Non-Gaussian Component Analysis: SQ Hardness and Applications

Ilias Diakonikolas (UW Madison)

Simons Institute, Berkeley
September 2021

NON-GAUSSIAN COMPONENT ANALYSIS (NGCA)

Given samples from a distribution on $\mathbb{R}^d$, find a hidden "non-Gaussian" direction.

- Introduced in [Blanchard-Kawanabe-Sugiyama-Spokoiny-Muller'06].

- Studied extensively from algorithmic standpoint.
  [Kawanabe-Theis'06; Kawanabe-Sugiyama-Blanchard-Muller'07;
  Diederichs-Juditsky-Spokoiny-Schutte'10; Diederichs-Juditsky-Nemirovski-Spokoiny'13;
  Bean'14; Sasaki-Niu-Sugiyama'16; Virta-Nordhausen-Oja'16;
  Vempala-Xiao'11; Tan-Vershynin'18; Goyal-Shetty'19]

Non-Gaussian Component Analysis (NGCA): Definition

Definition: Let v be a unit vector in $\mathbb{R}^d$ and $A : \mathbb{R} \to \mathbb{R}_+$ be a pdf. We define $\mathbf{P}_v^A$ to be the distribution with v- projection equal to A and $v^\perp$- projection an independent standard Gaussian.

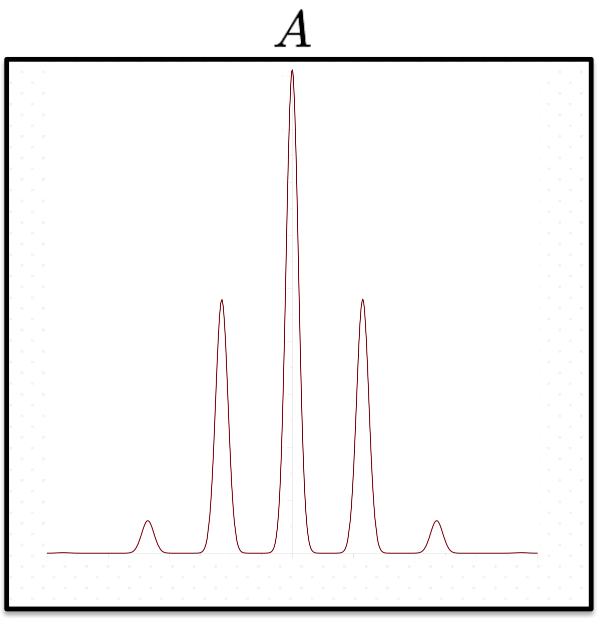
NGCA Problem: Given A that matches the first m moments with $\mathcal{N}(0,1)$:
Using i.i.d. samples from $\mathbf{P}_v^A$ where v is unknown, find the hidden direction v.

