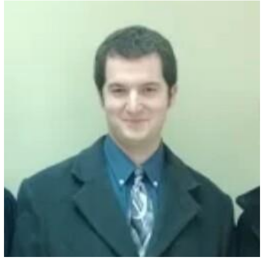
Efficient learning of simplices

Joseph Anderson (OSU)
Navin Goyal (MSR)
Luis Rademacher (OSU)

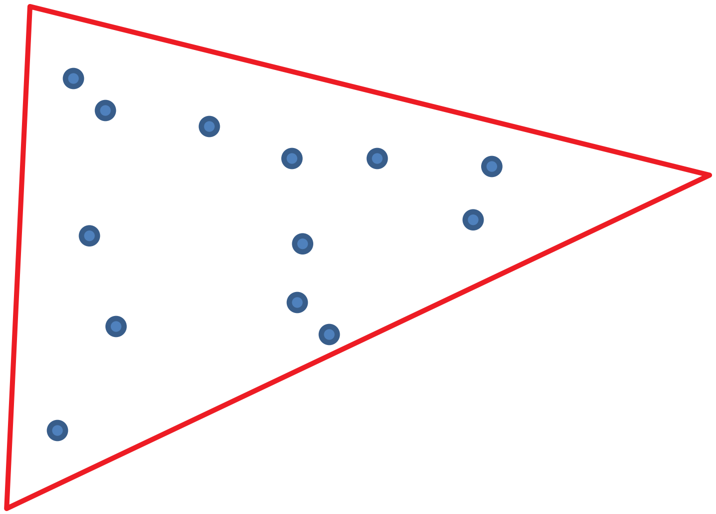


Motivation - Approach

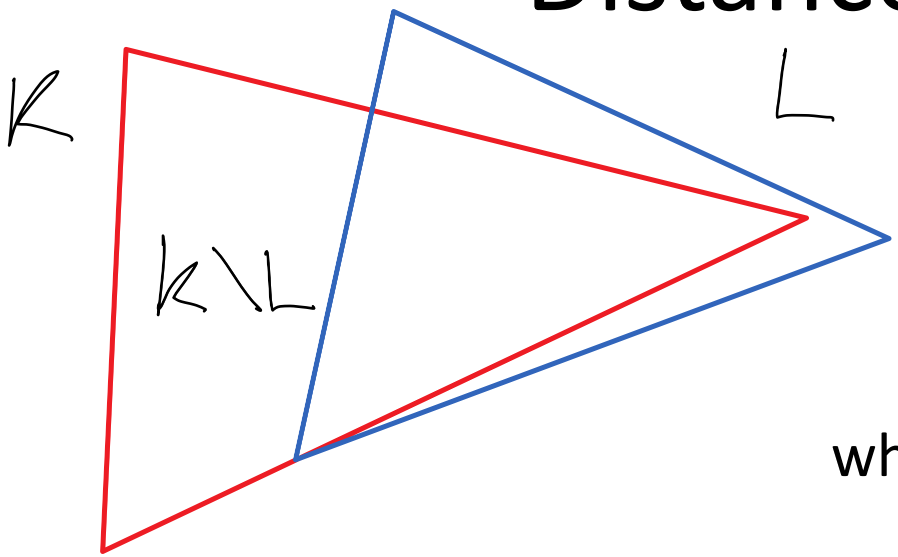
- Understand gap between sample complexity and algorithmic complexity in high dimensional estimation.
- Study prototypical problems: properties of distributions related to high dimensional convex bodies.

Problem

- Given uniformly random points from an n-dimensional simplex, estimate the simplex. Metric: total variation distance.



Distance



$$d(K, L) = \frac{\mathrm{vol}(K \setminus L)}{\mathrm{vol}\, K}$$

when $\mathrm{vol}\, K \geq \mathrm{vol}\, L$

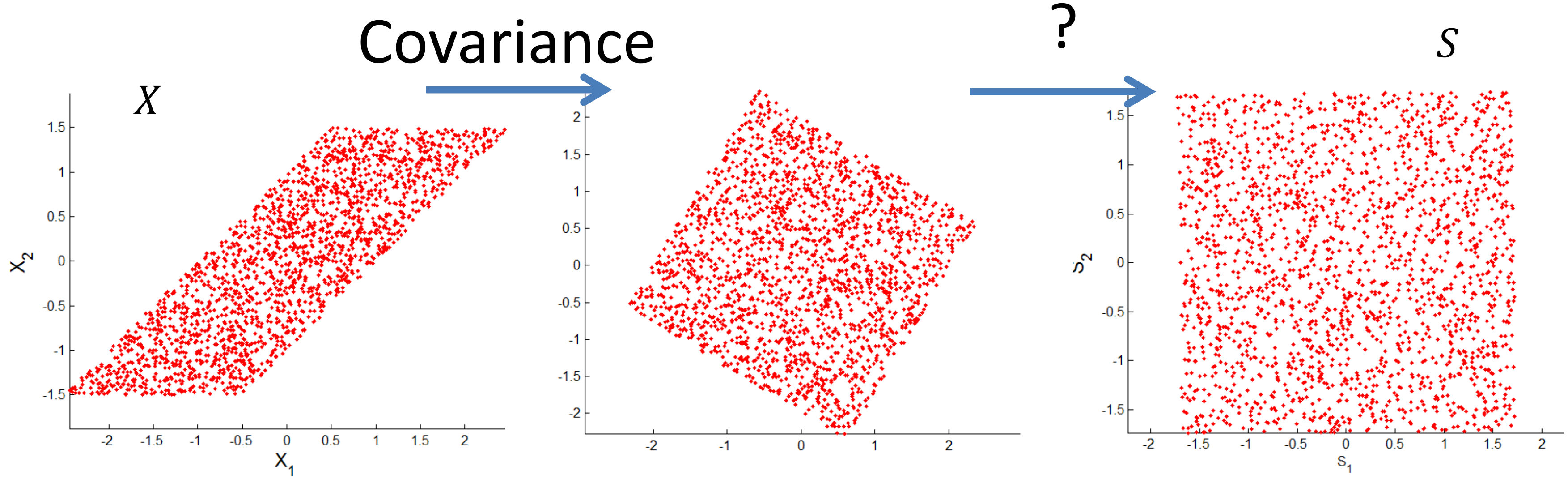
- $d(K, L) =$ total variation distance (statistical distance) between uniform distributions on K and L.

