



projection, learning and sparsity for efficient data processing

Rémi Gribonval
Inria Rennes - Bretagne Atlantique



remi.gribonval@inria.fr

Contributors & Collaborators



■ Anthony Bourrier



■ Nicolas Keriven



■ Yann Traonmilin



■ Gilles Puy



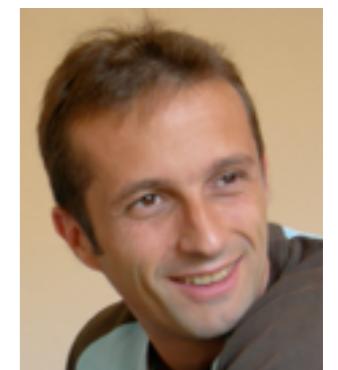
■ Gilles Blanchard



■ Mike Davies



■ Tomer Peleg



■ Patrick Perez

Agenda

- From Compressive Sensing to Compressive Learning ?
- Information-preserving projections & sketches
- Compressive Clustering / Compressive GMM
- Conclusion

Machine Learning

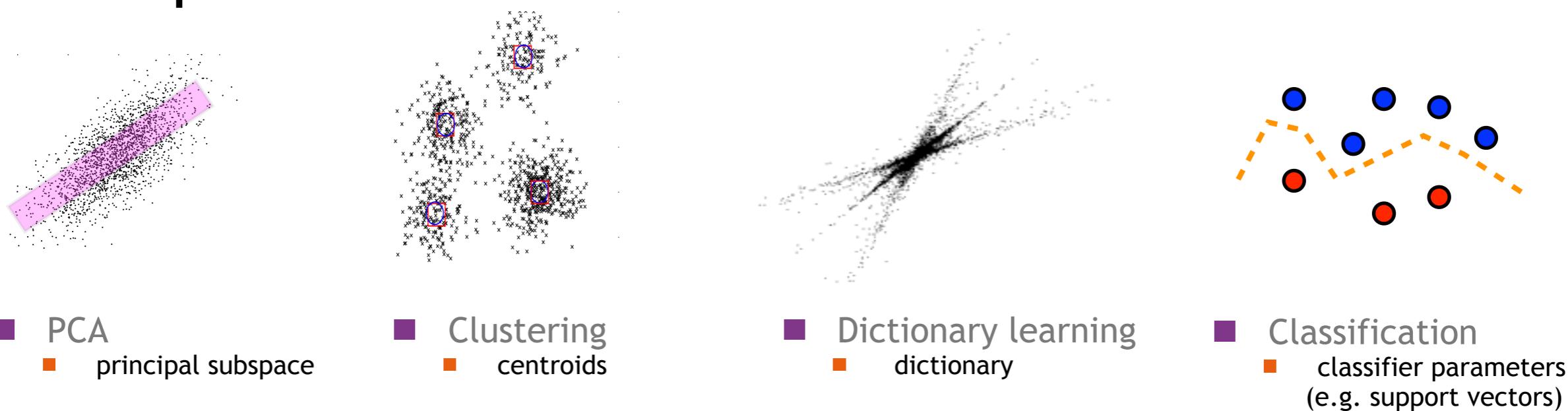
■ Available data

- training collection of feature vectors = point cloud \mathcal{X}

■ Goals

- infer parameters to achieve a certain task
- generalization to future samples with the same probability distribution

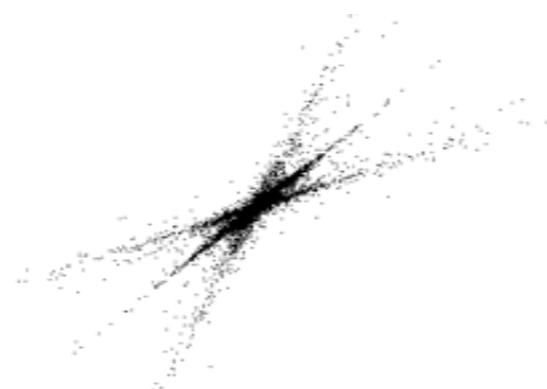
■ Examples



Challenging dimensions

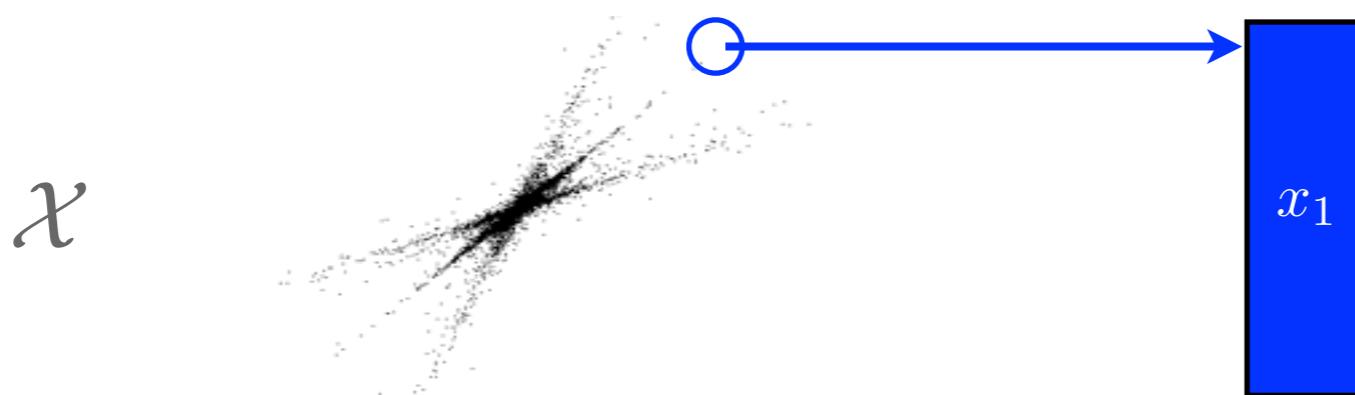
- Point cloud = large matrix of feature vectors

\mathcal{X}



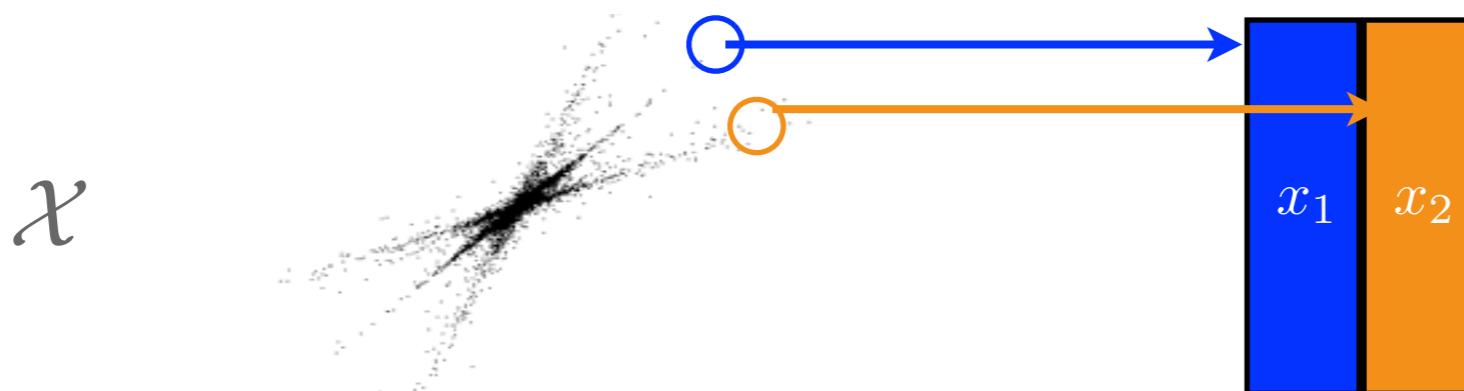
Challenging dimensions

- Point cloud = large matrix of feature vectors



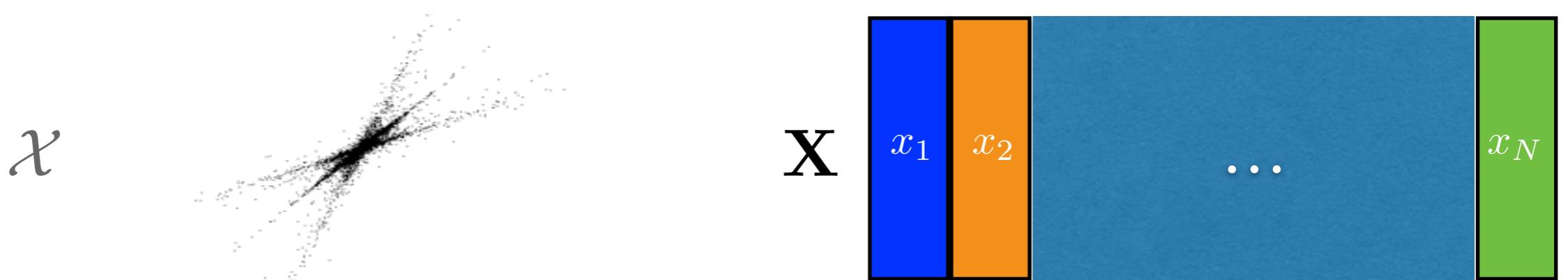
Challenging dimensions

■ Point cloud = large matrix of feature vectors



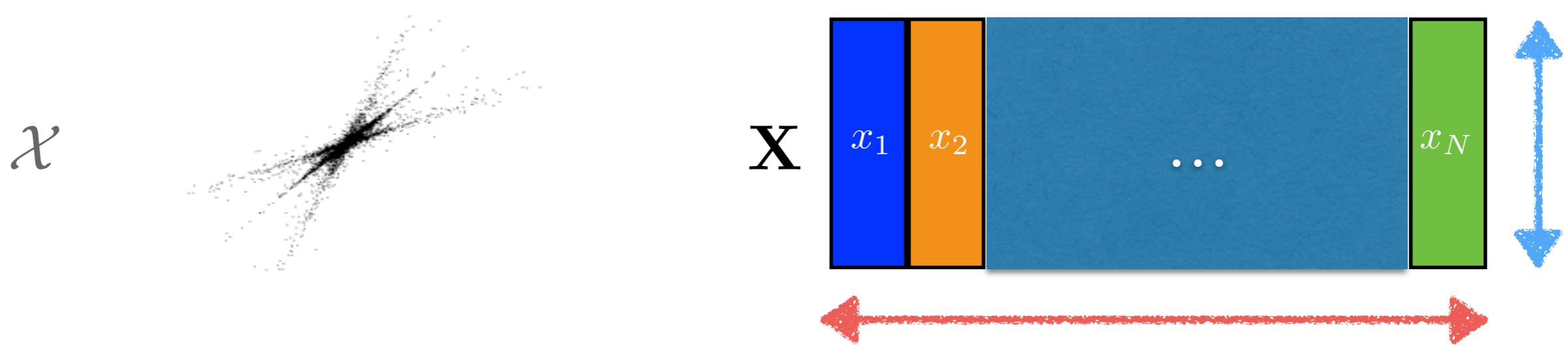
Challenging dimensions

- Point cloud = large matrix of feature vectors



Challenging dimensions

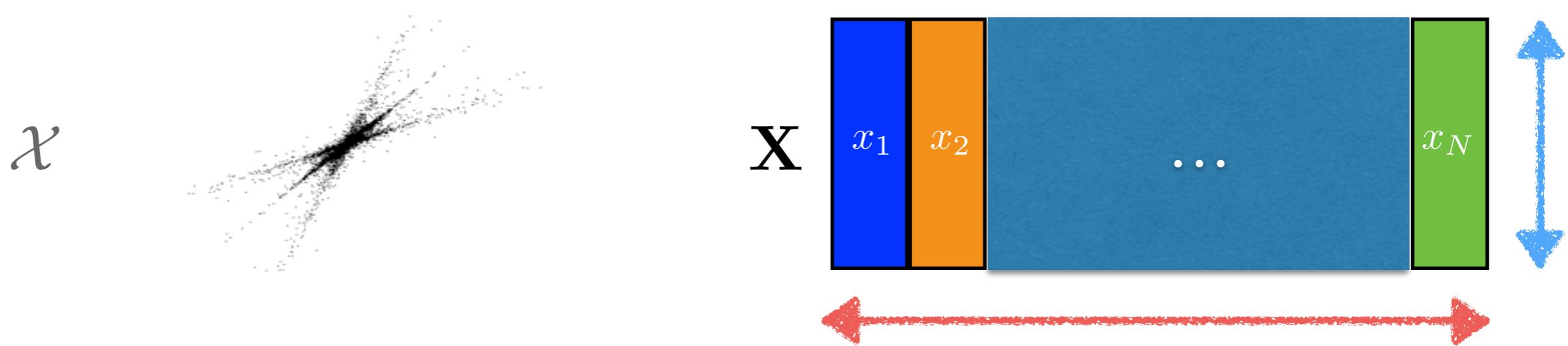
- Point cloud = large matrix of feature vectors



