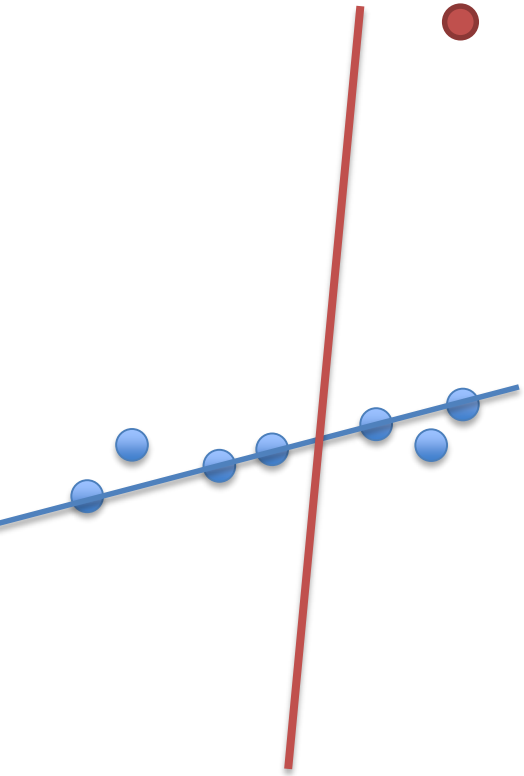
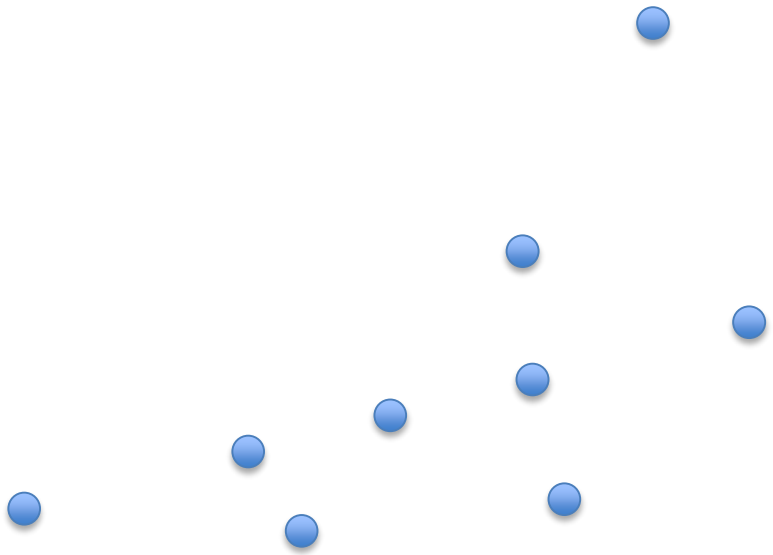


Algorithms and Hardness for Robust Subspace Recovery

Moritz Hardt
IBM Research Almaden

Joint work with Ankur Moitra (Princeton)

