organized as follows. In the next section we will summarize the NGCA framework and briefly review the multi-index PP algorithm developed in Blanchard et al. (2006). Then in Section 3, the new NGCA algorithm will be proposed. Simulation results will be presented in Section 4, where the new algorithm is compared with existing methods. Researches on adaptive PP/ICA will be summarized in Section 5, showing the relation to our approach. Finally, conclusions and open problems will be discussed in Section 6.

2 Non-Gaussian Component Analysis

2.1 Semiparametric Model of NGCA

The probability density function $p(\boldsymbol{x})$ of the observations defined by the mixing model (1) can be framed in the following semi-parametric form:

$$p(\boldsymbol{x}) = g(T\boldsymbol{x})\phi_\Gamma(\boldsymbol{x}), \tag{2}$$

where T is an unknown linear mapping from $\mathbb{R}^d$ to another subspace $\mathbb{R}^m$, g is an unknown function on $\mathbb{R}^m$ related to the distribution of the source $\boldsymbol{S}$ and ϕ_Γ is a centered Gaussian density with unknown covariance matrix Γ. Note that $g(\cdot)$ is not necessarily a probability density function: in other terms, it does not necessarily integrate to unity, although it is of course necessarily nonnegative. In the appendix, we show more precisely how to formulate the model (1) under the form (2). For now, we comment on the interpretation of this model and define our goals.

In this paper, we assume that the dimension m of the non-Gaussian subspace is known. Eq.(2) takes the form of a semiparametric statistical model, but we would like to avoid direct inference of the Gaussian covariance Γ and the function g which accounts for the non-Gaussian components. We remark that the model (2) includes as particular cases both the pure parametric ($m = 0$) and pure non-parametric ($m = d$) models. In practice, we are interested in an intermediate case where d is large and m is rather small.

Note that the decomposition (2) is non-unique, and in particular neither g nor Γ are actually identifiable in this formulation. However, it can be shown that the following m-dimensional *linear* subspace $\mathcal{I}$ of $\mathbb{R}^d$ is *identifiable* (Theis and Kawanabe, 2006):

$$\mathcal{I} = \mathrm{Ker}(T)^\perp = \mathrm{Range}(T^\top)\,.$$

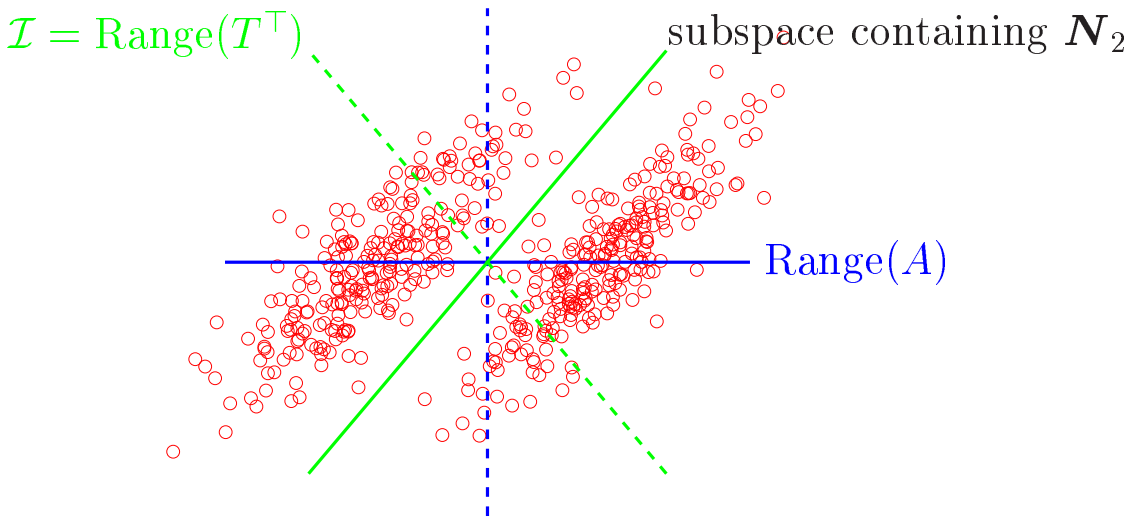


Figure 1: Geometrical interpretation of the non-Gaussian index space $\mathcal{I}$.

We call $\mathcal{I}$ the *non-Gaussian index space*; the goal we set here is the estimation of the space $\mathcal{I}$. Its geometrical meaning is the following: in the model (1), the noise term can be decomposed into two components, $\boldsymbol{N} = \boldsymbol{N}_1 + \boldsymbol{N}_2$, where $\boldsymbol{N}_1 = A\boldsymbol{\eta} \in \mathrm{Range}(A)$ and $\boldsymbol{N}_2$ is restricted in the $(d-m)$-dimensional complementary subspace s.t. $\mathrm{Cov}(\boldsymbol{N}_1, \boldsymbol{N}_2) = 0$ (i.e. $\boldsymbol{N}_1$ and $\boldsymbol{N}_2$ are independent). Thus, we have the representation

$$\boldsymbol{X} = A\widetilde{\boldsymbol{S}} + \boldsymbol{N}_2,$$

where $\widetilde{\boldsymbol{S}} := \boldsymbol{S} + \boldsymbol{\eta}$ and the distribution of the noise term $\boldsymbol{N}_2$ is a $(d-m)$-dimensional degenerated Gaussian independent of $\widetilde{\boldsymbol{S}}$. The subspace $\mathcal{I}$ is then the orthogonal complement of the $(d-m)$-dimensional subspace containing the independent Gaussian component $\boldsymbol{N}_2$ (see Fig. 1). Once we have an estimate of the index space $\mathcal{I}$, we can project out the noise $\boldsymbol{N}_2$ by projecting the data $\boldsymbol{X}$ onto that space. We remark that the projection does not have to be orthogonal, e.g., an oblique projection gives the *best linear unbiased estimator* (Sugiyama et al., 2006). In the representation (2) we can assume that $TA = I_m$ and $T\boldsymbol{X} = \widetilde{\boldsymbol{S}}$ without loss of generality, in which case T corresponds to the demixing matrix in under-complete ICA, but here we are not interested in the individual directions of the components $\widetilde{S}_j$ (which are not assumed to be independent).

2.2 Key Property and Outline of the NGCA Procedure

The main idea underlying NGCA is summed up in the following Proposition (proof in Appendix).

Proposition 1 *Let $\boldsymbol{X}$ be a random variable whose density function $p(\boldsymbol{x})$ satisfies Eq.(2) and suppose that $h(\boldsymbol{x})$ is a smooth real function on $\mathbb{R}^d$. Then under mild regularity*

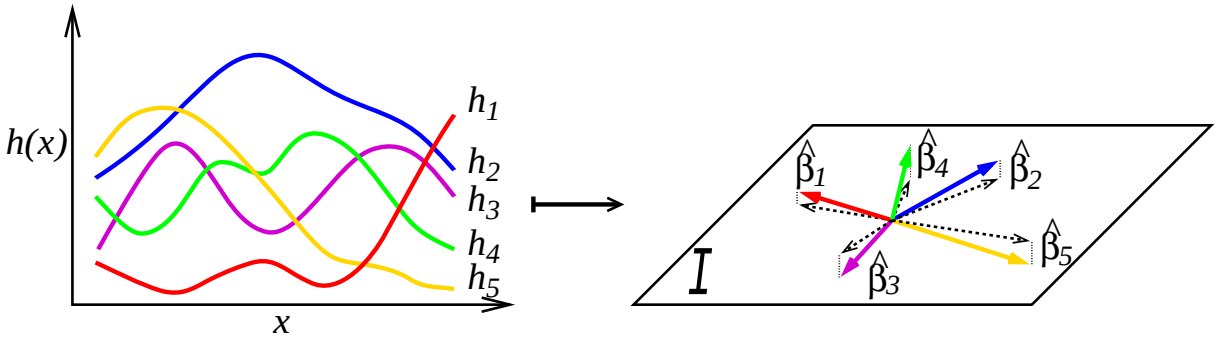


Figure 2: The principle underlying NGCA: from a family of real functions h, a family of vectors $\widehat{\boldsymbol{\beta}}$ belonging to the target space up to small estimation error is computed.

conditions the following vector belongs to the target space $\mathcal{I}$*:*

$$\boldsymbol{\beta}(h) = \Sigma^{-1}\mathbb{E}_{\boldsymbol{X}}\left[\boldsymbol{X}\, h(\boldsymbol{X})\right] - \mathbb{E}_{\boldsymbol{X}}\left[\nabla h(\boldsymbol{X})\right], \tag{3}$$

where $\Sigma = \mathbb{E}_{\boldsymbol{X}}\left[\boldsymbol{X}\boldsymbol{X}^\top\right]$.