- High feature dimension n
- Large collection size N

Challenging dimensions

- Point cloud = large matrix of feature vectors

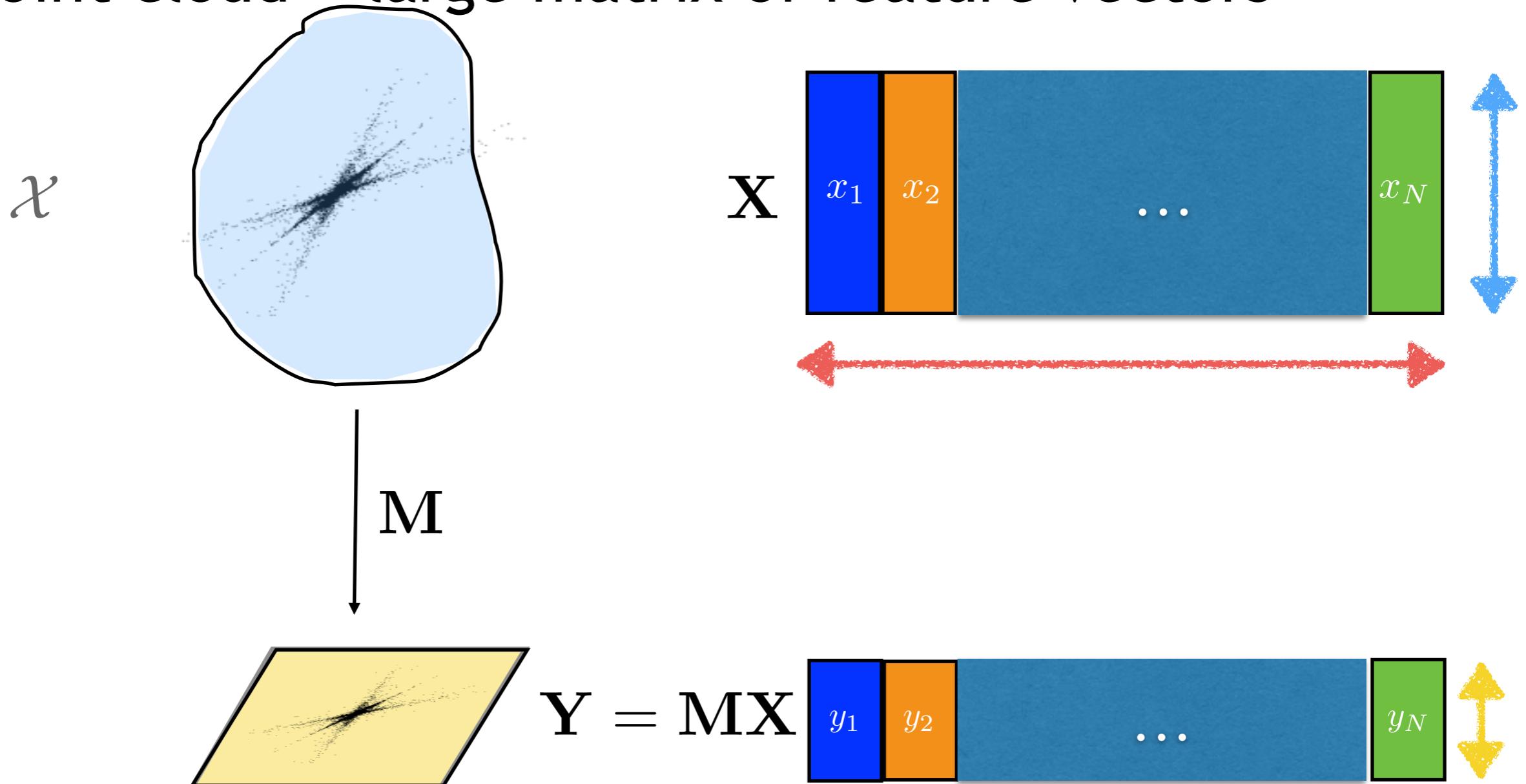


- High feature dimension n
- Large collection size N

Challenge: compress \mathcal{X} before learning ?

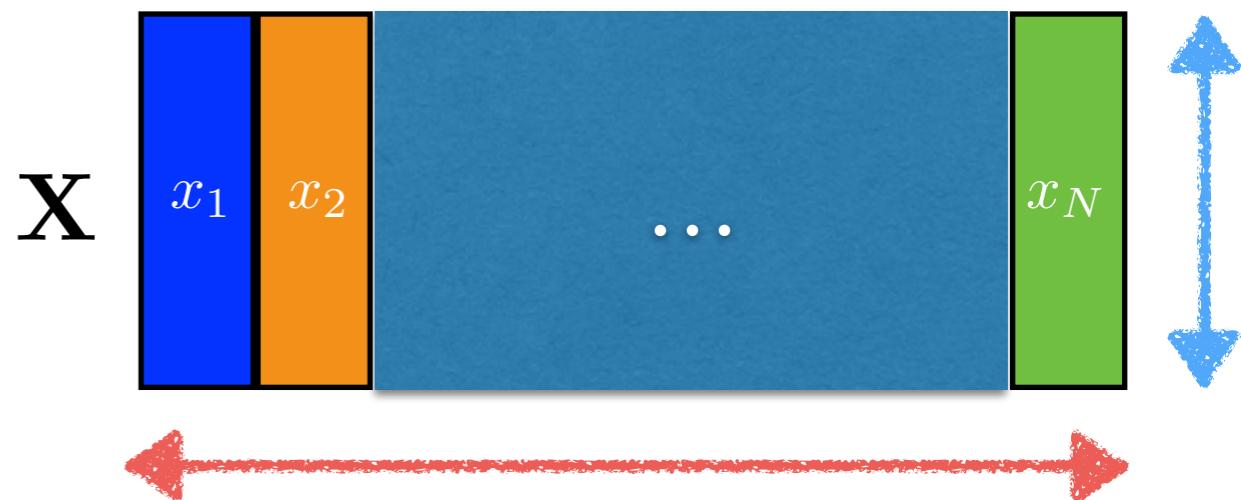
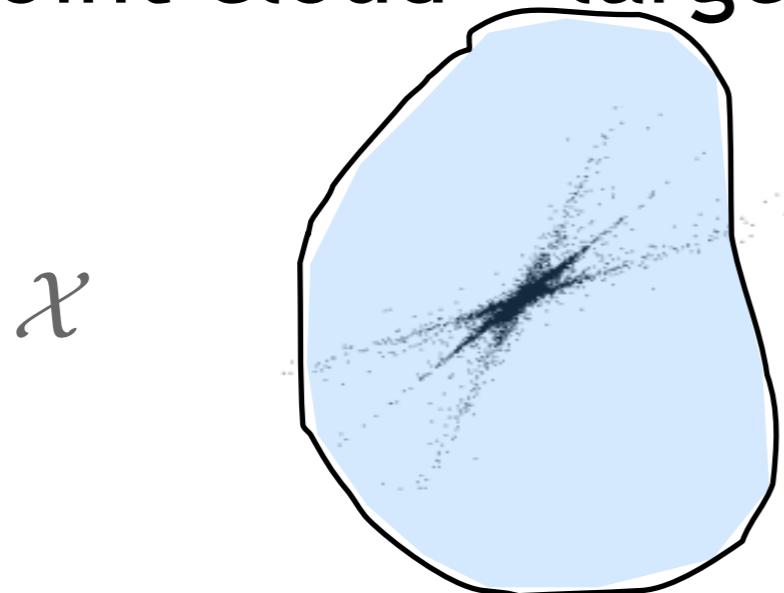
Compressive Machine Learning ?

■ Point cloud = large matrix of feature vectors



Compressive Machine Learning ?

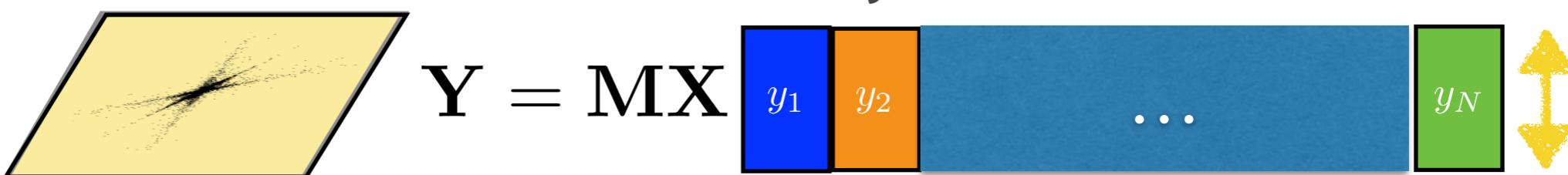
- Point cloud = large matrix of feature vectors



- Reduce feature dimension

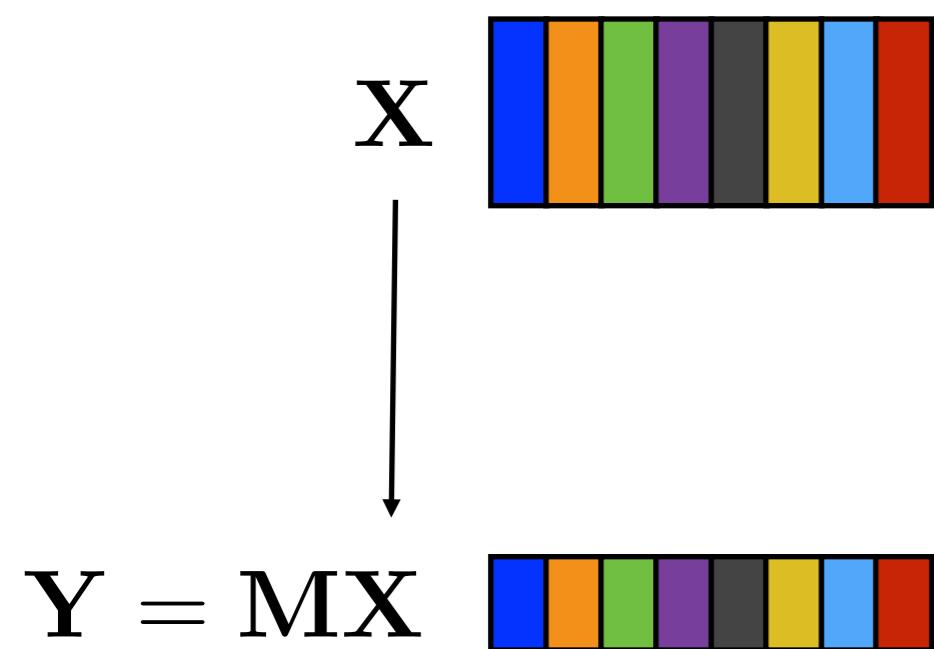
[Calderbank & al 2009, Reboredo & al 2013]