NGCA captures interesting instances of several well-studied learning tasks

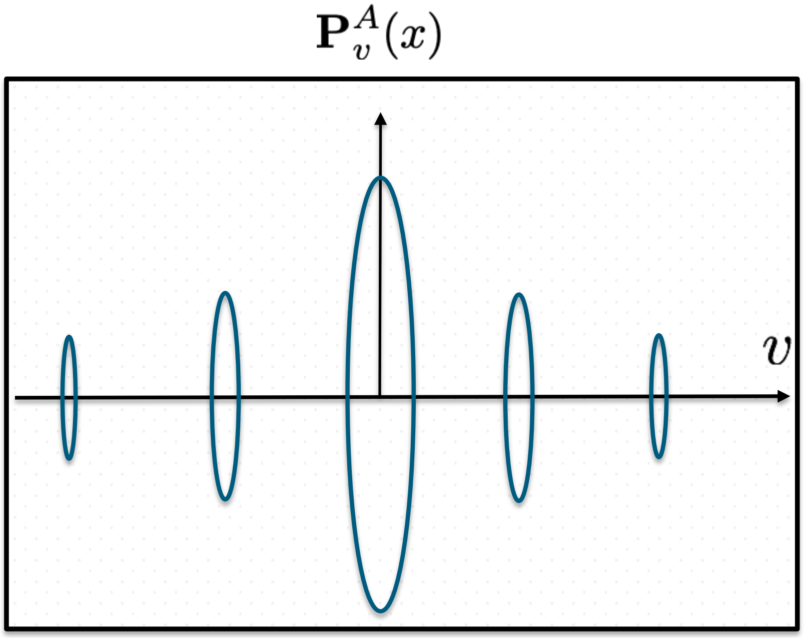
EXAMPLE: NGCA ENCODES GMMS

Note that $\mathbf{P}_v^A(x) = A(v \cdot x) \exp(-\|x - (v \cdot x)v\|_2^2)/(2\pi)^{(d-1)/2}$

Suppose that



then



(Informal) Main Result of This Talk

Fact: Non-Gaussian Component Analysis
- Can be solved with $\mathrm{poly}(d, m)$ samples.
- All known efficient algorithms require at least $d^{\Omega(m)}$ samples (and time).

Informal Theorem: For *any* “nice” univariate distribution A matching its first m moments with the standard Gaussian, any* algorithm that solves NGCA
- either draws at least $d^{\Omega(m)}$ samples
- or has runtime $2^{d^{\Omega(1)}}$.

*holds for any **Statistical Query (SQ)** algorithm

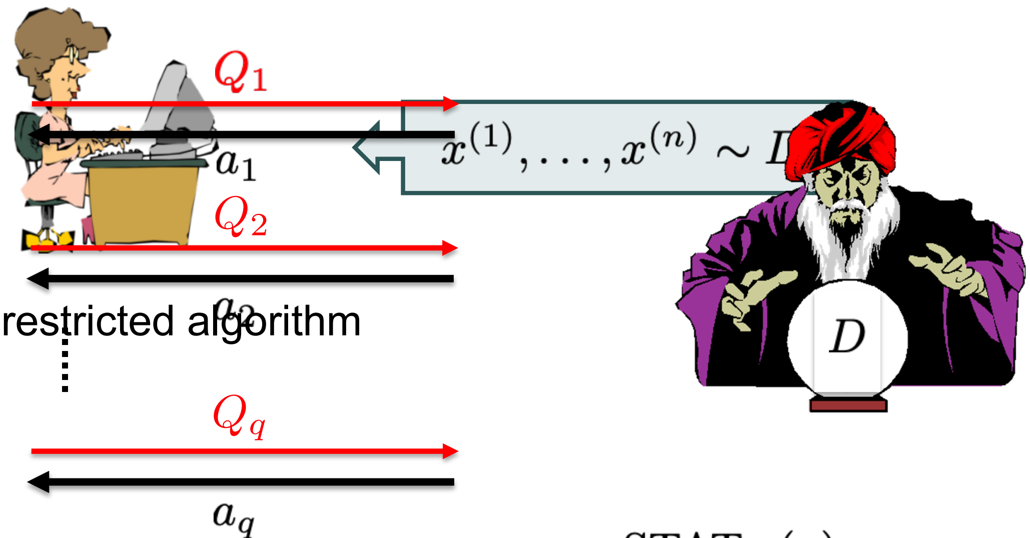
[D-Kane-Stewart, FOCS’17]

NGCA captures SQ hard instances of several well-studied learning tasks

- Learning GMMs [D-Kane-Stewart'17]
- Robust mean and covariance estimation [D-Kane-Stewart'17]
- Robust sparse mean estimation, sparse PCA [D-Kane-Stewart'17, D-Stewart'18]
- Robust linear regression [D-Kong-Stewart'19]
- List-decodable learning [D-Kane-Stewart'18, D-Kane-Pensia-Pittas-Stewart'21]
- Adversarially robust PAC learning [Bubeck-Price-Razenshteyn'18]
- Agnostic PAC Learning [Goel-Gollakota-Klivans'20, D-Kane-Zarifis'20, D-Kane-Pittas-Zarifis'21]
- Learning Neural Networks [Goel-Gollakota-Jin-Karmalkar-Klivans'20, D-Kane-Kontonis-Zarifis'20]
- Learning with Massart Noise [D-Kane'20]