The vector $\boldsymbol{\beta}$ defined by Eq.(3) plays the central role in NGCA. This is very close in spirit to Stein's identity (Stein, 1981) which claims that $\boldsymbol{\beta}(h) = \mathbf{0}$ for any function h, if and only if $\boldsymbol{X}$ is a Gaussian random vector. On the other hand, when the data $\boldsymbol{X}$ contains a non-Gaussian subspace as assumed, then the vector $\boldsymbol{\beta}(h)$ provides information about this subspace.

Since the definition of $\boldsymbol{\beta}$ contains expectations with respect to the unknown density $p(\boldsymbol{x})$, it must be estimated, for instance by replacing expectations with their empirical counterparts

$$\widehat{\boldsymbol{\beta}}(h) = \widehat{\Sigma}^{-1}\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{x}_i h(\boldsymbol{x}_i) - \frac{1}{n}\sum_{i=1}^{n}\nabla h(\boldsymbol{x}_i),$$

where $\widehat{\Sigma} := \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{x}_i\boldsymbol{x}_i^\top$. When the sample size n is large, the estimator $\widehat{\boldsymbol{\beta}}(h)$ is a good approximation of the true vector $\boldsymbol{\beta}(h)$ for any smooth non-linear function h. If we take a sufficiently large number of different functions $\{h_k\}$, we can expect that their corresponding vectors $\{\boldsymbol{\beta}(h_k)\}$ span the entire subspace $\mathcal{I}$ to be estimated. We can then obtain a good approximation $\widehat{\mathcal{I}}$ of $\mathcal{I}$ from the family of estimated vectors $\{\widehat{\boldsymbol{\beta}}(h_k)\}$, for example by applying PCA to this family (see Fig. 2). Note that the approximation error of $\widehat{\boldsymbol{\beta}}(h)$ is bounded *uniformly* for exponentially many $\widehat{\boldsymbol{\beta}}(h)$ and the error tends to zero asymptotically (see Theorem 1 in Section 2.5 for detail).

The outline of the general NGCA procedure is summarized as follows. We first perform a "whitening" transformation as preprocessing, because it is preferable both in the theoretical and practical senses (see Blanchard et al., 2006).

1. Apply "whitening" to the data $\{\boldsymbol{x}_i\}$, resulting in the whitened data $\{\widehat{\boldsymbol{y}}_i\}$ having identity (empirical) covariance. Formally, $\widehat{\boldsymbol{y}}_i = \widehat{\Sigma}^{-1/2}\boldsymbol{x}_i$ is the preprocessed data, where $\widehat{\Sigma} := \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{x}_i\boldsymbol{x}_i^\top$.

2. Consider a family of smooth functions $\{h_k\}$. Compute the family of vectors $\{\widehat{\boldsymbol{\beta}}_k\}$ given by
$$\widehat{\boldsymbol{\beta}}_k := \widehat{\boldsymbol{\beta}}(h_k) = \frac{1}{n}\sum_{i=1}^{n}\{\widehat{\boldsymbol{y}}_i h_k(\widehat{\boldsymbol{y}}_i) - \nabla h_k(\widehat{\boldsymbol{y}}_i)\}. \tag{4}$$

3. Apply PCA to the family $\{\widehat{\boldsymbol{\beta}}_k\}$ to obtain m principal directions.

4. Pull back the obtained m-dimensional subspace into the original (non-whitened) data space.

2.3 Implementation Issues

Although the principle of NGCA is very simple, there are two implementation issues. At first, since the mapping $h \mapsto \boldsymbol{\beta}(h)$ is linear, we need an appropriate renormalization of h or $\boldsymbol{\beta}(h)$, otherwise it would be meaningless to apply PCA to the family $\{\boldsymbol{\beta}_k\}$ computed using various functions $\{h_k\}$. In our previous paper (Blanchard et al., 2006), we proposed renormalizing $\widehat{\boldsymbol{\beta}}(h)$ by the trace of the variance $\mathrm{Var}\{\widehat{\boldsymbol{\beta}}(h)\}$. Under this condition the norm of each vector is proportional to its signal-to-noise ratio, so that longer vectors are more informative, while vectors with a too small norm are uninformative and can be discarded (Blanchard et al., 2006).

As we will explain in Section 2.5, the theoretical results guarantee that convergence occurs (as the number of sample points n grows to infinity) for any family of smooth functions $\{h_k\}_{k=1}^{L}$ with mild regularity conditions. However, in practice, it is important to find out those functions which provide most information on the index space $\mathcal{I}$. This will make the estimator $\widehat{\mathcal{I}}$ more accurate, and this point is crucial because there exist many uninformative functions. As stated above, the amount of information brought forth by a fixed function h_k is measured through the norm of the (renormalized) associated vector $\boldsymbol{\beta}_k$.

To sum up, we need to combine extra steps with the plain NGCA algorithm: (i) normalization, (ii) thresholding and (iii) iterative search for informative functions.

2.4 The Previous NGCA Algorithm: Multi-index Projection Pursuit

Previously we restricted our attention to functions of the form (Blanchard et al., 2006)
$$h_{f,\boldsymbol{\omega}}(\boldsymbol{y}) = f(\langle \boldsymbol{\omega}, \boldsymbol{y} \rangle), \tag{5}$$
where $\boldsymbol{\omega} \in \mathbb{R}^d$, $\|\boldsymbol{\omega}\| = 1$, and f belongs to a finite family $\mathcal{F}$ of smooth real functions of a real variable. That is, each function h depends only on a single direction of the multi-dimensional data determined by $\boldsymbol{\omega}$. To find good candidates $\boldsymbol{\omega}_f$, for a fixed f, we employed a well-known PP algorithm, FastICA (Hyvärinen, 1999), as a heuristic. In the FastICA update the vector $\widehat{\boldsymbol{\beta}}$ after normalization becomes the next candidate for the directional

parameter $\boldsymbol{\omega}$. The algorithmic detail can be found in the pseudocode in Table 1. We remark that FastICA, as a standalone procedure, requires to fix the "index function" f beforehand. On the other hand, the NGCA algorithm in Blanchard et al. (2006) employs a possibly large spectrum of arbitrary index functions f and *combines* the results of all single PP methods in the end. This is why we call the algorithm *Multi-index PP* in order to distinguish the new algorithm proposed in Section 3. The following functions are used as the indices f in the implementation:

$$\begin{aligned} f_\sigma^{(1)}(z) &= z^3 \exp(-z^2/2\sigma^2), \quad \sigma^2 \in [0.5, 5] \qquad \text{(Gauss-Pow3)}, \\ f_b^{(2)}(z) &= \tanh(bz), \quad b \in [0, 5] \qquad \text{(Hyperbolic Tangent)}, \\ f_a^{(3)}(z) &= \sin(az), \cos(az), \quad a \in [0, 4] \qquad \text{(Fourier)}. \end{aligned}$$

Each of the ranges of the extra parameters was divided into 1000 equispaced values, thus yielding a family (f_k) of size 4000 (Fourier functions count twice because of the sine and cosine parts). In the thresholding step, we remove uninformative vectors $\widehat{\boldsymbol{\beta}}_k$ with norm smaller than a threshold ε after normalization by their estimated variances $\widehat{\mathrm{Var}}(\widehat{\boldsymbol{\beta}}_k)$ (for details, see the pseudocode in Table 1). Some preliminary calibration on purely Gaussian data suggested to take $\varepsilon = 1.5$ as the threshold. Finally we fixed the number of FastICA iterations $T_{\max} = 10$.