- (Random) feature projection
- Exploits / needs low-dimensional *feature model*



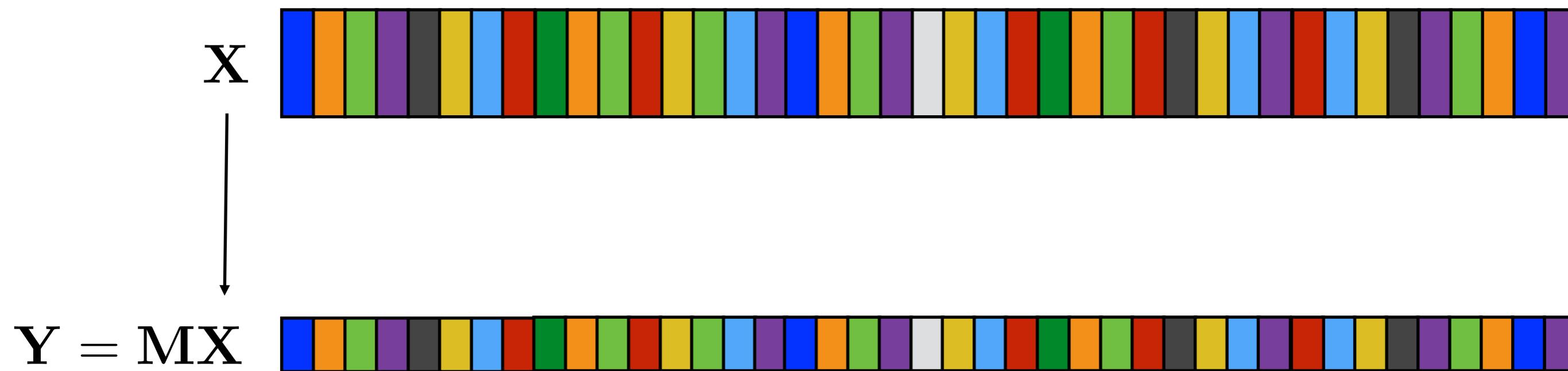
Challenges of large collections

- Feature projection: limited impact



Challenges of large collections

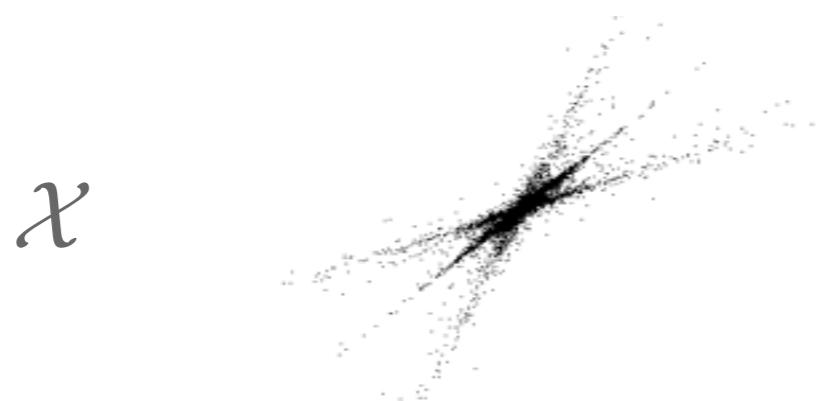
■ Feature projection: limited impact



“Big Data” Challenge: compress **collection size**

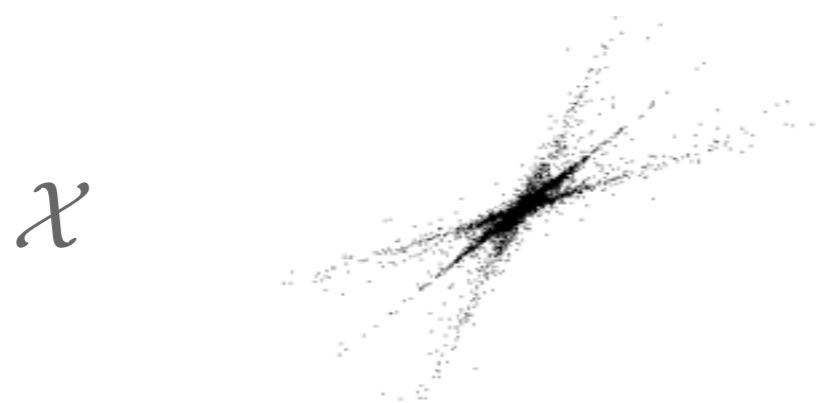
Compressive Machine Learning ?

■ Point cloud



Compressive Machine Learning ?

■ Point cloud



■ Reduce collection dimension

■ coresets

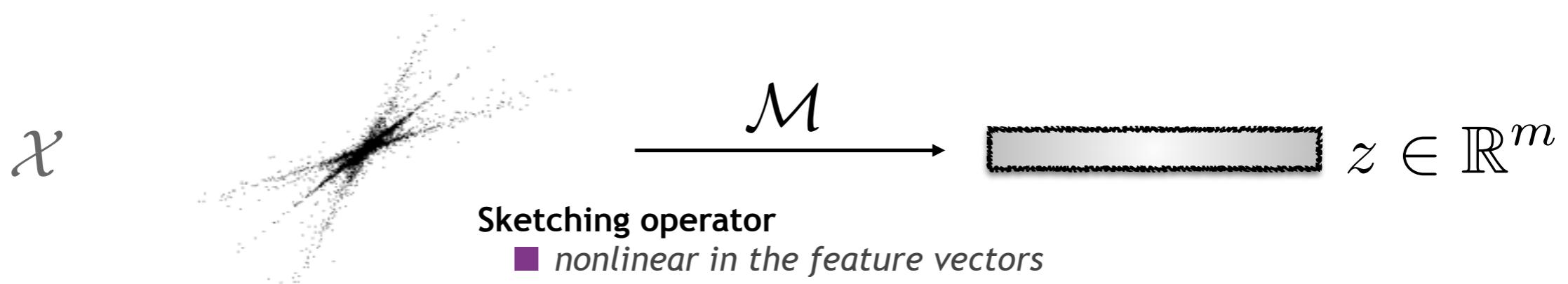
see e.g. [Agarwal & al 2003, Felman 2010]

■ sketching & hashing

see e.g. [Thaper & al 2002, Cormode & al 2005]

Compressive Machine Learning ?

■ Point cloud



■ Reduce collection dimension

■ coresets

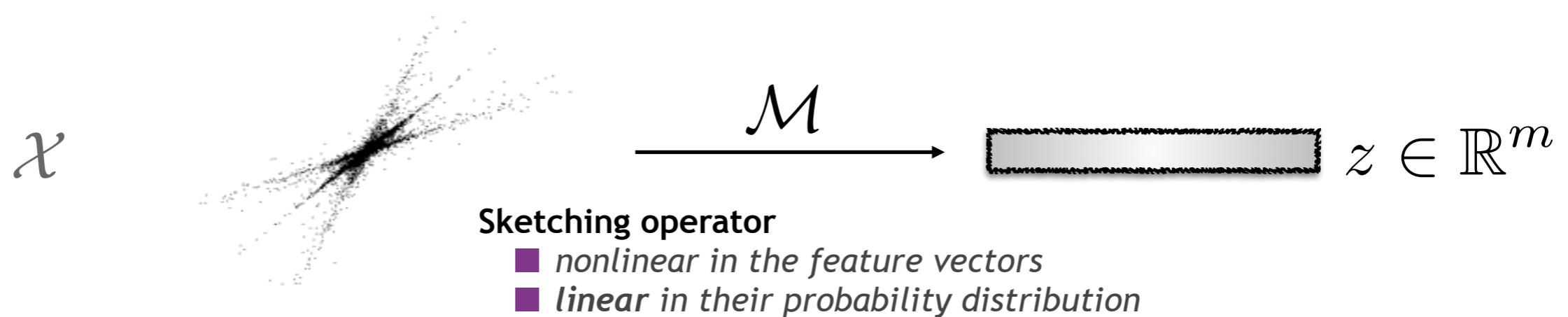
see e.g. [Agarwal & al 2003, Felman 2010]

■ sketching & hashing

see e.g. [Thaper & al 2002, Cormode & al 2005]

Compressive Machine Learning ?

■ Point cloud = ... empirical probability distribution



■ Reduce collection dimension

■ coresets

see e.g. [Agarwal & al 2003, Felman 2010]

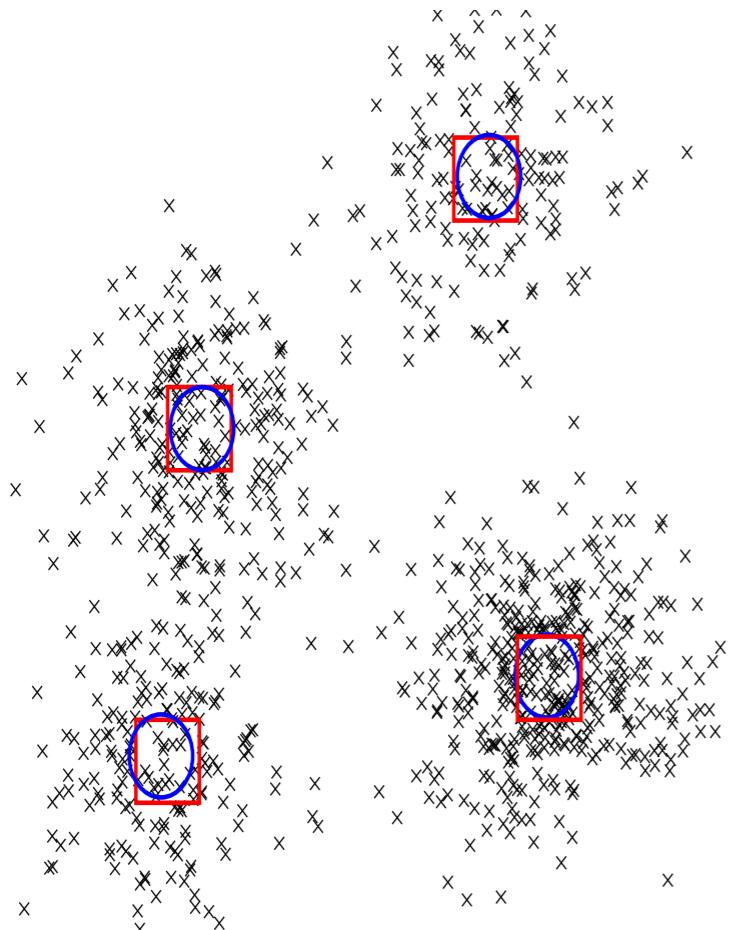
■ sketching & hashing

see e.g. [Thaper & al 2002, Cormode & al 2005]