Robust Subspace Recovery



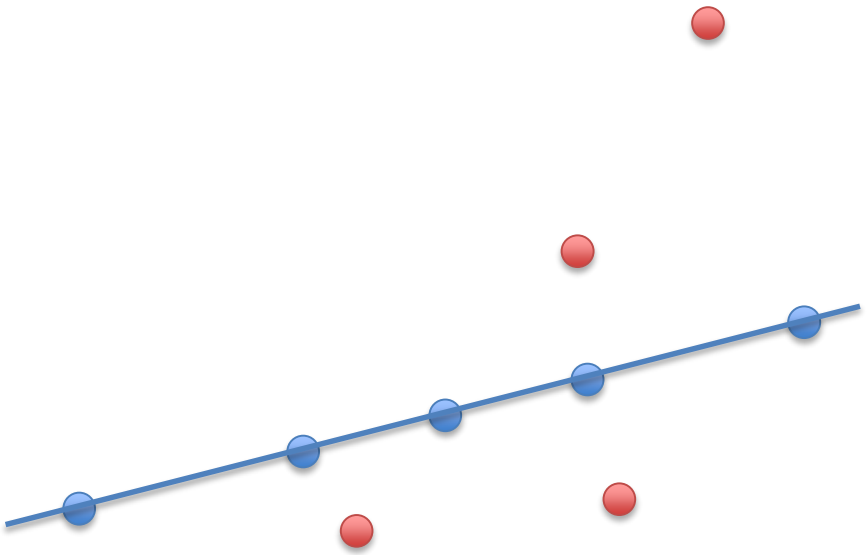
Goal:
Recover subspace

Given: m points in $\mathbb{R}^n$

Inliers: contained in *unknown* d-dimensional subspace

Outliers: general position but otherwise adversarial

Robust Subspace Recovery



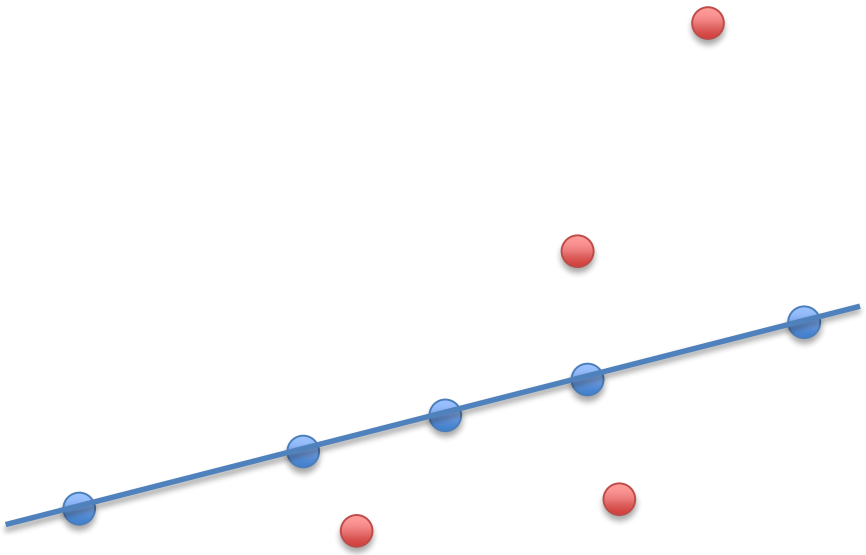
Goal:
Recover subspace

Given: m points in $\mathbb{R}^n$

Inliers: contained in *unknown* d-dimensional subspace

Outliers: general position but otherwise adversarial

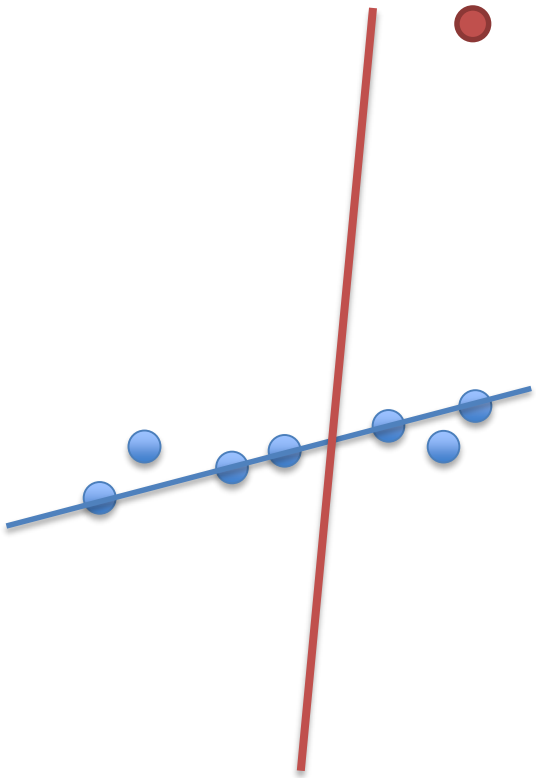
Robust Subspace Recovery



Goal:
Recover subspace

The Dilemma: In high dimension,
algorithms either not robust to *adversarial* outliers
or not *computationally* efficient