2.5 Probabilistic Error Bounds

A theoretical validity of the NGCA method is guaranteed by the following theorem, which basically tells that the estimation error, i.e. the distance between the true non-Gaussian index space and the estimated vector converges to 0 with high probability as the sample size n goes to infinity. Together with the assumption that the set of betas spans the index space $\mathcal{I}$, we come to the conclusion that the estimator $\widehat{\mathcal{I}}$ converges to $\mathcal{I}$. The details and the proof of the probabilistic error bounds can be found in Blanchard et al. (2006).

Table 1:

PSEUDOCODE FOR THE MULTI-INDEX PP ALGORITHM

Input: Data points $(\boldsymbol{x}_i) \in \mathbb{R}^d$, dimension m of target subspace.
Parameters: Number $T_{\max}$ of FastICA iterations; threshold ε;
family of real functions (f_k).

Whitening.

The data $\boldsymbol{x}_i$ is recentered by subtracting the empirical mean.
Let $\widehat{\Sigma}$ denote the empirical covariance matrix of the data sample $(\boldsymbol{x}_i)$;
put $\widehat{\boldsymbol{y}}_i = \widehat{\Sigma}^{-\frac{1}{2}} \boldsymbol{x}_i$ the empirically whitened data.

Main Procedure.

Loop on $k = 1, \ldots, L$:
Draw $\boldsymbol{\omega}_0$ at random on the unit sphere of $\mathbb{R}^d$.
Loop on $t = 1, \ldots, T_{\max}$: *[FastICA loop]*

$$\text{Put } \widehat{\boldsymbol{\beta}}_t \leftarrow \frac{1}{n} \sum_{i=1}^{n} \left(\widehat{\boldsymbol{y}}_i f_k(\langle \boldsymbol{\omega}_{t-1}, \widehat{\boldsymbol{y}}_i \rangle) - f_k'(\langle \boldsymbol{\omega}_{t-1}, \widehat{\boldsymbol{y}}_i \rangle) \boldsymbol{\omega}_{t-1} \right).$$

$$\text{Put } \boldsymbol{\omega}_t \leftarrow \widehat{\boldsymbol{\beta}}_t / \|\widehat{\boldsymbol{\beta}}_t\|.$$

End Loop on t
Let N_k be the trace of the empirical covariance matrix of $\widehat{\boldsymbol{\beta}}_{T_{\max}}$:

$$N_k = \frac{1}{n} \sum_{i=1}^{n} \left\| \widehat{\boldsymbol{y}}_i f_k(\langle \boldsymbol{\omega}_{T_{\max}-1}, \widehat{\boldsymbol{y}}_i \rangle) - f_k'(\langle \boldsymbol{\omega}_{T_{\max}-1}, \widehat{\boldsymbol{y}}_i \rangle) \boldsymbol{\omega}_{T_{\max}-1} \right\|^2 - \left\| \widehat{\boldsymbol{\beta}}_{T_{\max}} \right\|^2.$$

Store $v^{(k)} \leftarrow \widehat{\boldsymbol{\beta}}_{T_{\max}} * \sqrt{n/N_k}$. *[Normalization]*
End Loop on k

Thresholding.

From the family $\boldsymbol{v}^{(k)}$, throw away vectors having norm smaller than threshold ε.

PCA step.

Perform PCA on the set of remaining $\boldsymbol{v}^{(k)}$.
Let V_m be the space spanned by the first m principal directions.

Pull back in original space.

Output: $W_m = \widehat{\Sigma}^{-\frac{1}{2}} V_m$.

Theorem 1 *Assume:*

(i) There exist $\lambda_0 > 0$, $a_0 > 0$ *such that* $\mathbb{E}_{\boldsymbol{X}} \left[\exp\left(\lambda_0 \|\boldsymbol{X}\|^2\right) \right] \leq a_0 < \infty$;
(ii) The matrix $\Sigma = \mathbb{E}_{\boldsymbol{X}} \left[\boldsymbol{X} \boldsymbol{X}^\top \right]$ *is such that* $\|\Sigma^{-1}\|_{\text{op}} < K^2$;
(iii) $\sup_{k,\boldsymbol{y}} \max(\|\nabla h_k(\boldsymbol{y})\|, \|\boldsymbol{y} h_k(\boldsymbol{y})\|) < B$;
(iv) The function $\widetilde{h}_k(\boldsymbol{y}) := \boldsymbol{y} h_k(\boldsymbol{y}) - \nabla h_k(\boldsymbol{y})$ *is Lipschitz with constant* M.

Then, there exists an integer n_0 *such that for any* $n > n_0$, *with probability* $1 - \frac{4}{n} - 4\delta$ *the*

following bounds hold true simultaneously for all $k \in \{1, \ldots, L\}$

$$\begin{aligned}\operatorname{dist}(\widehat{\Sigma}^{-1/2}\widehat{\boldsymbol{\beta}}(h_k), \mathcal{I}) \;\leq\; & C_1\sqrt{\frac{d\log n}{n}} + 2K\sqrt{\widehat{\sigma}^2(h_k)\frac{\log L\delta^{-1} + \log d}{n}} \\ & + C_2\frac{\log(nL\delta^{-1})\log(L\delta^{-1})}{n^{\frac{3}{4}}},\end{aligned}$$

where $\widehat{\Sigma} = \frac{1}{n}\sum_{i=1}^n \boldsymbol{x}_i\boldsymbol{x}_i^\top$, $\widehat{\boldsymbol{\beta}}(h_k) = \frac{1}{n}\sum_{i=1}^n \widetilde{h}_k(\widehat{\boldsymbol{y}}_i)$,
$\widehat{\sigma}^2(h_k) = \frac{1}{n}\sum_{i=1}^n \|\widetilde{h}_k(\widehat{\boldsymbol{y}}_i) - \widehat{\boldsymbol{\beta}}(h_k)\|^2$, C_1 *is a constant depending on parameters* $(\lambda_0, a_0, B, K, M)$ *only and* C_2 *on* $(d, \lambda_0, a_0, B, K, M)$.

The above bound leads to a uniform convergence over the whole set of functions with a rate of order $\mathcal{O}(\sqrt{d\log n/n} + \sqrt{\log L/n})$. Therefore, taking, e.g., $L = O(n^d)$ we have insurance that global convergence holds.

Roughly speaking, the second term of the bound in Theorem 1 comes from the main procedure of NGCA (cf. Theorem 3 of Blanchard et al., 2006), while the first term is caused by the whitening preprocessing. Our normalization scheme aims at keeping the second term of the bound below a constant, because the coefficient C_1 in the first term is uncomputable.

3 New NGCA Algorithm with Radial Kernel Functions

In the iterative function selection of Multi-index PP, we do not directly optimize our criterion of informativeness, given by the length of the renormalized vector $\widehat{\boldsymbol{\beta}}$:

$$\frac{\|\widehat{\boldsymbol{\beta}}(h_{f,\boldsymbol{\omega}})\|}{\sqrt{N(h_{f,\boldsymbol{\omega}})}} \tag{6}$$

over the directional parameter $\boldsymbol{\omega}$, where

$$N(h_{f,\boldsymbol{\omega}}) := \frac{1}{n}\sum_{i=1}^n \|\widehat{\boldsymbol{y}}_i f_k(\langle\boldsymbol{\omega}, \widehat{\boldsymbol{y}}_i\rangle) - f_k'(\langle\boldsymbol{\omega}, \widehat{\boldsymbol{y}}_i\rangle)\boldsymbol{\omega}\|^2 - \left\|\widehat{\boldsymbol{\beta}}(h_{f,\boldsymbol{\omega}})\right\|^2 \tag{7}$$

is n times the trace of the estimated variance $\widehat{\operatorname{Var}}[\widehat{\boldsymbol{\beta}}(h_{f,\boldsymbol{\omega}})]$. Instead, the FastICA loop is used as a heuristic, which makes the algorithm much simpler. This motivated us to have a more direct approach: finding a function h that maximizes the above informativeness measure Eq.(6).