Example: Compressive Clustering

$$\mathcal{X} \xrightarrow{\mathcal{M}} z \in \mathbb{R}^m$$

$N = 1000; n = 2$ $m = 60$



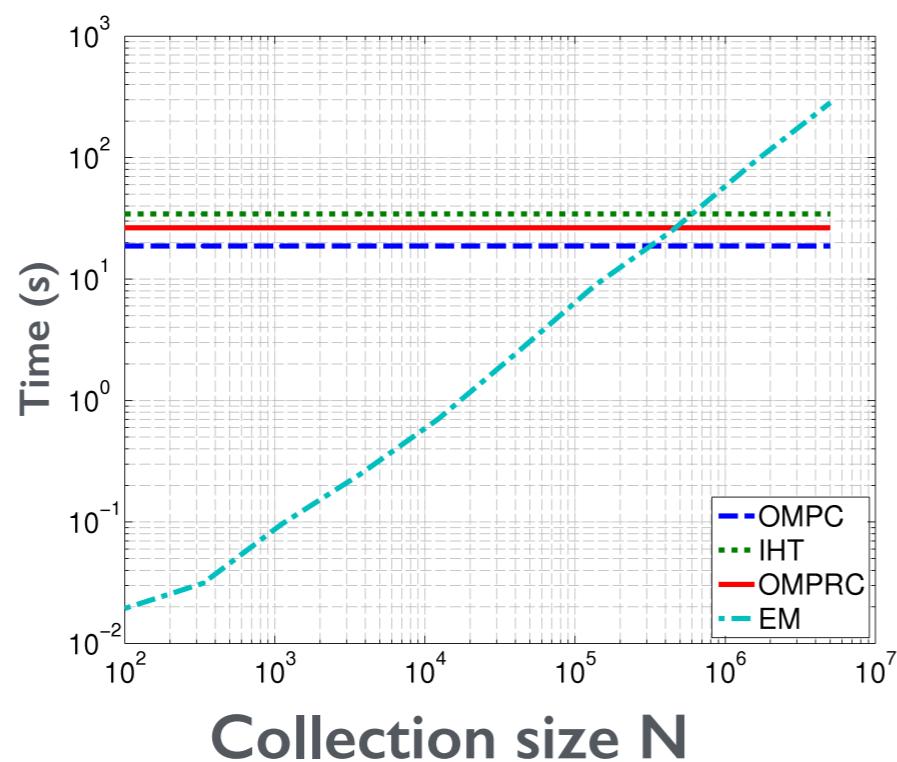
Recovery algorithm

□ estimated centroids

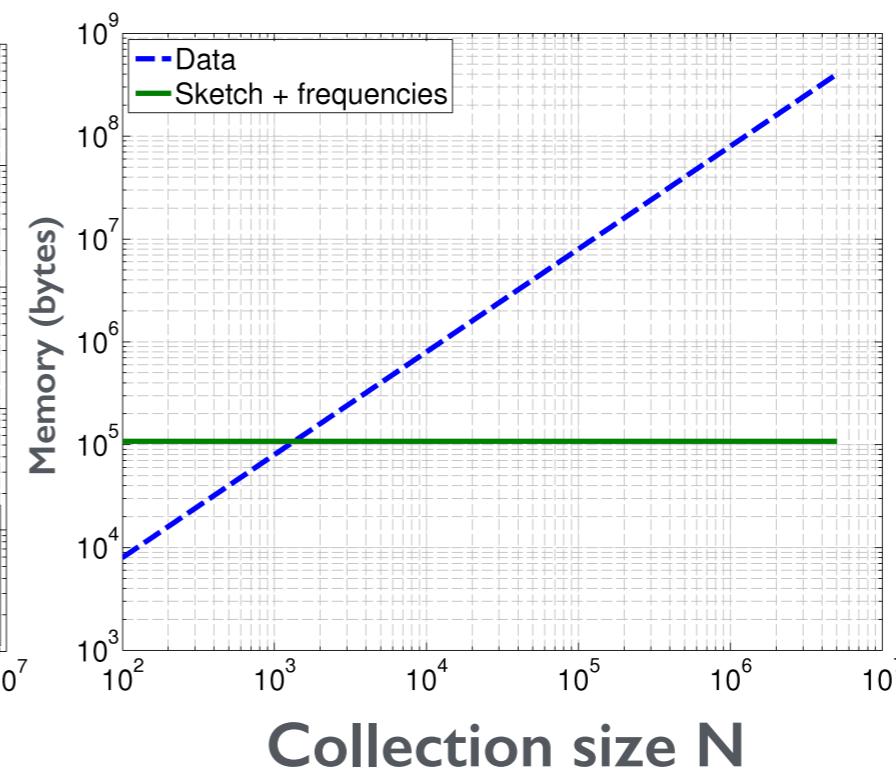
○ ground truth

Computational impact of sketching

Computation time



Memory



Ph.D. A. Bourrier & N. Keriven

The Sketch Trick

■ Data distribution

$$X \sim p(x)$$

■ Sketch

$$z_\ell = \frac{1}{N} \sum_{i=1}^N h_\ell(x_i)$$

The Sketch Trick

■ Data distribution

$$X \sim p(x)$$

■ Sketch

$$z_\ell = \frac{1}{N} \sum_{i=1}^N h_\ell(x_i)$$

$$\approx \mathbb{E} h_\ell(X)$$

The Sketch Trick

■ Data distribution

$$X \sim p(x)$$

■ Sketch

$$z_\ell = \frac{1}{N} \sum_{i=1}^N h_\ell(x_i)$$

$$\approx \mathbb{E} h_\ell(X)$$

$$= \int h_\ell(x) p(x) dx$$

The Sketch Trick

■ Data distribution

$$X \sim p(x)$$

■ Sketch

$$z_\ell = \frac{1}{N} \sum_{i=1}^N h_\ell(x_i)$$

$$\approx \mathbb{E} h_\ell(X)$$

$$= \int h_\ell(x) p(x) dx$$

- nonlinear in the feature vectors
- linear in the distribution $p(x)$

The Sketch Trick

■ Data distribution

$$X \sim p(x)$$

■ Sketch

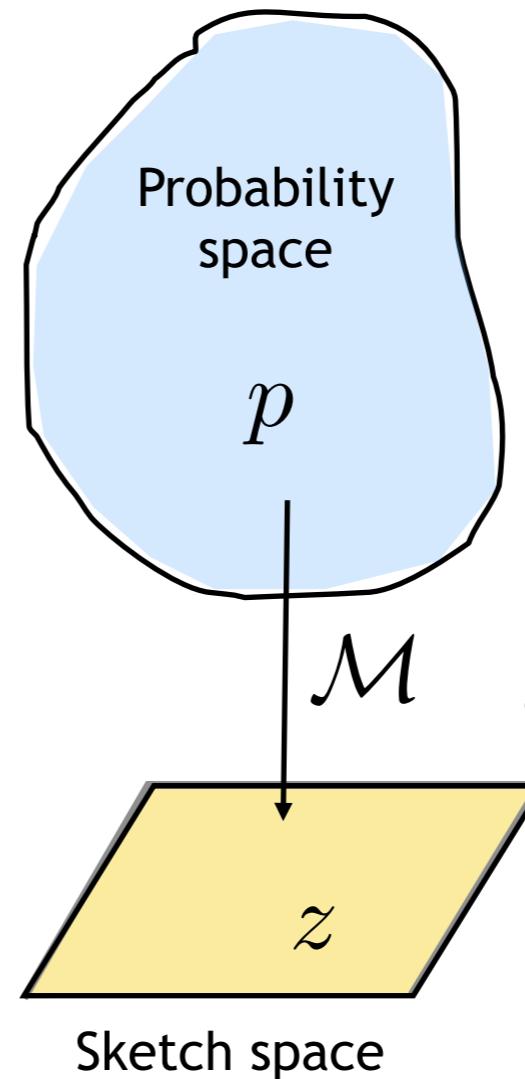
$$z_\ell = \frac{1}{N} \sum_{i=1}^N h_\ell(x_i)$$

$$\approx \mathbb{E} h_\ell(X)$$

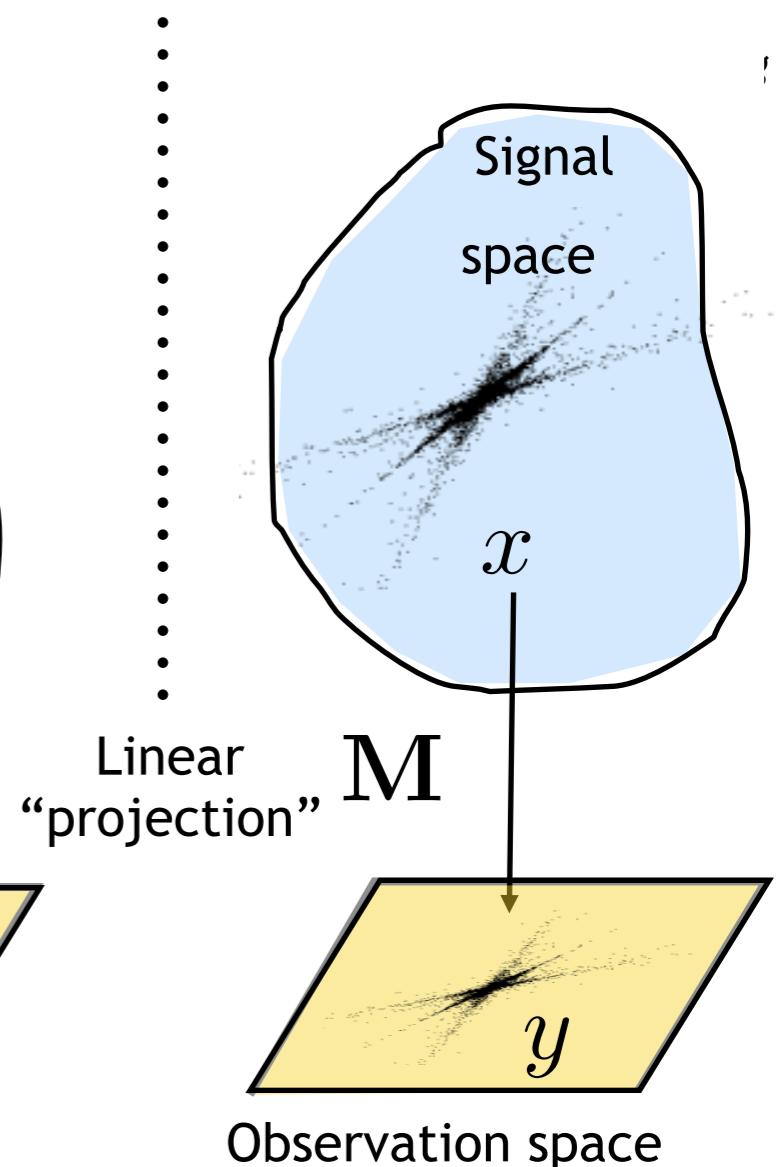
$$= \int h_\ell(x)p(x)dx$$

- nonlinear in the feature vectors
- linear in the distribution $p(x)$

■ Machine Learning



■ Signal Processing



The Sketch Trick

Information preservation ?

■ Data distribution

$$X \sim p(x)$$

■ Sketch

$$z_\ell = \frac{1}{N} \sum_{i=1}^N h_\ell(x_i)$$

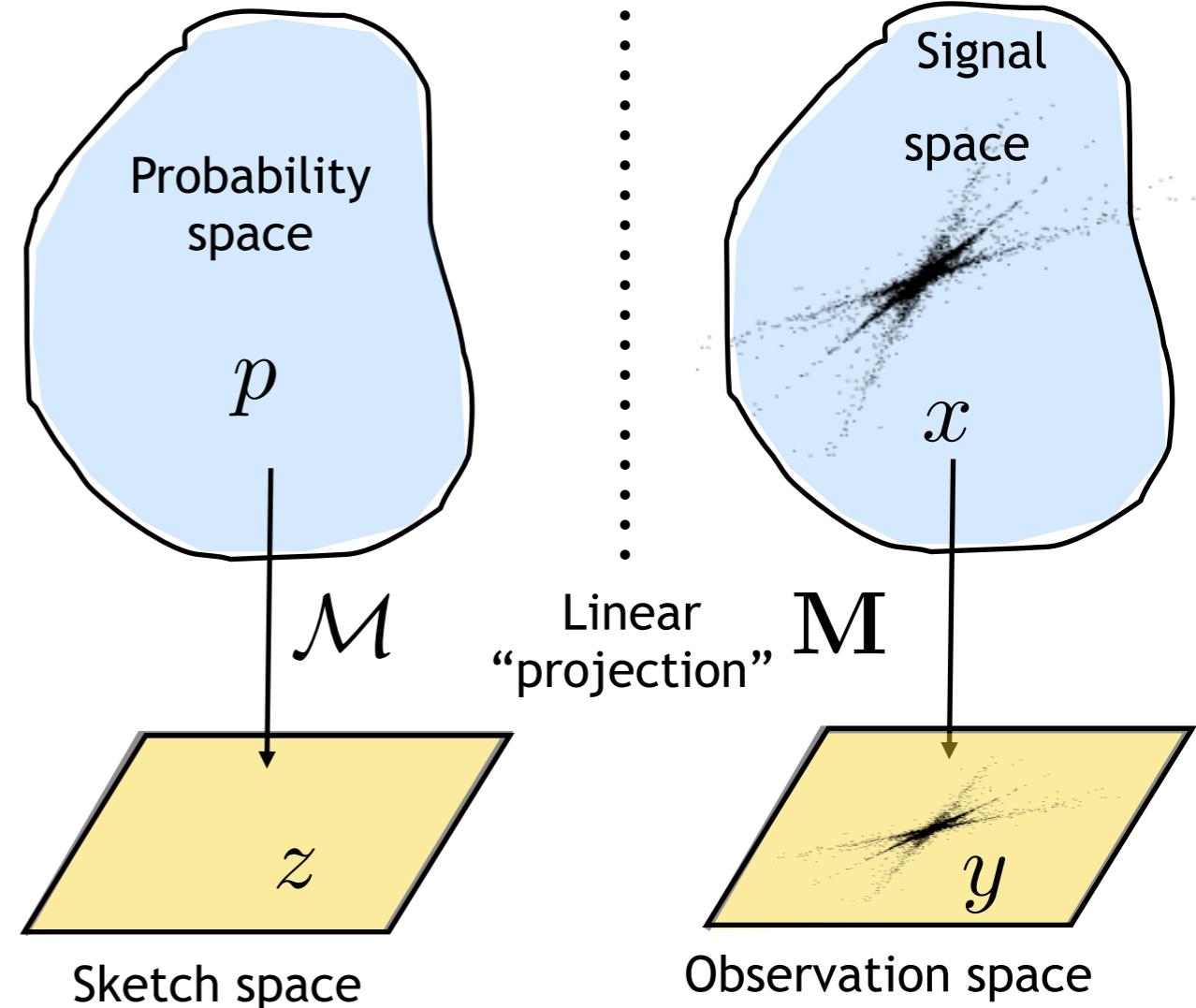
$$\approx \mathbb{E} h_\ell(X)$$

$$= \int h_\ell(x)p(x)dx$$

- nonlinear in the feature vectors
- linear in the distribution $p(x)$

■ Machine Learning
■ method of moments

■ Signal Processing
■ inverse problems



The Sketch Trick

Dimension reduction ?