Statistical Query (SQ) Model [Kearns'93]

SQ algorithm



Statistical Query: $Q_i : X \to [-1, 1]$

$\mathrm{STAT}_D(\tau)$ returns a_i :

$|a_i - \mathbf{E}_{x \sim D}[Q_i(x)]| \leq \tau$

tolerance of the query

Complexity measures

- Number of queries: q — Runtime
- Query tolerance: τ — Sample complexity

Interpretation of SQ *Lower Bounds*

Suppose we have proved:

Any SQ algorithm for problem P

- either requires queries of **tolerance** at most τ
- or makes at least q **queries**.

Then we can interpret:

Any SQ algorithm* for problem P

- either requires at least $1/\tau^2$ **samples**
- or has **runtime** at least q.

POWER OF SQ ALGORITHMS

- **Restricted Model**: Can prove unconditional lower bounds.

- **Powerful Model**: Wide range of algorithmic techniques in ML are implementable using SQs:

  - PAC Learning: AC^0, decision trees, linear separators, boosting
  - Unsupervised Learning: stochastic convex optimization, moment-based methods, *k*-means clustering, EM, … [Feldman-Grigorescu-Reyzin-Vempala-Xiao, JACM'17]

- **Known Exception**: Gaussian elimination over finite fields (aka, learning parities).

- For all problems in this talk, strongest known algorithms are SQ.

General Methodology For SQ Lower Bounds

Hypothesis Testing Problem: Given access to a distribution D on $\mathbb{R}^d$ with promise that

- either $D = D_0$
- or D is selected randomly from $\mathcal{D} = \{D_u\}_{u \in S}$ according to prior μ

the goal is to distinguish between the two cases.

Pairwise correlation: $\chi_{D_0}(p, q) = \mathbf{E}_{x \sim D_0}[(p/D_0)(x)(q/D_0)(x)] - 1$

Theorem [FGRVX'17]: Suppose there exists a "large" set of distributions in $\mathcal{D}$ with "small" pairwise correlation with respect to D_0 . Then any SQ algorithm for hypothesis testing task:

- either requires at least one "high-accuracy" query
- or requires a "large" number of queries.

Statistical Query Hardness of NGCA

Testing Version of NGCA: Given access to a distribution D on $\mathbb{R}^d$ with the promise that

- either $D = \mathcal{N}(0, I)$
- or $D = \mathbf{P}_v^A$, where v is a uniformly random unit vector

the goal is to distinguish between the two cases.

Main Theorem [D-Kane-Stewart'17]

Suppose that A matches its first m moments with $\mathcal{N}(0,1)$ and $\chi^2(A, \mathcal{N}(0,1)) < \infty$.

Any SQ algorithm for the testing version of NGCA:

- either requires a query of tolerance at most $d^{-\Omega(m)} \chi^2(A, \mathcal{N}(0,1))^{1/2}$
- or requires at least $2^{d^{\Omega(1)}}$ many queries.

INTUITION: WHY IS NGCA "HARD"?

Claim 1: Low-degree moments do not help.

- Degree at most m moment tensor of $\mathbf{P}_v^A$ identical to that of $\mathcal{N}(\mathbf{0}, I_d)$