3.1 Radial Kernel Functions

In the following, we focus on the class of functions spanned by radial kernel functions

$$k_{\sigma^2,M}(\boldsymbol{y},\boldsymbol{y}') = \kappa\left(\frac{1}{2\sigma^2}(\boldsymbol{y}-\boldsymbol{y}')^\top M(\boldsymbol{y}-\boldsymbol{y}')\right), \tag{8}$$

where κ is a non-negative smooth function and M is a non-negative definite matrix which determines the shape of ellipsoid. This is a generalization of radial basis functions which have been applied in the field of neural network and machine learning (see e.g. Moody and Darken, 1989; Bishop, 1995; Schölkopf and Smola, 2001). The reason why the metric M is contained will be explained later. In the following, we mainly use Gaussian kernels

$$k_{\sigma^2,M}(\boldsymbol{y},\boldsymbol{y}') = \exp\left\{-\frac{1}{2\sigma^2}(\boldsymbol{y}-\boldsymbol{y}')^\top M(\boldsymbol{y}-\boldsymbol{y}')\right\},$$

but other basis function families are also allowed. To sum up, we will consider functions which are represented as a linear combination of radial kernel functions centered at each sample. This bears a similarity with kernel methods in machine learning or statistics (cf. Schölkopf and Smola, 2001; Müller et al., 2001).

$$h_{\sigma^2,M,\boldsymbol{a}}(\boldsymbol{y}) = \sum_{i=1}^{n} a_i k_{\sigma^2,M}(\boldsymbol{y},\widehat{\boldsymbol{y}}_i) \tag{9}$$

This family of functions has three parameters: the scale parameter σ^2, the metric M and the weight $\boldsymbol{a} = (a_i)$. The scale parameter σ^2 can be included in M, but we separate them for convenience; more specifically, we use several different scale values for the same metric M at the same time. In the following implementation, we take 10 different values between 0.01 and 60 which are equally spaced in log scale. Similarly to the role of extra parameters in the case of multi-index projection pursuit (e.g., the frequency parameters for trigonometric functions), this enables us to combine information coming from various resolutions.

In the following, we explain how $\boldsymbol{a}$ and M are determined.

3.2 Optimization of Weight Vector

The weight vector $\boldsymbol{a}$ is obtained from the maximization of the informativeness criterion

$$\frac{\|\widehat{\boldsymbol{\beta}}(h_{\sigma^2,M,\boldsymbol{a}})\|^2}{N(h_{\sigma^2,M,\boldsymbol{a}})}, \tag{10}$$

where $N(h_{\sigma^2,M,\boldsymbol{a}}) = n\,\mathrm{tr}\left(\widehat{\mathrm{Var}}[\widehat{\boldsymbol{\beta}}(h_{\sigma^2,M,\boldsymbol{a}})]\right)$ is the normalization factor. Due to the fact that the functions $h_{\sigma^2,M,\boldsymbol{a}}$ are linear in the parameter $\boldsymbol{a}$, both the numerator and denominator of Eq.(10) are expressed as quadratic forms of the vector $\boldsymbol{a}$. Therefore, the optimization of Eq.(10) can be solved by a generalized eigenvalue problem as we will show

in the following. We underline that this characterization of the optimal coefficient $\boldsymbol{a}$ of the linear combination h defined by Eq.(9) is more generally true in the case where h is expressed as a linear combination of some given basis functions; i.e. it is not specific to the radial kernels $k_{\sigma^2,M}$. Below, we derive the explicit analytic formulation for this particular case.

Let $K = (k_{\sigma^2,M}(\widehat{\boldsymbol{y}}_i, \widehat{\boldsymbol{y}}_j))$ be the Gram matrix of the kernel and $\widehat{Y} = (\widehat{\boldsymbol{y}}_1, \ldots, \widehat{\boldsymbol{y}}_n)$ be the matrix representation of all whitened data. From Eq.(9), the r-th component of the vector-valued function $\widetilde{h}_{\sigma^2,M,\boldsymbol{a}}(\widehat{\boldsymbol{y}}_i) := \widehat{\boldsymbol{y}}_i \; h_{\sigma^2,M,\boldsymbol{a}}(\widehat{\boldsymbol{y}}_i) - \nabla h_{\sigma^2,M,\boldsymbol{a}}(\widehat{\boldsymbol{y}}_i)$ can be expressed as

$$\{\widetilde{h}_{\sigma^2,M,\boldsymbol{a}}(\widehat{\boldsymbol{y}}_i)\}_r = \boldsymbol{e}_r^\top \widehat{\boldsymbol{y}}_i \; (K\boldsymbol{a})_i - (\partial_r K \boldsymbol{a})_i,$$

where $\boldsymbol{e}_r$ is the r-th basis vector of $\mathbb{R}^d$ and ∂_r denotes the partial derivative w.r.t. the r-th component. Specifically, the (i,j)-th component of the matrix $\partial_r K$ becomes

$$\partial_r K_{ij} = \frac{1}{\sigma^2}\left\{(M\widehat{\boldsymbol{y}}_i)_r - (M\widehat{\boldsymbol{y}}_j)_r\right\} \; \kappa'\left(\frac{1}{2\sigma^2}(\widehat{\boldsymbol{y}}_i - \widehat{\boldsymbol{y}}_j)^\top M(\widehat{\boldsymbol{y}}_i - \widehat{\boldsymbol{y}}_j)\right),$$

when we use the radial kernel functions (8). Then the norm of $\widehat{\boldsymbol{\beta}}(h_{\sigma^2,M,\boldsymbol{a}})$ and $n\,\widehat{\mathbb{E}}\left[\|\widehat{\boldsymbol{\beta}}(h_{\sigma^2,M,\boldsymbol{a}})\|^2\right]$ are expressed by the following quadratic forms.

$$\begin{aligned}
\|\widehat{\boldsymbol{\beta}}(h_{\sigma^2,M,\boldsymbol{a}})\|^2 &= \sum_{r=1}^d \left[\frac{1}{n}\sum_{i=1}^n \{\widetilde{h}_{\sigma^2,M,\boldsymbol{a}}(\widehat{\boldsymbol{y}}_i)\}_r\right]^2 \\
&= \sum_{r=1}^d \frac{1}{n^2}\left[\boldsymbol{e}_r^\top \widehat{Y}\; K\boldsymbol{a} - \mathbf{1}_n^\top \partial_r K \boldsymbol{a}\right]^2 \\
&= \boldsymbol{a}^\top \left\{\frac{1}{n^2}\sum_{r=1}^d \left(\boldsymbol{e}_r^\top \widehat{Y} K - \mathbf{1}_n^\top \partial_r K\right)^\top \left(\boldsymbol{e}_r^\top \widehat{Y} K - \mathbf{1}_n^\top \partial_r K\right)\right\} \boldsymbol{a},
\end{aligned}$$

$$\begin{aligned}
n\,\widehat{\mathbb{E}}\left[\|\widehat{\boldsymbol{\beta}}(h_{\sigma^2,M,\boldsymbol{a}})\|^2\right] &= \frac{1}{n}\sum_{i=1}^n \sum_{r=1}^d \{\widetilde{h}_{\sigma^2,M,\boldsymbol{a}}(\widehat{\boldsymbol{y}}_i)\}_r^2 \\
&= \boldsymbol{a}^\top \left[\frac{1}{n}\sum_{r=1}^d \left\{\mathrm{diag}(\boldsymbol{e}_r^\top \widehat{Y}) K - \partial_r K\right\}^\top \left\{\mathrm{diag}(\boldsymbol{e}_r^\top \widehat{Y}) K - \partial_r K\right\}\right] \boldsymbol{a}.
\end{aligned}$$