■ Data distribution

$$X \sim p(x)$$

■ Sketch

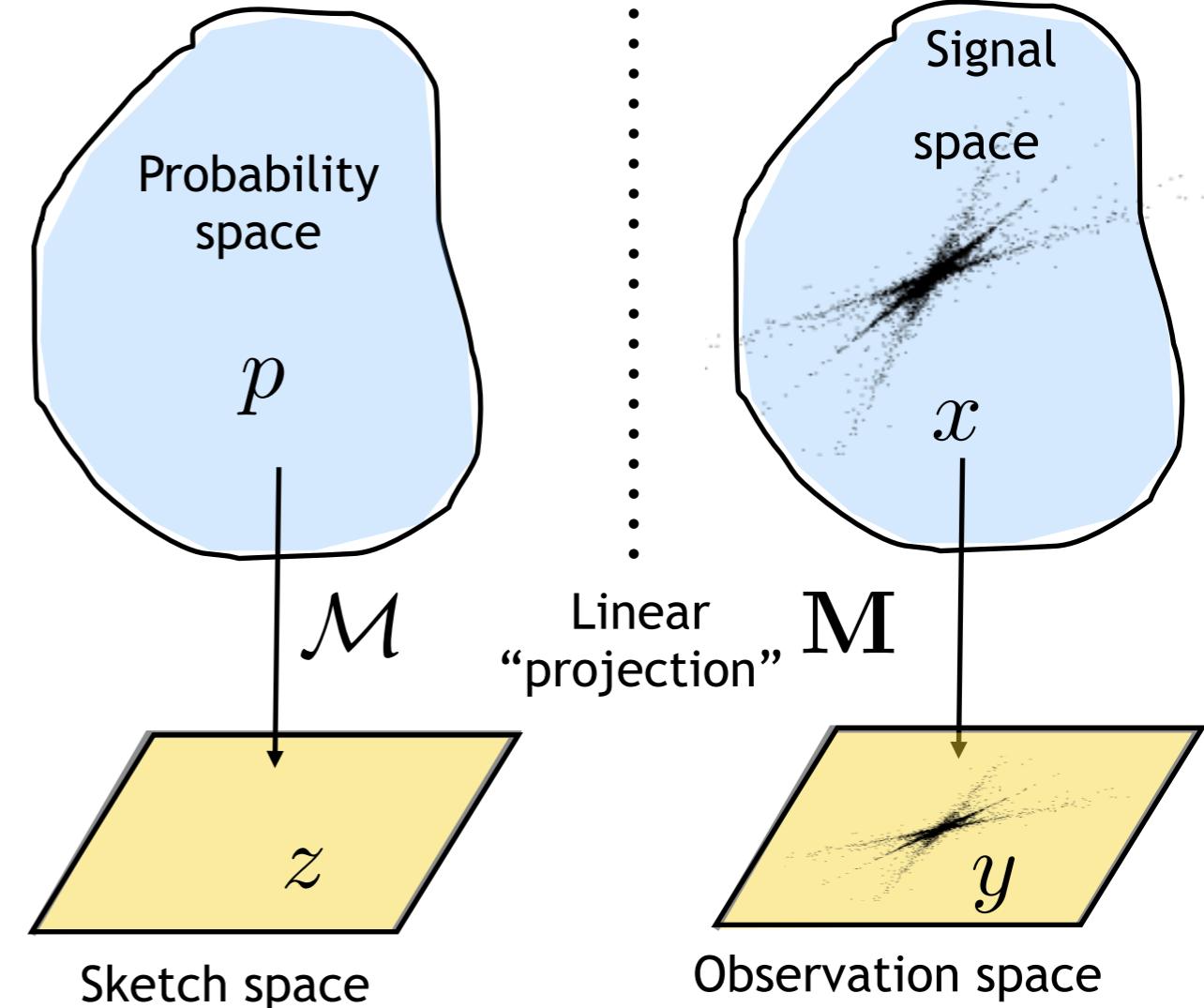
$$z_\ell = \frac{1}{N} \sum_{i=1}^N h_\ell(x_i)$$