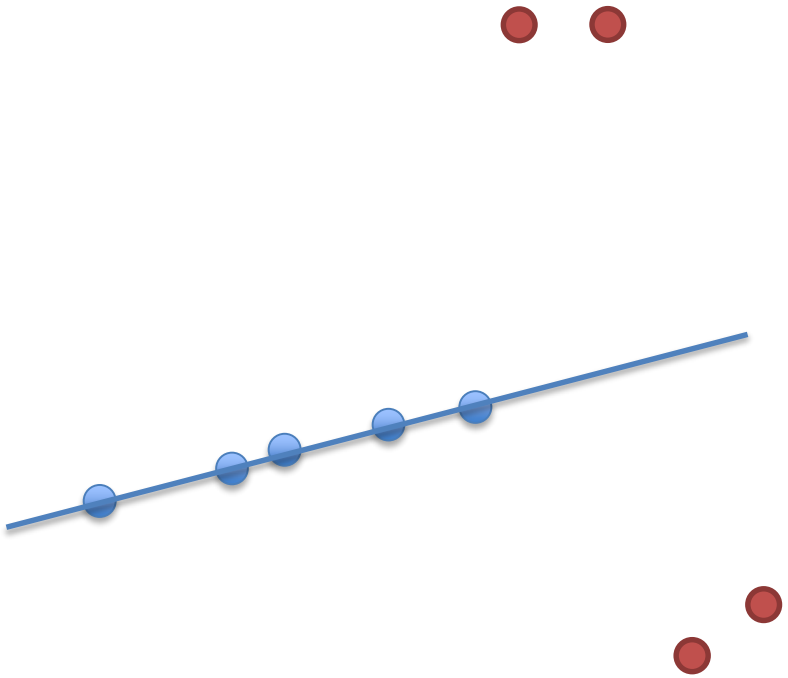
Example (Least Squares): Find subspace that minimizes sum of squared Euclidean distances



Efficient!

Non-Robust!

Example (Least Median of Squares, Rousseeuw 1984): Find subspace that minimizes median of squared Euclidean distances



Robust!

Not efficiently computable!

Robust Stats Terminology

Definition: An estimator (i.e., algorithm) has a breakdown point of *p* if it recovers subspace from any *p* fraction of outliers.

Statistical threshold = highest *breakdown point of any statistical estimator*

Extremely well-studied in Robust Stats

Computational threshold = highest *breakdown point of any* ***computationally efficient*** *estimator*

Not well understood

In this talk:

Example: d = n/100, can recover from 99% outliers, and, 99.1% is intractable.

The computational threshold of
Robust Subspace Recovery is **(1 - d/n)**

* assuming plausible complexity assumption

New algorithms: Efficient algorithm that recovers subspace from (1-d/n) fraction of outliers

New hardness result: It is "Small Set Expansion hard" to recover from more than (1-d/n)+ε fraction of outliers

New convex program and duality principle:
Feasibility of robust subspace recovery is dual to natural geometric condition

Lots of related work

- Robust PCA [CLMW09,XCS10,CSPW11, TLMZ12,XCM13,...]
  - e.g., [XCM13]: different objective function (expressed variance), different assumptions (at most 50% outliers, constant d, distribution on inliers)
- Variants of least squares in other norms
  - same behavior in terms of breakdown point
- Outlier Removal [DV00]
- RANSAC method [FB81]

Overview

- Randomized Algorithm
- Connection to Small Set Expansion
- Derandomization, Convex Program, Duality
  - (Homework: See paper)

Simple Randomized Algorithm

Condition 1: A set of n points is linearly independent iff it contains at most d inliers.