Let us define the following matrices

$$F := \frac{1}{n^2}\sum_{r=1}^d \left(\boldsymbol{e}_r^\top \widehat{Y} K - \mathbf{1}_n^\top \partial_r K\right)^\top \left(\boldsymbol{e}_r^\top \widehat{Y} K - \mathbf{1}_n^\top \partial_r K\right), \tag{11}$$

$$G := \frac{1}{n}\sum_{r=1}^d \left\{\mathrm{diag}(\boldsymbol{e}_r^\top \widehat{Y}) K - \partial_r K\right\}^\top \left\{\mathrm{diag}(\boldsymbol{e}_r^\top \widehat{Y}) K - \partial_r K\right\} - F. \tag{12}$$

Then the informativeness criterion (10) can be represented as a Rayleigh coefficient

$$\frac{\|\widehat{\boldsymbol{\beta}}(h_{\sigma^2,M,\boldsymbol{a}})\|^2}{N(h_{\sigma^2,M,\boldsymbol{a}})} = \frac{\boldsymbol{a}^\top F \boldsymbol{a}}{\boldsymbol{a}^\top G \boldsymbol{a}}. \tag{13}$$

Hence, the maximizer of the criterion is given by the generalized eigenvector associated to the largest generalized eigenvalue of the following generalized eigenvalue problem

$$F\boldsymbol{a} = \lambda G\boldsymbol{a}. \tag{14}$$

In our implementation, we add a regularization term $0.01I_n$ to the matrix G in order to avoid the undesired situation that the denominator of Eq.(13) is close to zero. Because functions with the maximizer $\boldsymbol{a}_1$ may correspond only to one direction in the non-Gaussian index space $\mathcal{I}$ for all scales $\sigma_1^2, \ldots, \sigma_L^2$, we should take at least m eigenvectors $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m$ of Eq.(14). We also remark that the weight $\boldsymbol{a}$ is usually normalized such that $\boldsymbol{a}^\top G\boldsymbol{a} = 1$ by eigen-solvers; then further normalization of $\boldsymbol{\beta}$ is not necessary. Note that the n-dimensional matrix F has rank d. If we take this into account, the above eigenvalue problem may be solved more efficiently. Computational cost could be further reduced, e.g., by using a subset of samples.

3.3 Update of Metric

Finally, the non-negative definite matrix M in the radial kernel functions $k_{\sigma^2,M}$ is iteratively updated several times from the identity matrix $M_0 = I_d$. We change the metric M based on the current estimator $\widehat{\mathcal{I}}$ of the index space such that the functions $h_{\sigma^2,M,\boldsymbol{a}}$ become less sensitive (or in other words the radial kernels have larger radius) in Gaussian directions. This extra step improves the accuracy substantially compared to the algorithm with just spherical bases ($M = I_d$). Suppose we get the set of vectors $\left\{\boldsymbol{\beta}_k^{(t)}\right\}$ at the t-th step. Then the simplest rule would be using the covariance matrix of $\left\{\boldsymbol{\beta}_k^{(t)}\right\}$, that is,

$$M_t \propto \sum_k \boldsymbol{\beta}_k^{(t)} \left(\boldsymbol{\beta}_k^{(t)}\right)^\top, \tag{15}$$

where M_t is scaled so that its trace remains equal to d. However, this procedure might miss non-Gaussian components that are relatively weak, if the data also contains some strongly non-Gaussian directions. To alleviate this issue, we propose equalizing the weights in the first m eigen directions, that is,

$$M_t \propto \overline{\mu} \sum_{i=1}^{m} \boldsymbol{u}_i \boldsymbol{u}_i^\top + \sum_{i=m+1}^{d} \mu_i \boldsymbol{u}_i \boldsymbol{u}_i^\top, \tag{16}$$

where μ_i and $\boldsymbol{u}_i$ are the i-th eigenvalue and vector of the matrix $\sum_k \boldsymbol{\beta}_k^{(t)} \left(\boldsymbol{\beta}_k^{(t)}\right)^\top$ in descending order and $\overline{\mu} = m^{-1} \sum_{i=1}^{m} \mu_i$. For illustration purposes, we plot the improvement

Table 2:

PSEUDOCODE FOR
ITERATIVE METRIC ADAPTATION FOR RADIAL KERNEL FUNCTIONS (IMAK)

Main Procedure.
Put $M_0 = I_d$.
Loop on $t = 1, \ldots, T_{\max}$:
Loop on $l = 1, \ldots, L$:
Calculate the Gram matrix $K = (k_{\sigma_l^2, M_t}(\widehat{\boldsymbol{y}}_i, \widehat{\boldsymbol{y}}_j))$ and its derivatives $\partial_r K$ $(r = 1, \ldots, d)$.
Calculate the matrices F and G by Eq.(11) and Eq.(12).
Get the m largest eigen vectors $\boldsymbol{a}_{l1}, \ldots, \boldsymbol{a}_{lm}$ of $F\boldsymbol{a} = \lambda G\boldsymbol{a}$.
Calculate $\widehat{\boldsymbol{\beta}}_{lj} = n^{-1}\left\{\widehat{Y}K - \mathbf{1}_n^\top \left((\partial_1 K)^\top, \ldots, (\partial_d K)^\top\right)^\top\right\}\boldsymbol{a}_{lj}$ for $j = 1, \ldots, m$.
End of Loop on l.
Put $M_t \propto \sum_{l=1}^{L}\sum_{j=1}^{m} \widehat{\boldsymbol{\beta}}_{lj}\widehat{\boldsymbol{\beta}}_{lj}^\top$ with $\mathrm{tr}(M_t) = d$ (or as in Eq.(16)).
End of Loop on t.

of the estimation error $\mathcal{E}$ by the metric adaptation with the data set (A) in Fig. 3 (see Section 4 for details). Indeed, the error decreases drastically after a few iterations. At the same time, the metric M converges to an ideal (for this dataset) matrix which has non-zero elements only in the first 2×2 submatrix. This makes the functions (h_l) not affected by the Gaussian components and generates more accurate vectors $(\boldsymbol{\beta}_l)$.

We summarized the main part of the new NGCA algorithm called *Iterative Metric Adaptation for radial Kernel functions (IMAK)* in Table 2. Since the "whitening" preprocessing, the PCA step and the final operation for pulling back the result in the original space are the same as Multi-index PP in Table 1, we omit them. Thresholding is not necessary in our new IMAK procedure. The iteration number $T_{\max}$ of the metric update is 10 in the numerical experiments, but when there existed almost no change, we stopped the iterations earlier for computational efficiency.

4 Numerical Experiments

We performed numerical experiments using various synthetic data sets. We report exemplary results using 4 data sets. Each data set includes 1000 samples $\boldsymbol{x}_i = [s_{1,i}, s_{2,i}, n_{3,i}, \ldots, n_{10,i}]^\top$ $(i = 1, \ldots, 1000)$ in 10 dimensions, which consists of 8-dimensional independent standard Gaussian $[n_{3,i}, \ldots, n_{10,i}]^\top$ and 2-dimensional non-Gaussian components $\boldsymbol{s}_i = [s_{1,i}, s_{2,i}]^\top$ as follows:
(A) Simple Gaussian Mixture: 2-dimensional independent bimodal Gaussian mixtures;
(B) Dependent super-Gaussian: 2-dimensional density is proportional to $\exp(-\|\boldsymbol{x}\|)$;
(C) Dependent sub-Gaussian: 2-dimensional uniform on the unit circle;
(D) Dependent super- and sub-Gaussian: 1-dimensional Laplacian with density pro-