$$\approx \mathbb{E} h_\ell(X)$$

$$= \int h_\ell(x)p(x)dx$$

- nonlinear in the feature vectors
- linear in the distribution $p(x)$

- Machine Learning
 - method of moments
 - compressive learning*

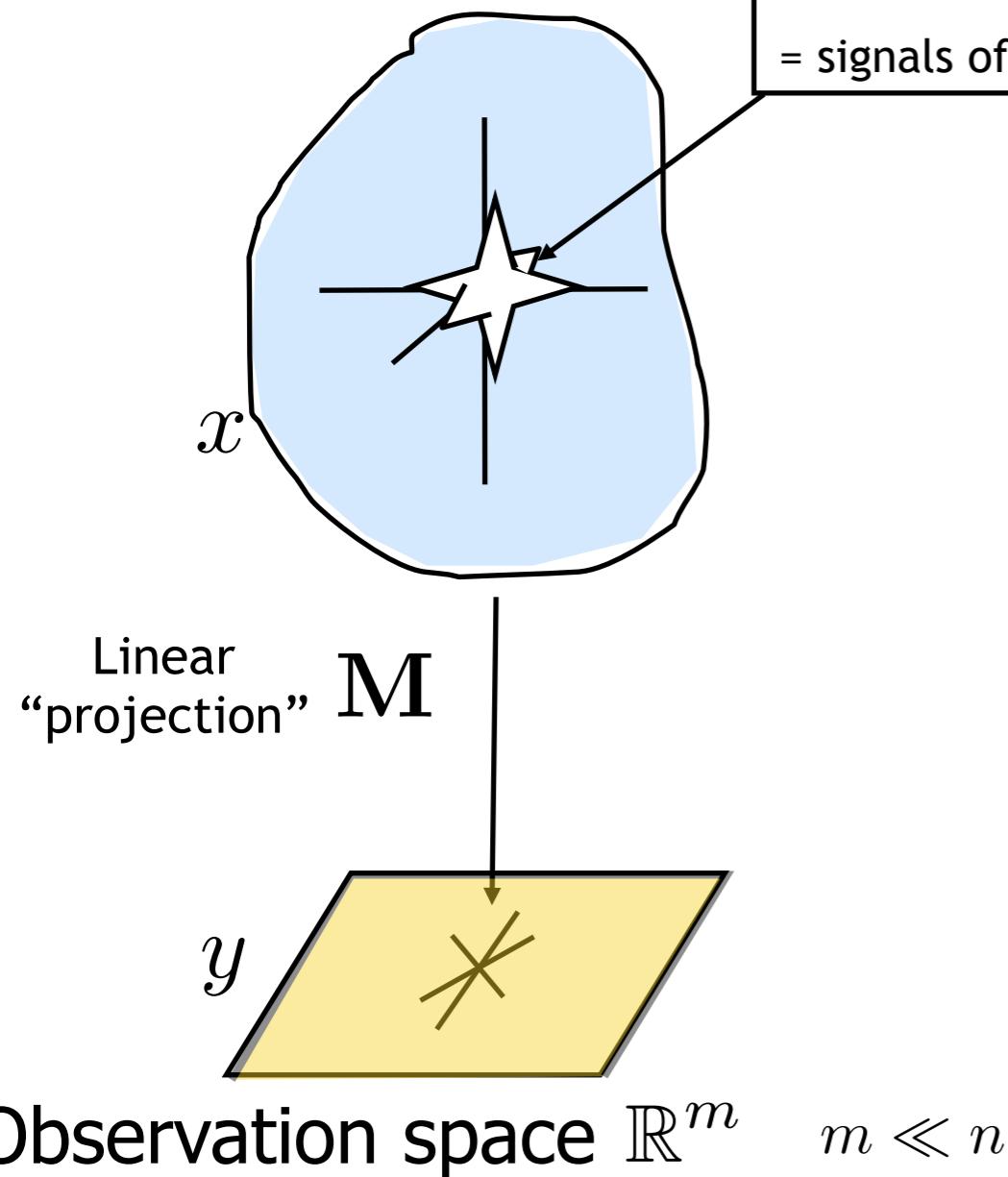




Information preserving projections

Stable recovery

Signal space \mathbb{R}^n

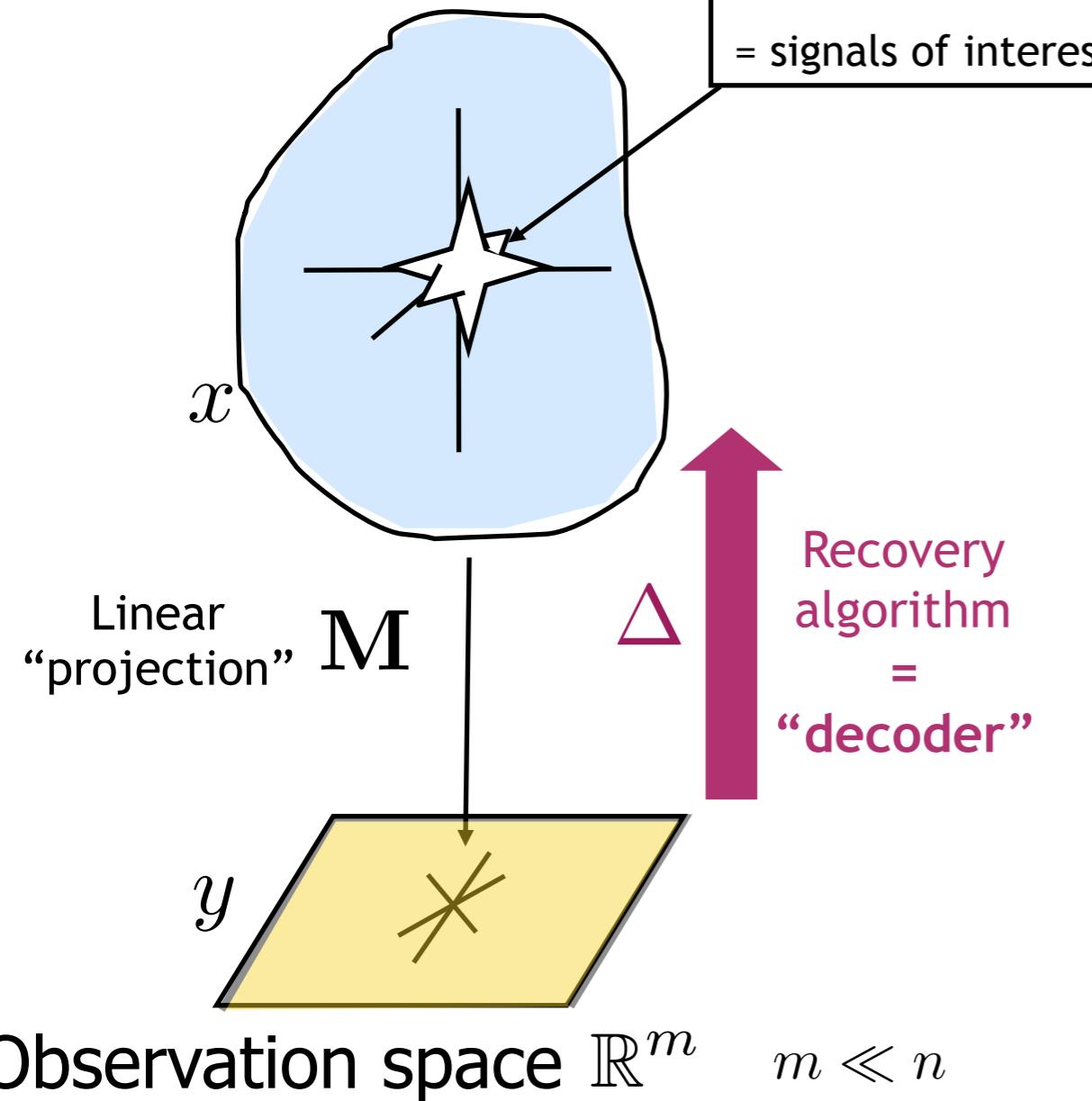


Ex: set of k-sparse vectors

$$\Sigma_k = \{x \in \mathbb{R}^n, \|x\|_0 \leq k\}$$

Stable recovery

Signal space \mathbb{R}^n



Ex: set of k-sparse vectors
 $\Sigma_k = \{x \in \mathbb{R}^n, \|x\|_0 \leq k\}$

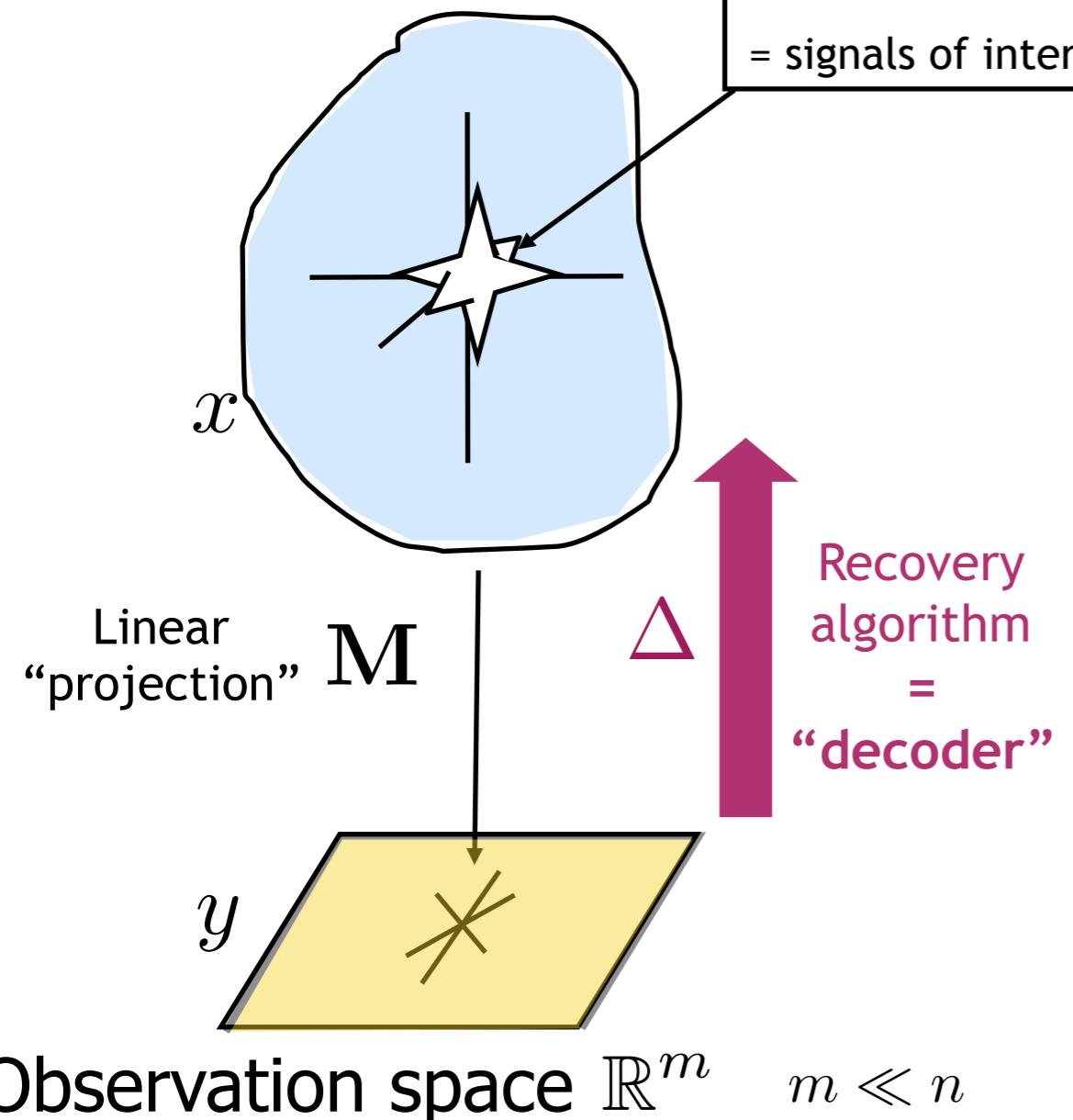
Ideal goal: build decoder Δ with the guarantee that

$$\|x - \Delta(Mx + e)\| \leq C\|e\|, \forall x \in \Sigma$$

(instance optimality [Cohen & al 2009])

Stable recovery

Signal space \mathbb{R}^n



Ex: set of k-sparse vectors
 $\Sigma_k = \{x \in \mathbb{R}^n, \|x\|_0 \leq k\}$

Ideal goal: build decoder Δ with the guarantee that

$$\|x - \Delta(Mx + e)\| \leq C\|e\|, \forall x \in \Sigma$$

(instance optimality [Cohen & al 2009])

Are there such decoders?

Stable recovery of k-sparse vectors

■ Typical decoders

■ L1 minimization

- LASSO [Tibshirani 1994], Basis Pursuit [Chen & al 1999]

$$\Delta(y) := \arg \min_{x: \mathbf{M}x=y} \|x\|_1$$

■ Greedy algorithms

- (Orthonormal) Matching Pursuit [Mallat & Zhang 1993],
- Iterative Hard Thresholding (IHT) [Blumensath & Davies 2009],
- ...

■ Guarantees

■ Assume Restricted isometry property

[Candès & al 2004]

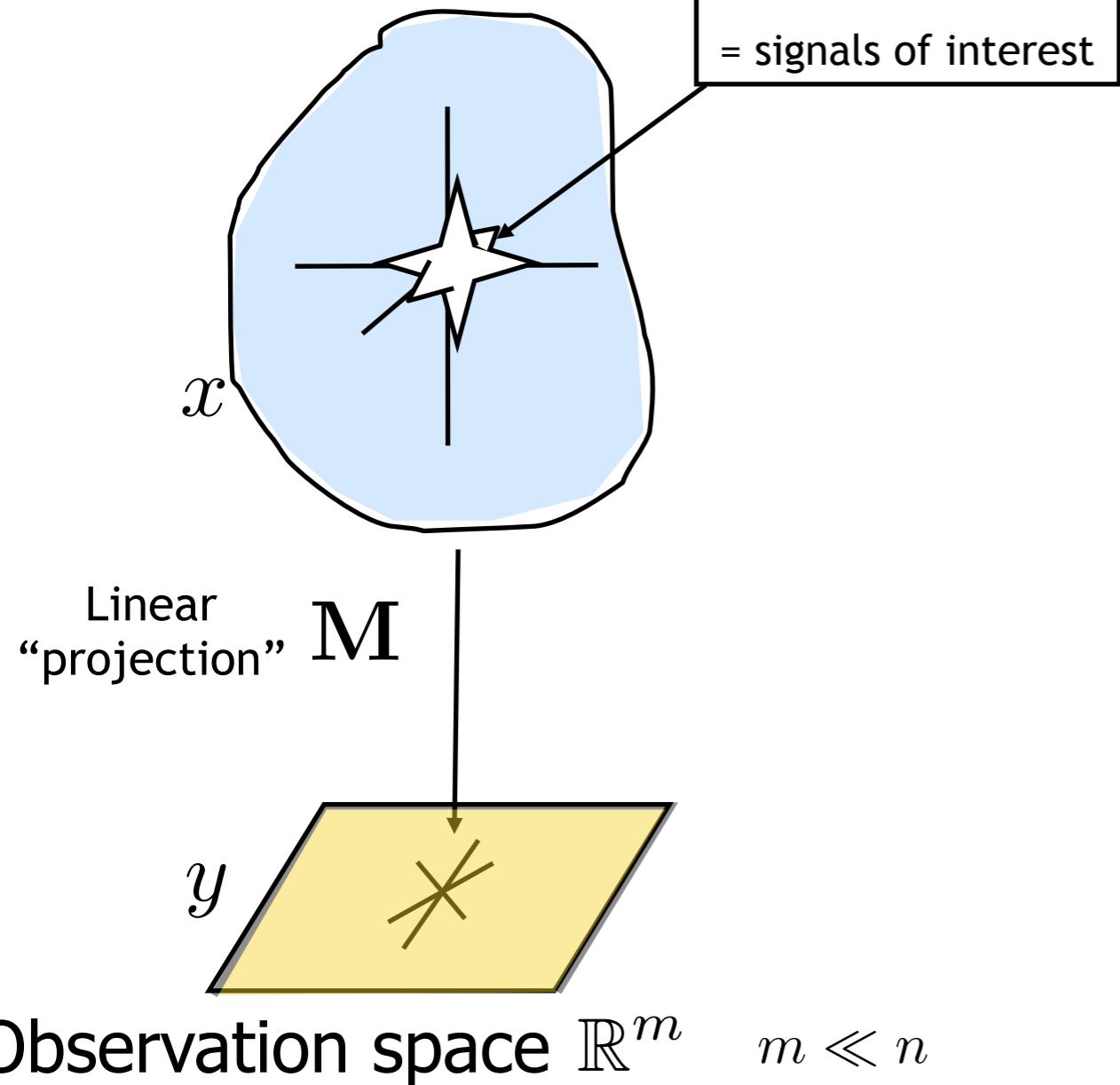
- Exact recovery
- Stability to noise
- Robustness to model error