Theorem: If a set of points contains at least d/n fraction of inliers and meets Condition 1, then we can find the hidden subspace in randomized polynomial time.

Can replace Condition 1 by:

Condition 2: Any set of at most n points has "det > C" if it contains at most d inliers and "det < C" otherwise.

Algorithm (Randomized Find):
Input: Matrix A = $[x_1 | \ldots | x_m]$
Let U = [m].
Sample subset V of size n.
If rank(A_V) < n:
    Let u ≠ 0 in ker(A_V)
    Let S = supp(u)
    Output L^* = span(A_S)
Else:
    repeat

Repeatedly sample set of *n* points and check for linear dependence.

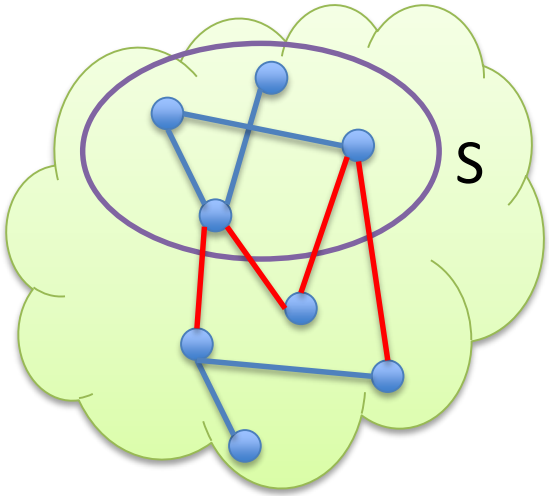
E[#inliers in *V*] = d, since d/n fraction of inliers in A

Analysis: Support of linear dependence reveals inliers (by Condition 1)

Hardness

Informal Statement:

Improving over threshold (1-d/n) *refutes* Small Set Expansion Hypothesis [Raghavendra-Steurer'10].



G=(V,E)
Δ-regular graph

Small set expansion

$$\phi_G(S) = \frac{|E_G(S, V \setminus S)|}{\Delta |S|}$$

SSE(ε,δ): Given G, decide if

(YES) $\forall S, |S| = \delta|V|: \phi_G(S) \geq 1 - \epsilon$

(NO) $\exists S, |S| = \delta|V|: \phi_G(S) \leq \epsilon$

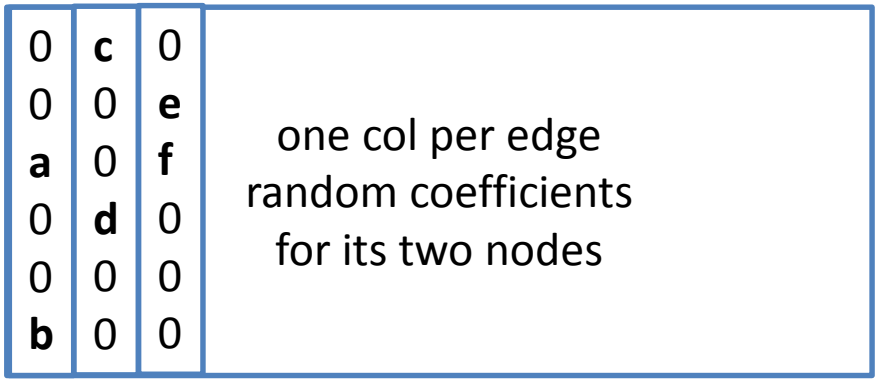
SSE Hypothesis: For every $\varepsilon>0$, there exists δ such that Gap-SSE(ε,δ) is NP-hard.