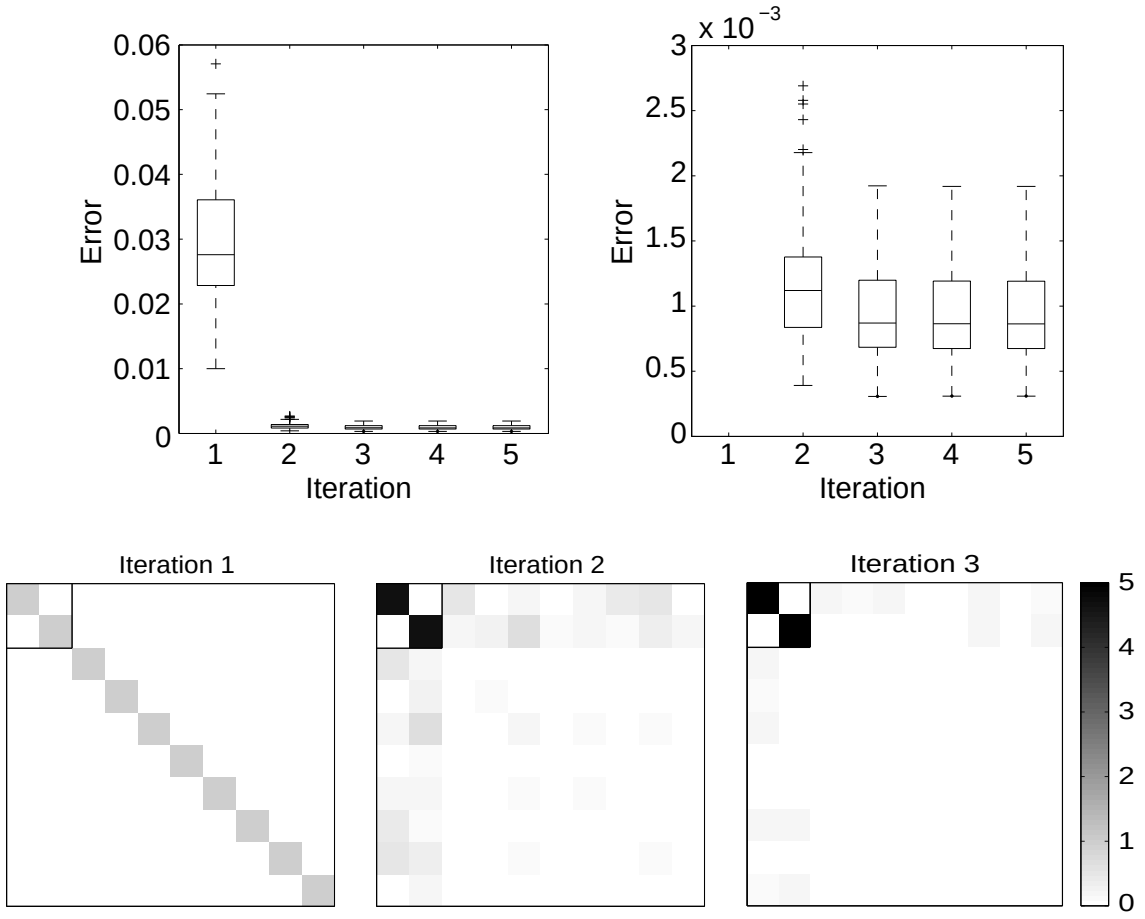
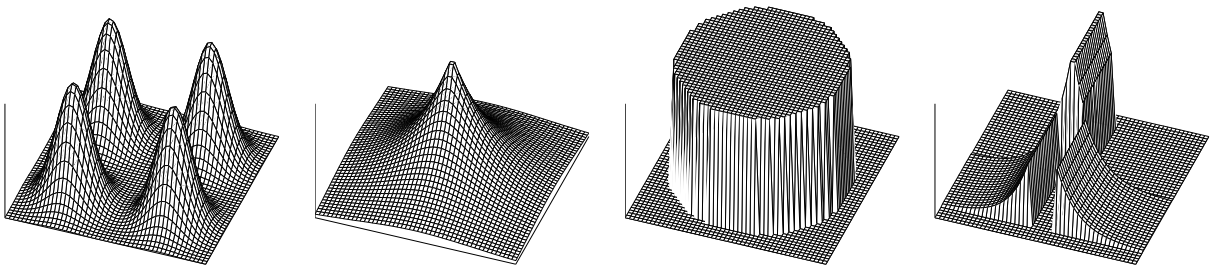


Figure 3: Effectiveness of Iterative Metric Adaptation. The upper row shows that the estimation error $\mathcal{E}$ defined in Section 4 decreases drastically after a few iterations (left: coarse scale, right: fine scale). In the lower row, we plot the absolute values of the components of the metrics at first, second and third iterations (i.e.$M_0 = I_d$, M_1 and M_2, respectively). We used the data set (A), where the first two coordinates correspond to the non-Gaussian index space and the others are Gaussian noise (see Section 4 for details). Hence, if the metric M has non-zero values only in the first 2×2 submatrix, the function h becomes independent of the Gaussian components. For this data set, we can see that the metric M converges to the ideal matrix after only three iterations.

portional to $\exp(-|x_{Lap}|)$ and 1-dimensional dependent uniform $U(c, c+1)$, where $c = 0$ for $|x_{Lap}| \leq \log 2$ and $c = -1$ otherwise.
This is a simple case of our setting $\boldsymbol{x}_i = A\boldsymbol{s}_i + \boldsymbol{n}_i$ with $A = [I_2\ 0_{2\times 8}]^\top$ and $\boldsymbol{n}_i = [0, 0, n_{3,i}, \ldots, n_{10,i}]^\top$. We compare the new NGCA algorithm (denoted by 'IMAK') against Multi-index PP ('PPMI') and standalone FastICA with two different index functions ('PP(pow3)' and 'PP(tanh)', respectively). Because the single index PPs tend to get trapped into local optima of the index function that it optimizes, we restarted it 10 times with random starting points and took the subspace obtaining the best index value. How-



(A) (B) (C) (D)

Figure 4: Densities of the non-Gaussian components.

ever, even when it is restarted 10 times, PP with 'pow3' index still gets caught in local optima in a small percentage of cases. Fig. 5 shows boxplots, over 100 samples, of the error criterion

$$\mathcal{E}(\widehat{\mathcal{I}}, \mathcal{I}) = m^{-1} \sum_{i=1}^{m} \|(I_d - \Pi_{\mathcal{I}})\widehat{\boldsymbol{v}}_i\|^2,$$

where $\{\widehat{\boldsymbol{v}}_i\}_{i=1}^m$ is an orthonormal basis of $\widehat{\mathcal{I}}$, I_d is the identity matrix, and $\Pi_{\mathcal{I}}$ denotes the orthogonal projection on $\mathcal{I}$. In datasets (A) and (B), our algorithm appears to be essentially on par with Multi-index PP and the best FastICA method, while in dataset (C), IMAK is better than the others. As expected the best index for FastICA is data-dependent: the 'tanh' index is more suited to the super-Gaussian data (B) while the 'pow3' index works best with the sub-Gaussian data (C). We note that we reproduced the simulation in Blanchard et al. (2006) on different realizations of the same datasets. Concerning data distribution (C), although the error distributions for PP(pow3) are basically consistent, the boxplot in Blanchard et al. (2006) looked worse than that on Fig. 5. This is because the proportion of large errors (caused by local optima) was slightly beyond 25%, which is not the case here. Finally, the advantage of the implicit index adaptation feature of NGCA can be clearly observed in the data set (D), which includes both sub- and super-Gaussian components. In this case neither of the two FastICA index functions taken alone does well. Multi-index PP gives significantly lower error than either FastICA and the new algorithm IMAK provides further improvement.

5 Historical Remarks and Related Works

In this section, we discuss other work related to our research.

At the early stage of PP development, Huber (1985) already suggested the negative Shannon entropy with an non-parametric density estimator as a projection index. There were also a number of flexible projection indices in the 80s and early 90s (see Nason, 1992, for details). Nason (1992) also proposed a non-parametric projection index based on multimodality of probability densities.