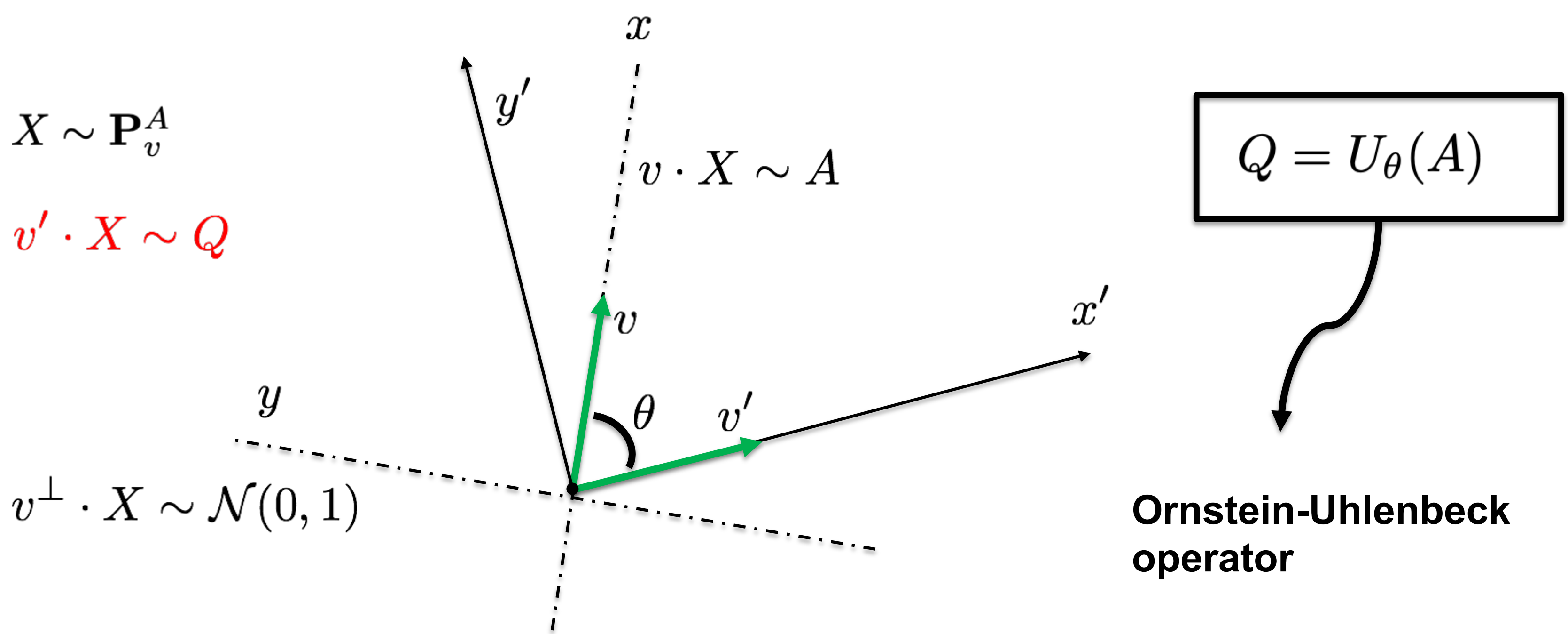
Claim 2: Random projections do not help.

Distinguishing requires exponentially many random projections.

KEY LEMMA: RANDOM PROJECTIONS ARE *ALMOST* GAUSSIAN

Key Lemma: Let Q be the distribution of $v' \cdot X$, where $X \sim \mathbf{P}_v^A$. Then, we have that:

$$\chi^2(Q, \mathcal{N}(0,1)) \leq (v \cdot v')^{2(m+1)} \chi^2(A, \mathcal{N}(0,1))$$



SQ Lower Bound: Proof Overview

Want exponentially many $\mathbf{P}_v^A$'s that are nearly uncorrelated.

- Pick set $\mathcal{V}$ of near-orthogonal unit vectors. Can get $|\mathcal{V}| = 2^{d^{\Omega(1)}}$
- Have

$$\chi_{\mathcal{N}(\mathbf{0},I_d)}(\mathbf{P}_v^A, \mathbf{P}_{v'}^A) = \chi_{\mathcal{N}(0,1)}(A, U_\theta A) \leq |\cos^{m+1}(\theta)|\chi^2(A, \mathcal{N}(0,1))$$

■

Recipe For SQ Hardness Results

Main Theorem [D-Kane-Stewart'17]

Suppose that A matches its first m moments with $\mathcal{N}(0,1)$ and $\chi^2(A, \mathcal{N}(0,1)) < \infty$.

Any SQ algorithm for the testing version of NGCA:

- either requires a query of tolerance at most $d^{-\Omega(m)} \chi^2(A, \mathcal{N}(0,1))^{1/2}$
- or requires at least $2^{d^{\Omega(1)}}$ many queries.