Reduction from Gap-SSE

Given graph
G=(*V*,*E*)

$|E|$

$|V|$

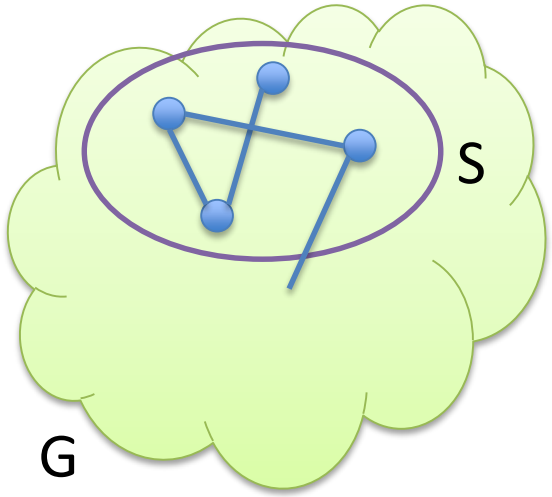


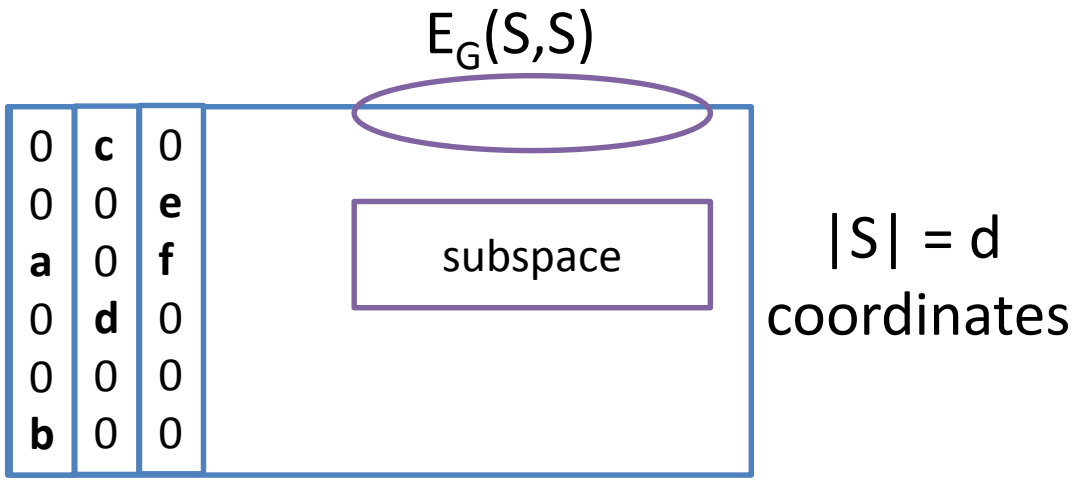
a,b,c,d,e,f,... random numbers in [0,1]

Only linear dependences due to coordinate subspaces!

Completeness (easy direction):

Small non-expanding set → Low-dimensional coordinate subspace containing many points





Soundness (bit more sublte):

Subspace containing many points → *Coordinate* subspace containing many points → Small non-expanding set

Summary

Pinned down computational threshold of natural formalization of robust subspace recovery

- No distributional assumptions necessary, adversarial outliers
- Interesting connections to small set expansion, (not in talk: basis polytope, functional analysis)
- Can relax exact subspace containment slightly
  - Is it possible to understand computational threshold of *further relaxations*?

Where's the challenge?

Distributional assumptions on points	Adversarial/deterministic points
Leads to algorithms with stronger guarantees	Difficult to design algorithms with *provable* guarantees
Hardness machinery likely to break down	Hardness results within scope of existing techniques

Open Problems

What is the *computational threshold* of other variants of robust subspace recovery, robust PCA, robust subspace clustering, etc?

Broad goal:

Understand trade-off between robustness and computational efficiency in unsupervised learning

Thank you.

Hardness

Informal Result: Improving over threshold d/n *refutes* Small Set Expansion Hypothesis [Raghavendra-Steurer'10].

Theorem: For every ε>0, there is an efficient reduction taking a graph G to a point set such that:

1. small ε-expanding set in G
   → subspace of dimension d containing (1−ε) d/n fraction of the points
2. every small set (1-ε)-expanding
   → every subspace of dimension d contains at most ε d/n fraction of the points.