After the late 90s projection pursuit attracted researchers working on blind source sep-

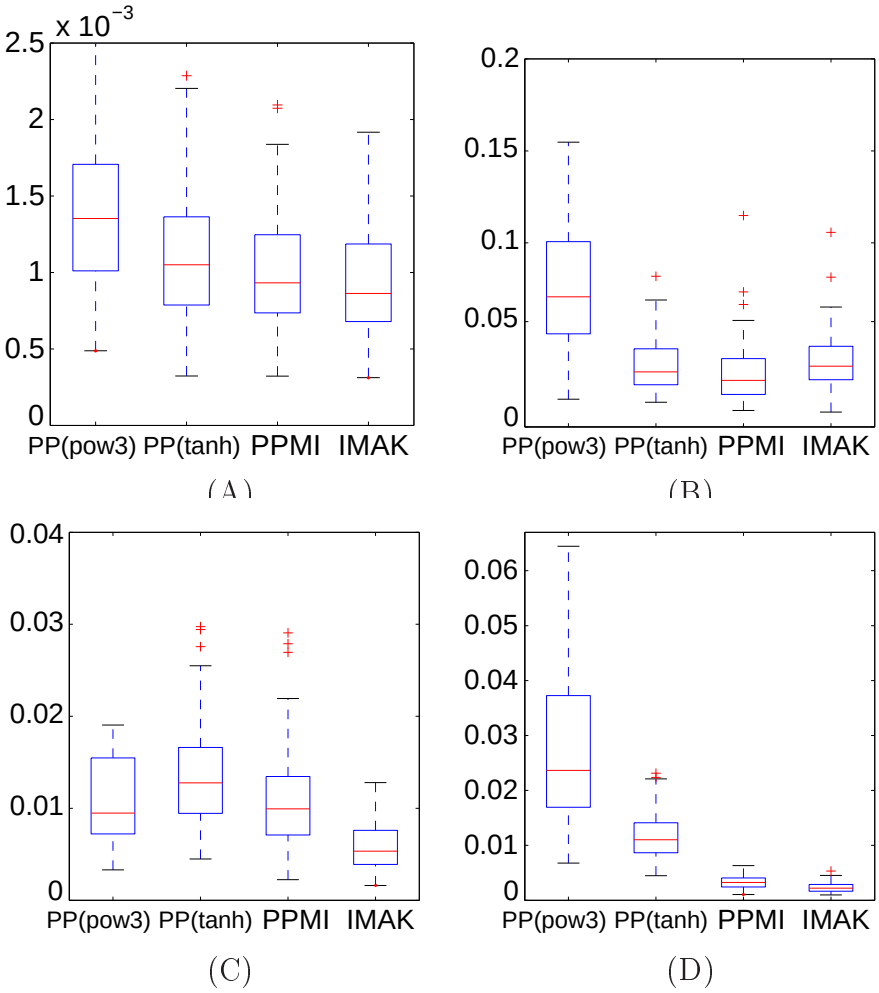


Figure 5: Boxplots of the error criterion $\mathcal{E}(\widehat{\mathcal{I}},\mathcal{I})$ over 100 training samples of size 1000.

aration and has been further developed under the name of deflational ICA (e.g. Hyvärinen, 1999; Hyvärinen et al., 2001). Immediately, people became aware of the problems with non-Gaussianity indices, that some indices work well for sub-Gaussian sources, while others perform well with super-Gaussian signals. Lee et al. (1999) proposed the extended infomax which switches between the two kinds of non-Gaussian indices based on kurtosis of each extracted component.

Furthermore, Hyvärinen et al. (2001) proved that the negative Shannon entropy is the optimal index in the sense of asymptotic variance (Theorem 14.2). Since then, more flexible and adaptive non-Gaussianity indices (or non-linearity for symmetric ICA methods such as infomax) have been proposed. Some assume wider parametric families for source distributions (e.g. Eriksson et al., 2000), while others employ various kinds of non-parametric estimators such as kernel density estimators or Gaussian mixture models (Boscolo et al., 2004; Vassis and Motomura, 2001; Attias, 1999), maximum likelihood with a smoothness penalty (Hastie and Tibshirani, 2003), order statistics spacings (e.g.

RADICAL, Learned-Miller and Fisher, 2003), and characteristic functions (e.g. Eriksson et al., 2000).

The most elegant algorithm among non-parametric ICA is KernelICA (Bach and Jordan, 2002; Gretton et al., 2005) which utilizes kernel mutual information or kernel generalized covariance with functions in a reproducing kernel Hilbert space. Rigorous theoretical analysis of non-parametric ICA was done in Chen and Bickel (2006). For developing IMAK—the new version of NGCA, we drew inspiration from these papers.

The differences between our approach and PP/ICA with non-parametric estimators, especially the kernel density estimator are summarized as follows.

1. We are not optimizing a particular index. Rather, based on Proposition 1, we combine information from multiple "indices" at the same time, while other methods use single adaptive indices only. Thus, for example, we do not need any window width selection.

2. The other approaches are generally based on 1-dimensional indices which are then optimized via some search procedure. By contrast our non-Gaussianity function starts from the entire space and "shrinks" towards the index space through the adaptation of the metric.

6 Conclusion

In this paper we proposed an alternative realization of the NGCA procedure for constructing a linear projection to separate an uninteresting, multivariate Gaussian 'noise' subspace of possibly large amplitude from the 'signal-of-interest' subspace. The new IMAK algorithm uses radial kernel functions and also iteratively updates the metric of the kernels. In general, PP methods need to pre-specify a projection index with which they search non-Gaussian components. By contrast, as was explained in Blanchard et al. (2006), NGCA can simultaneously deal with several families of nonlinear functions; moreover, also within a function family we are able to use an entire range of parameters (such as frequency for Fourier functions). This may be interpreted intuitively by saying that NGCA automatically chooses the functional indices that are the most relevant for analyzing the data at hand. Thus, NGCA provides higher flexibility with *a priori* less restricting assumptions on the data. Moreover, IMAK optimizes the informativeness criterion directly to obtain useful functions, while multi-index PP uses the FastICA updates of the directional parameters as heuristics. Thus, IMAK is theoretically better justified within the NGCA framework. Optimization in IMAK is done elegantly by solving a generalized eigenvalue problem. Numerically, we found comparable or superior performance to, e.g., FastICA in deflation mode and multi-index PP.

Future research will adapt the theory to simultaneously estimate the dimension of the non-Gaussian subspace. Experimentally testing the new IMAK algorithm in some application domains such as signal denoising (Sugiyama et al., 2006) is being carried out. Extending the proposed framework to non-linear projection scenarios (Roweis and Saul,

2000; Tenenbaum et al., 2000; Belkin and Niyogi, 2003; Harmeling et al., 2003) and to finding the most discriminative directions using labels are examples for which the current theory could be taken as a basis.

Acknowledgements
MK acknowledges Shinto Eguchi and Vladimir Spokoiny for valuable comments. This work was supported in part by the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778, MEXT, Grant-in-Aid for Young Scientists 17700142, Grand-in-Aid for Scientific Research (B) 1830057 and BMBF, CCNB grant 01GQ0415. This publication only reflects the authors' views.

Appendix

A From model (1) to formulation (2)

We rephrase briefly here the argument of Lemma 1 in Blanchard et al. (2006). We start from the model (1):

$$\boldsymbol{X} = A\boldsymbol{S} + \boldsymbol{N}\,,$$

where $\boldsymbol{S}$ and $\boldsymbol{N}$ are independent and $\boldsymbol{N}$ is Gaussian. We first assume that the covariance matrix of the Gaussian noise component $\boldsymbol{N}$ is identity. Let $E = \mathrm{Range}(A)$, and $F = E^{\perp}$ its orthogonal complement. Let T_E, resp. T_F denote the $m \times d$, resp $(d-m) \times d$ orthogonal projection matrices on some orthogonal basis of E, resp. F. Then, $\boldsymbol{X}_1 := T_E\boldsymbol{X} = T_E A\boldsymbol{S} + T_E\boldsymbol{N} \in \mathbb{R}^m$ and $\boldsymbol{X}_2 := T_F\boldsymbol{X} = T_F\boldsymbol{N} \in \mathbb{R}^{d-m}$, which are the coordinate vectors of $\boldsymbol{X}$ on the subspaces E and F, respectively, are mutually independent (since $T_F\boldsymbol{N}$ is independent both of $T_E A\boldsymbol{S}$ and of $T_E\boldsymbol{N}$).