$$1 - \delta \leq \frac{\|\mathbf{M}z\|_2^2}{\|z\|_2^2} \leq 1 + \delta$$

when $\|z\|_0 \leq 2k$

Stable recovery

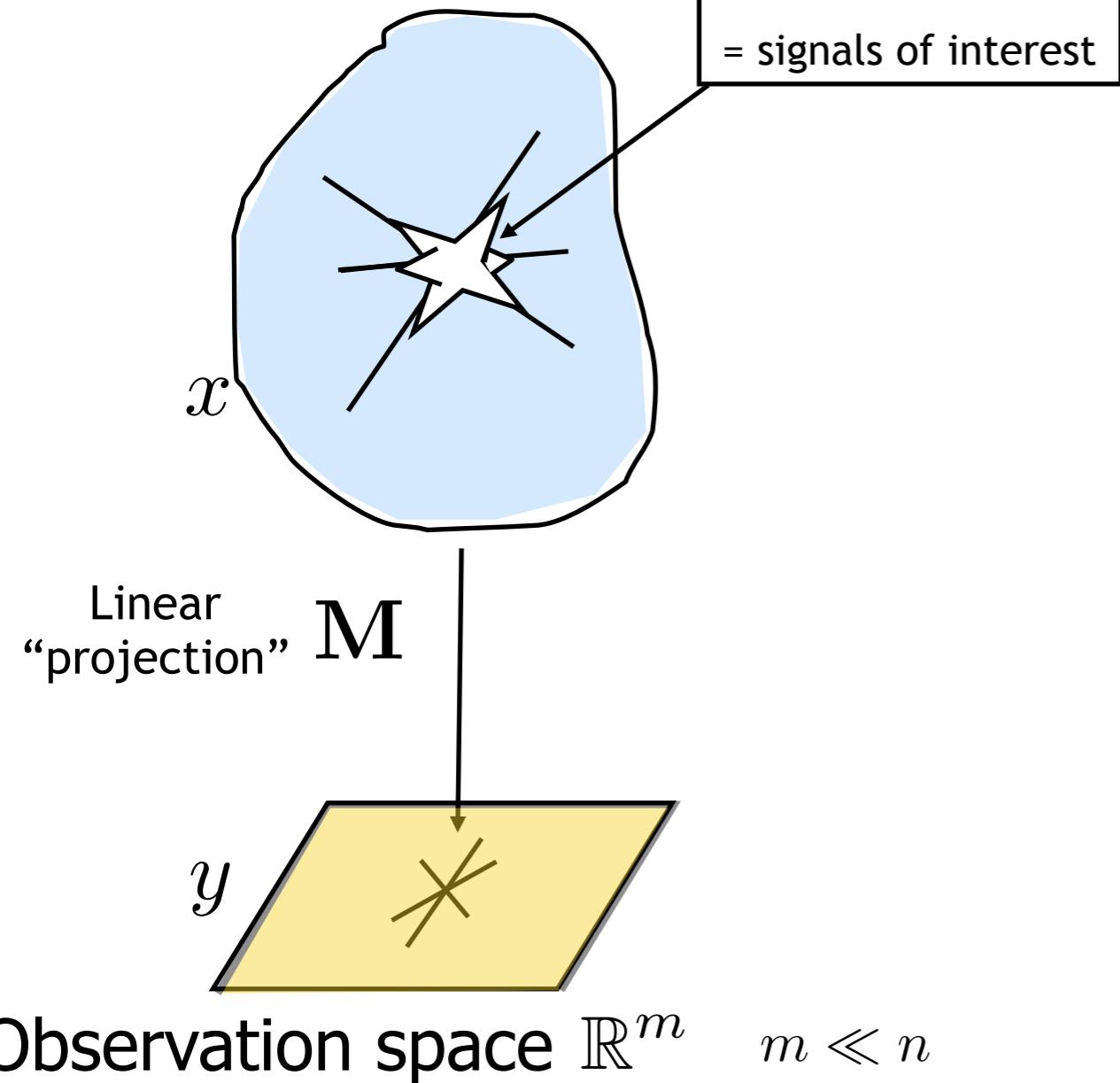
Signal space \mathbb{R}^n



- Low-dimensional model
- Sparse

Stable recovery

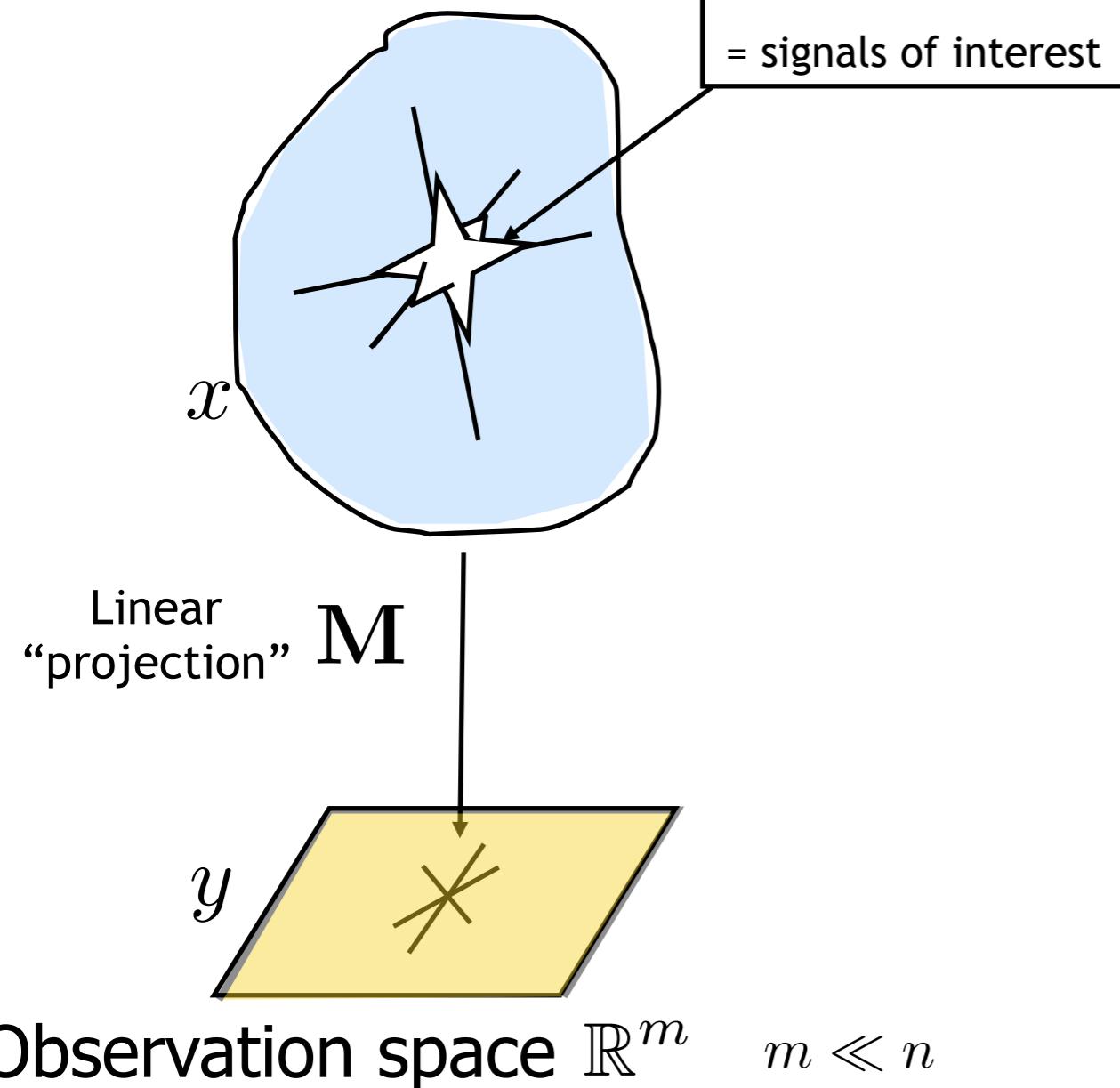
Signal space \mathbb{R}^n



- Low-dimensional model
 - Sparse
 - Sparse in dictionary \mathbf{D}

Stable recovery

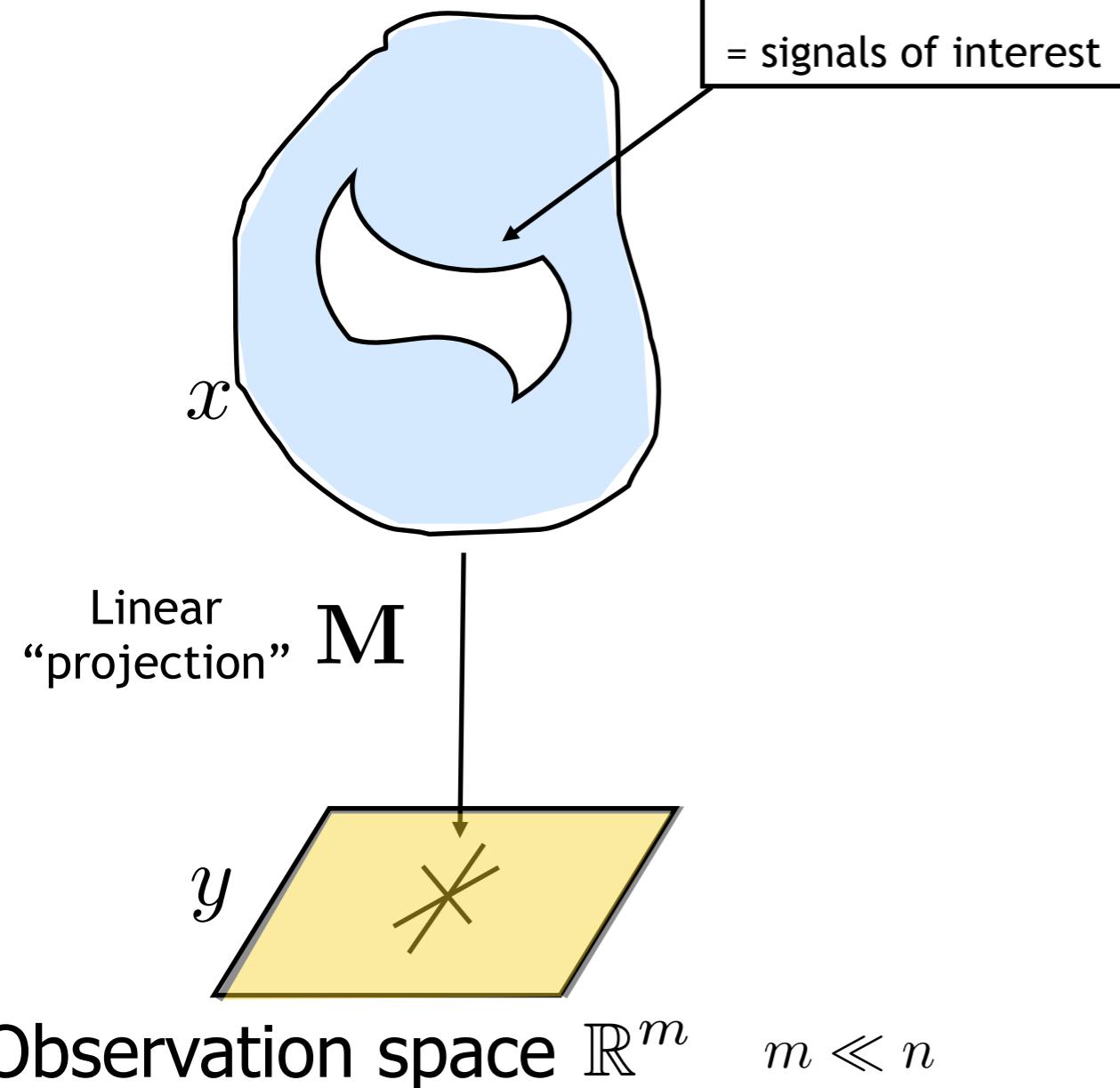
Signal space \mathbb{R}^n



- Low-dimensional model
 - Sparse
 - Sparse in dictionary D
 - Co-sparse in analysis operator A
 - \blacksquare *total variation,*
 - \blacksquare *physics-driven sparse models ..*

Stable recovery

Signal space \mathbb{R}^n

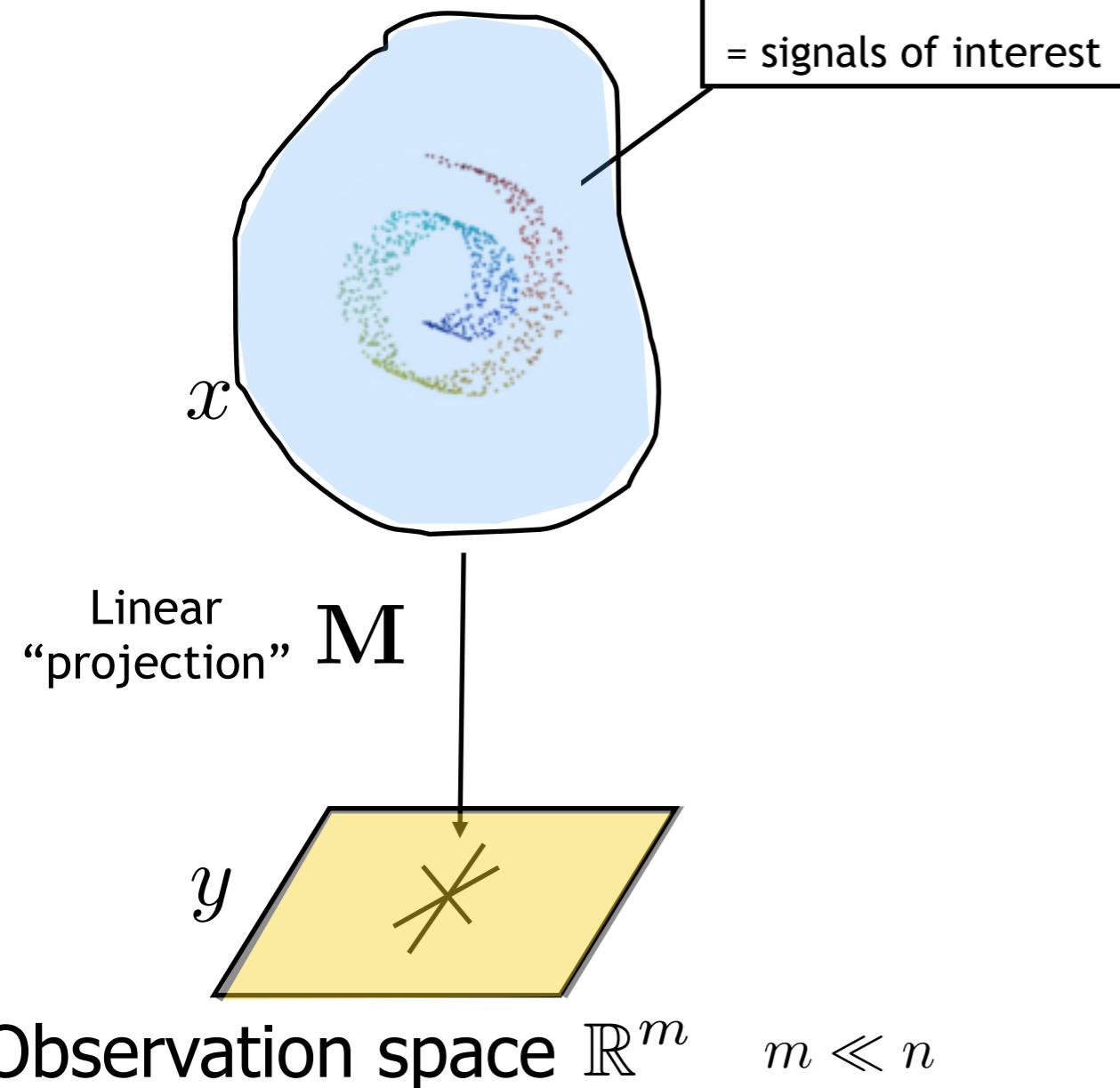


■ Low-dimensional model

- Sparse
- Sparse in dictionary D
- Co-sparse in analysis operator A
 - *total variation,*
 - *physics-driven sparse models ...*
- Low-rank matrix or tensor
 - *matrix completion,*
 - *phase-retrieval,*
 - *blind sensor calibration ...*