Recipe. Encode Π as a NGCA instance:

- Construct moment-matching distribution A such that $\mathbf{P}_v^A$ is a **valid instance** of Π .
- Match **as many low-degree moments as possible**.

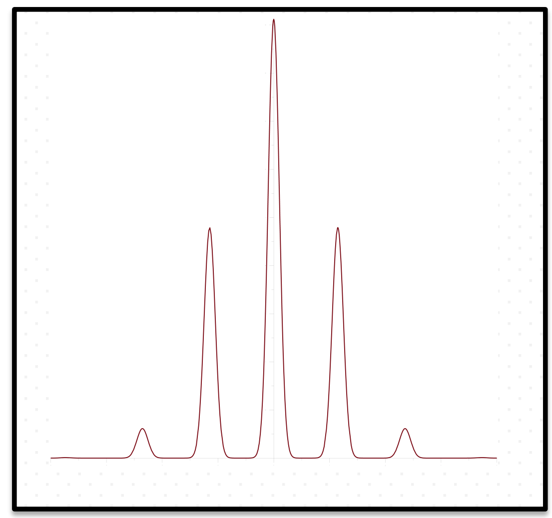
Example: SQ Hardness of Learning GMMs

Lemma: There exists a univariate k-GMM A with nearly non-overlapping components such that: A agrees with $\mathcal{N}(0,1)$ on the first $2k$-1 moments.

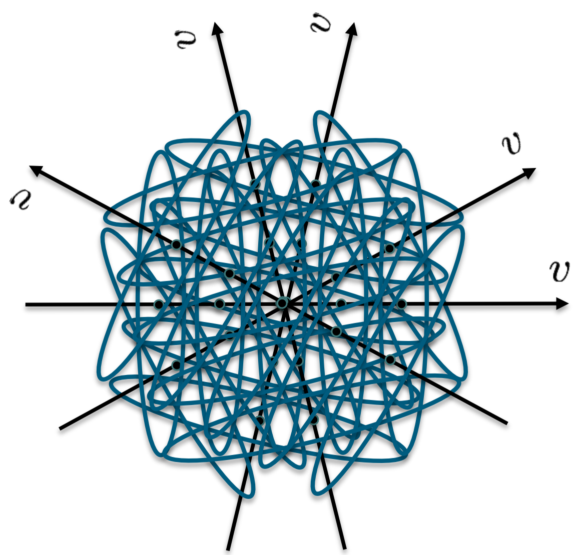
Proof Idea:

- Construct discrete distribution B with support k matching its first $2k$-1 moments with $\mathcal{N}(0,1)$.
- Rescale B and add a "skinny" Gaussian to get A.

A



SQ Hard Instances for GMMs: Parallel Pancakes



SQ Hardness For Wide Range of Problems

NGCA captures SQ hard instances of several well-studied learning tasks

- Learning GMMs [D-Kane-Stewart'17]
- Robust mean and covariance estimation [D-Kane-Stewart'17]
- Robust sparse mean estimation, sparse PCA [D-Kane-Stewart'17, D-Stewart'18]
- Robust linear regression [D-Kong-Stewart'19]
- List-decodable learning [D-Kane-Stewart'18, D-Kane-Pensia-Pittas-Stewart'21]
- Adversarially robust PAC learning [Bubeck-Price-Razenshteyn'18]
- Agnostic PAC Learning [Goel-Gollakota-Klivans'20, D-Kane-Zarifis'20, D-Kane-Pittas-Zarifis'21]
- Learning Neural Networks [Goel-Gollakota-Jin-Karmalkar-Klivans'20, D-Kane-Kontonis-Zarifis'20]
- Learning with Massart Noise [D-Kane'20]
- …

Conclusions and Open Problems

NGCA leads to wide range of hardness results in SQ model

Open Problem 1: Alternative evidence of hardness?

Already known for special cases (reduction-based):

- Robust sparse mean estimation [Brennan-Bresler'20]
- Learning GMMs [Bruna-Regev-Song-Tang'21]

SQ hard instances are computationally hard.

Open Problem 2: How general is this phenomenon?

Open Problem 3: Prove SoS lower bounds for NGCA.