Obviously, $\boldsymbol{X}_2$ has standard normal density in dimension $(d-m)$, $\phi_{d-m}(x_2)$. Now, let $h(x_1)$ denote the density of $\boldsymbol{X}_1 \in \mathbb{R}^m$. Then the density of $\boldsymbol{X}$ has the form

$$p(x) = h(T_E x)\phi_{d-m}(T_F x). \tag{17}$$

Now, let us define the *function* $g(u) = h(u)(\phi_m(u))^{-1}$. Then the above density can be rewritten

$$p(x) = g(T_E x)\phi_m(T_E x)\phi_{d-m}(T_F x) = g(T_E x)\phi_d(x), \tag{18}$$

i.e. can be put under the desired form (2). To deal with a more general covariance matrix of the noise, one performs a linear change of variables as $\widetilde{\boldsymbol{X}} = \Gamma^{-1/2}\boldsymbol{X}$, which brings us back to the above situation; then, simple calculations for linear change of variables in the density lead to the form (2).

B Proof of Proposition 1 (Blanchard et al., 2006)

Put $\boldsymbol{\alpha} = \mathbb{E}_{\boldsymbol{X}}[\boldsymbol{X}h(\boldsymbol{X})]$ and $\psi(\boldsymbol{x}) = h(\boldsymbol{x}) - \boldsymbol{\alpha}^\top \Sigma^{-1}\boldsymbol{x}$. Note that $\nabla\psi = \nabla h - \Sigma^{-1}\boldsymbol{\alpha}$, hence $\beta(h) = -\mathbb{E}_{\boldsymbol{X}}[\nabla\psi(\boldsymbol{X})]$. Furthermore, it holds by change of variable that

$$\int \psi(\boldsymbol{x}+\boldsymbol{u})p(\boldsymbol{x})d\boldsymbol{x} = \int \psi(\boldsymbol{x})p(\boldsymbol{x}-\boldsymbol{u})d\boldsymbol{x}.$$

Under mild regularity conditions on $p(\boldsymbol{x})$ and $h(\boldsymbol{x})$, differentiating the above equation with respect to $\boldsymbol{u}$ (or in other words, integration by parts) gives

$$\mathbb{E}_{\boldsymbol{X}}[\nabla\psi(\boldsymbol{X})] = \int \nabla\psi(\boldsymbol{x})p(\boldsymbol{x})d\boldsymbol{x} = -\int \psi(\boldsymbol{x})\nabla p(\boldsymbol{x})d\boldsymbol{x} = -\mathbb{E}_{\boldsymbol{X}}[\psi(\boldsymbol{X})\nabla \log p(\boldsymbol{X})]\ ,$$

where we have used $\nabla p(\boldsymbol{x}) = \nabla \log p(\boldsymbol{x})\, p(\boldsymbol{x})$. Eq.(2) now implies $\nabla \log p(\boldsymbol{x}) = \nabla \log g(T\boldsymbol{x}) - \Gamma^{-1}\boldsymbol{x}$, hence

$$\begin{aligned}\beta(h) &= \mathbb{E}_{\boldsymbol{X}}[\psi(\boldsymbol{X})\nabla \log g(T\boldsymbol{X})] - \mathbb{E}_{\boldsymbol{X}}\left[\psi(\boldsymbol{X})\Gamma^{-1}\boldsymbol{X}\right] \\ &= T^\top \mathbb{E}_{\boldsymbol{X}}[\psi(\boldsymbol{X})\nabla g(T\boldsymbol{X})/g(T\boldsymbol{X})] - \Gamma^{-1}\mathbb{E}_{\boldsymbol{X}}\left[\boldsymbol{X}h(\boldsymbol{X}) - \boldsymbol{X}\boldsymbol{X}^\top\Sigma^{-1}\mathbb{E}_{\boldsymbol{X}}[\boldsymbol{X}h(\boldsymbol{X})]\right]\ .\end{aligned}$$

The last term above vanishes because $\mathbb{E}_{\boldsymbol{X}}\left[\boldsymbol{X}\boldsymbol{X}^\top\right] = \Sigma$. The first term belongs to $\mathcal{I}$ by definition. This concludes the proof. □

References

H. Attias. Independent factor analysis. *Neural Computation*, 11(4):803–851, 1999.

F. R. Bach and M. I. Jordan. Kernel independent component analysis. *Journal of Machine Learning Research*, 3:1–48, 2002.

M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396, 2003.

C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

G. Blanchard, M. Kawanabe, M. Sugiyama, V. Spokoiny, and K.-R. Müller. In search of non-gaussian components of a high-dimensional distribution. *Journal of Machine Learning Research*, 7:247–282, 2006.

R. Boscolo, H. Pan, and V. P. Roychowdhury. Independent component analysis based on nonparametric density estimation. *IEEE Transactions on Neural Networks*, 15(1): 55–65, 2004.

A. Chen and P. J. Bickel. Efficient independent component analysis. *Annals of Statistics*, 34(6), 2006.

J. Eriksson, J. Karvanen, and V. Koivunen. Source distribution adaptive maximum likelihood estimation of ica model. In P. Pajunen and J. Karhunen, editors, *Proceedings of Second International Workshop on Independent Component Analysis and Blind Source Separation*, pages 227–232, 2000.

J. H. Friedman and J. W. Tukey. A projection pursuit algorithm for exploratory data analysis. *IEEE Transactions on Computers*, 23(9):881–890, 1974.

A. Gretton, R. Herbrich, A. Smola, O. Bousquet, and B. Schölkopf. Kernel methods for measuring independence. *Journal of Machine Learning Research*, 6:2075–2129, 2005.

S. Harmeling, A. Ziehe, M. Kawanabe, and K.-R. Müller. Kernel-based nonlinear blind source separation. *Neural Computation*, 15(5):1089–1124, 2003.

T. Hastie and R. Tibshirani. Independent components analysis through product density estimation. In S. Thrun S. Becker and K. Obermayer, editors, *Advances in Neural Information Processing Systems 15*, pages 649–656. MIT Press, Cambridge, MA, 2003.

P. J. Huber. Projection pursuit. *The Annals of Statistics*, 13:435–475, 1985.

A. Hyvärinen. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Transactions on Neural Networks*, 10(3):626–634, 1999.

A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis*. Wiley, 2001.

E. G. Learned-Miller and J. W. Fisher. ICA using spacing estimates of entropy. *Journal of Machine Learning Research*, 4:1271–1295, 2003.

T. W. Lee, M. Girolami, and T. J. Sejnowski. Independent component analysis using an extended informax algorithm for mixed subgaussian and supergaussian sources. *Neural Computation*, 11(2):417–441, 1999.

J. Moody and C.J. Darken. Fast learning in networks of locally-tuned processing units. *Neural Computation*, 1:281–294, 1989.

K.-R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf. An introduction to kernel-based learning algorithms. *IEEE Neural Networks*, 12(2):181–201, 2001.

G. Nason. *Design and Choice of Projection Indices*. PhD thesis, University of Bath, 1992.

S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

B. Schölkopf and A. Smola. *Learning with Kernels*. MIT Press, 2001.

C. M. Stein. Estimation of the mean of a multivariate normal distribution. *Annals of Statistics*, 9:1135–1151, 1981.

M. Sugiyama, M. Kawanabe, G. Blanchard, V. Spokoiny, and K.-R. Müller. Obtaining the best linear unbiased estimator of noisy signals by non-Gaussian component analysis. In *Proceedings of 2006 IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 608–611, 2006.

J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

F. J. Theis and M. Kawanabe. Uniqueness of non-gaussian subspace analysis. In *Proceedings of Sixth International Workshop on Independent Component Analysis and Blind Source Separation*, volume LNCS 3889, pages 917–924. Springer, 2006.

N. Vassis and Y. Motomura. Efficient source adaptivity in independent compoenent analysis. *IEEE Transactions on Neural Networks*, 12:559–566, 2001.