More specific motivation

- Efficient estimation is an open question in [Frieze Jerrum Kannan ‘96].
- VC-theory implies polynomial sample complexity. Efficient algorithm?
- Latent variable models: more general problems can be solved with similar techniques (independent work, see [Anandkumar Foster Hsu Kakade Liu ‘12] for Latent Dirichlet Allocation, [A. Ge H. K. Telgarsky ‘12] for other latent variable models.)
- Understand techniques from Independent Component Analysis “beyond independence.”

Independent Component Analysis (ICA)

- Idea: given samples from random vector $X = AS$, where $S \in R^n$ has independent coordinates, estimate $A \in R^{n \times n}$ (up to natural ambiguities, e.g. permutation of columns of A).



Our results

- An algorithm for estimating a simplex in polynomial time.
- A randomized reduction from learning a simplex to ICA.
- A similar reduction from learning an affine transformation of an l_p-ball to ICA.

Related work

- [FJK ‘96] Efficient learning of linear transformations, in particular, parallelepipeds.
- [Nguyen Regev ‘06] Efficient learning of parallelepipeds.
- [Goyal R. ‘09] [Klivans O’Donnell Servedio ’09] Learning an arbitrary convex body needs $c^{\sqrt{n}}$ samples.
- [KOS ’09], [Vempala ‘97 ‘10] [Kane Klivans Meka ‘13][…] Polynomial time algorithm to learn an intersection of any constant number of half-spaces under restricted distributions (superpolynomial for simplex)
- [Anandkumar Foster Hsu Kakade Liu ‘12] Different algorithm (matrix decompositions applied to 3rd moment) to estimate Dirichlet distribution (generalizes uniform distribution on simplex)
- [A. Ge H. K. Telgarsky ‘12] Tensor power method for tensor diagonalization. Can estimate Dirichlet distribution (similar to our algorithm, independent work).
- [Gravin Lasserre Pasechnik Robins ‘12] Reconstruction of polytopes with few vertices from higher order moments. No polynomial time guarantee.

Algorithm to learn a simplex

- Can assume that simplex is regular and centered (isotropic): subtract mean and apply $E(XX^T)^{-1/2}$. Need to find rotation. How?
- Given an isotropic (i.e. cov=I, mean=0) simplex S, consider the functional
$$F(u) = E_{X\in S}((u\cdot X)^3)$$
- Then the (normalized) vertices of S are a complete set of local and global maxima over unit sphere.
- Can provably enumerate local maxima via a variation of gradient descent.

Understanding 3rd moment

- Easier: affinely embed R^d in $\{x \in R^{d+1}: \sum x_i = 1\}$. Compute 3rd moment of standard simplex, $\mathrm{conv}\{e_i\}$:

$$G(u) = c_1 h_3(u) \ \text{e.g. [Lasserre Avrachenkov]}$$
$$\text{(3rd complete homogenenous polynomial)}$$
$$= c_2 \sum_i u_i^3$$

(under $\sum_i u_i = 0, \sum_i u_i^2 = 1$, using Newton's identities))

- Use Lagrange multipliers and 2nd order optimality conditions to enumerate local maxima with constraints.

Finding vertices efficiently

- Fixed point-like iteration:
  - From gradient of 3rd moment $G(u)$, can compute $u \mapsto (u_i^2)$, even under the current rotation leaving **1** invariant (i.e. coordinate-free):
    $\nabla G(u) = c(u_i^2) + c'\mathbf{1}[(\sum u_i)^2 + \sum u_i^2] + c''u\sum u_i$
  - The sequence of normalized iterates converges to some canonical vector, which is a vertex of the standard simplex
    (gradient descent leads to similar idea in [NR '06]).
  - Somewhat unexpectedly for a gradient-based method, the convergence is extremely fast, *quadratic.*

From learning l_p ball to ICA.

- Simplex reduction is similar, l_p ball (B_p^n) is easier.
- Idea:
  - If X is random in B_p^n,
  - T is $Gamma((n/p)+1,1)$,
  - then we show the coordinates of $T^{\frac{1}{p}}X$ are independent (using ideas from [Schechtman Zinn '90] [Barthe Guedon Mendelson Naor '05])

From learning l_p ball to ICA.

- Suppose X is random in B_p^n and $Y = AX$ is random in a linearly transformed l_p ball. It follows that $T^{1/p}Y = A(T^{1/p}X)$ has independent components.
  ICA can recover A (up to irrelevant ambiguities, permutation and sign).
- The reduction then takes a sample $Y_1, \dots, Y_N$ and outputs $T_1^{1/p}Y_1, \dots, T_N^{1/p}Y_N$.

Discussion

- Open problem: Efficient learning of polytopes with $poly(n)$ facets (or $poly(n)$ vertices).
- Questions?