Stable recovery

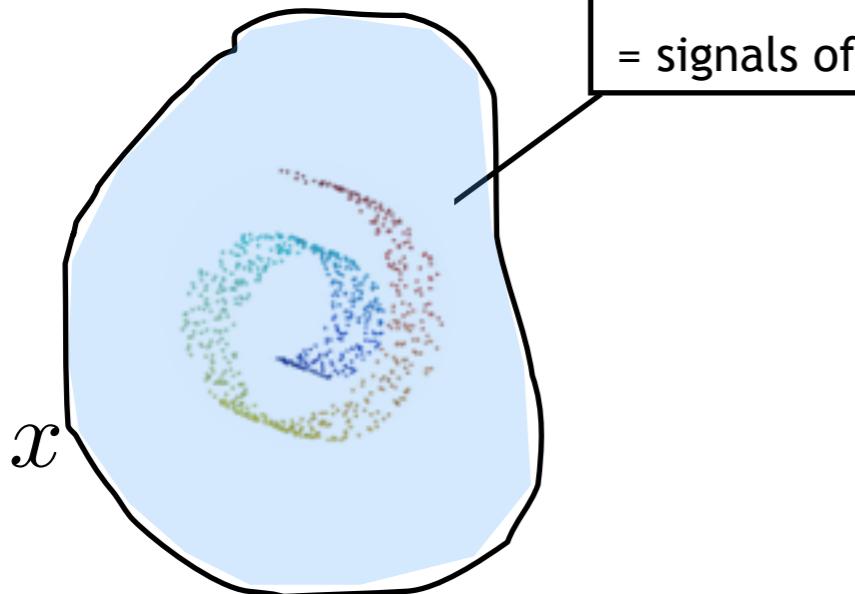
Signal space \mathbb{R}^n



- Low-dimensional model
 - Sparse
 - Sparse in dictionary D
 - Co-sparse in analysis operator A
 - *total variation,*
 - *physics-driven sparse models ...*
 - Low-rank matrix or tensor
 - *matrix completion,*
 - *phase-retrieval,*
 - *blind sensor calibration ...*
 - Manifold / Union of manifolds
 - *detection, estimation,*
 - *localization, mapping ...*
 - Matrix with sparse inverse
 - *Gaussian graphical models*
 - Given point cloud
 - *database indexing*

Stable recovery

Vector space \mathcal{H}
~~Signal space \mathbb{R}^n~~



Model set Σ
= signals of interest

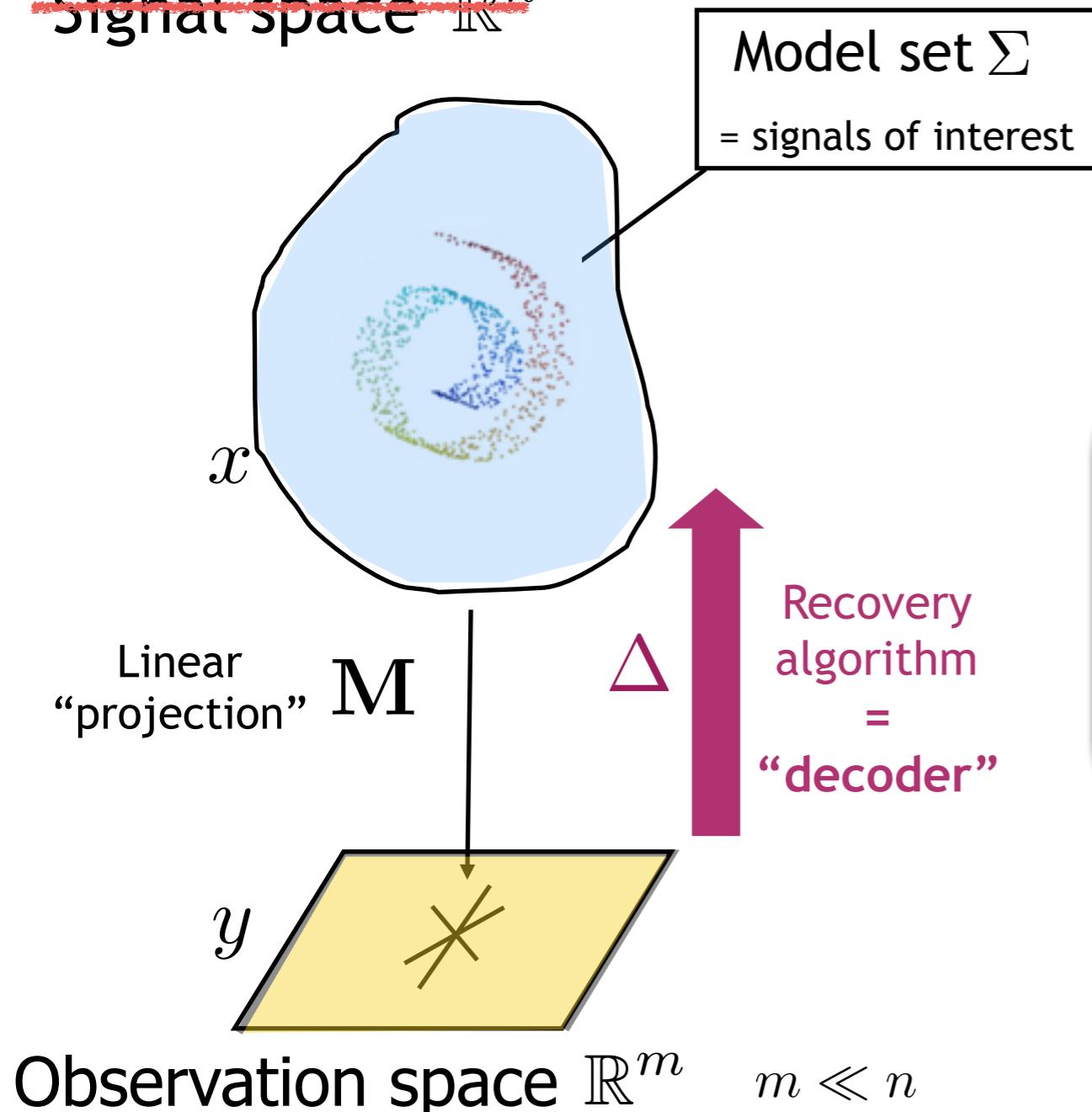
$m \ll n$

Observation space \mathbb{R}^m

- Low-dimensional model
 - Sparse
 - Sparse in dictionary D
 - Co-sparse in analysis operator A
 - *total variation,*
 - *physics-driven sparse models ...*
 - Low-rank matrix or tensor
 - *matrix completion,*
 - *phase-retrieval,*
 - *blind sensor calibration ...*
 - Manifold / Union of manifolds
 - *detection, estimation,*
 - *localization, mapping ...*
 - Matrix with sparse inverse
 - *Gaussian graphical models*
 - Given point cloud
 - *database indexing*
 - *Gaussian Mixture Model...*

General stable recovery

Vector space \mathcal{H}
~~Signal space \mathbb{R}^n~~



■ Low-dimensional model

■ *arbitrary* set $\Sigma \subset \mathcal{H}$

Ideal goal: build decoder Δ with the guarantee that

$$\|x - \Delta(\mathbf{M}x + e)\| \leq C\|e\|, \forall x \in \Sigma$$

(*instance optimality* [Cohen & al 2009])

Are there such decoders?

Stable recovery from arbitrary model sets

■ Theorem 1: RIP is necessary

- **Definition:** (general) Restricted Isometry Property (RIP) on *secant set*

$$\alpha \leq \frac{\|\mathbf{M}z\|}{\|z\|} \leq \beta \text{ when } z \in \Sigma - \Sigma := \{x - x', x, x' \in \Sigma\}$$

up to renormalization

$$\alpha = \sqrt{1 - \delta}; \beta = \sqrt{1 + \delta}$$

- RIP holds as soon as *there exists* an instance optimal decoder

Stable recovery from arbitrary model sets

■ Theorem 1: RIP is necessary

- **Definition:** (general) Restricted Isometry Property (RIP) on *secant set*

$$\alpha \leq \frac{\|\mathbf{M}z\|}{\|z\|} \leq \beta \text{ when } z \in \Sigma - \Sigma := \{x - x', x, x' \in \Sigma\}$$

up to renormalization
 $\alpha = \sqrt{1 - \delta}; \beta = \sqrt{1 + \delta}$

- RIP holds as soon as *there exists* an instance optimal decoder

■ Theorem 2: RIP is sufficient

- RIP implies *existence* of decoder with performance guarantees:

$$\|x - \Delta(\mathbf{M}x + e)\| \leq C(\delta)\|e\|$$

- Exact recovery
- Stable to noise

[Cohen & al 2009] for Σ_k

[Bourrier & al 2014] for arbitrary Σ

Stable recovery from arbitrary model sets

■ Theorem 1: RIP is necessary

- **Definition:** (general) Restricted Isometry Property (RIP) on *secant set*

$$\alpha \leq \frac{\|\mathbf{M}z\|}{\|z\|} \leq \beta \text{ when } z \in \Sigma - \Sigma := \{x - x', x, x' \in \Sigma\}$$

up to renormalization
 $\alpha = \sqrt{1 - \delta}; \beta = \sqrt{1 + \delta}$

- RIP holds as soon as *there exists* an instance optimal decoder

■ Theorem 2: RIP is sufficient

- RIP implies *existence* of decoder with performance guarantees:

$$\|x - \Delta(\mathbf{M}x + e)\| \leq C(\delta)\|e\| + C'(\delta)d_{\Sigma}(x, \Sigma)$$

- Exact recovery
- Stable to noise
- Bonus: robust to model error

[Cohen & al 2009] for Σ_k

[Bourrier & al 2014] for arbitrary Σ

Stable recovery from arbitrary model sets

■ Theorem 1: RIP is necessary

- **Definition:** (general) Restricted Isometry Property (RIP) on *secant set*

$$\alpha \leq \frac{\|\mathbf{M}z\|}{\|z\|} \leq \beta \text{ when } z \in \Sigma - \Sigma := \{x - x', x, x' \in \Sigma\}$$

up to renormalization
 $\alpha = \sqrt{1 - \delta}; \beta = \sqrt{1 + \delta}$

- RIP holds as soon as *there exists* an instance optimal decoder

■ Theorem 2: RIP is sufficient

- RIP implies *existence* of decoder with performance guarantees:

$$\|x - \Delta(\mathbf{M}x + e)\| \leq C(\delta)\|e\| + C'(\delta)d_{\Sigma}(x, \Sigma)$$

- Exact recovery
- Stable to noise
- Bonus: robust to model error

[Cohen & al 2009] for Σ_k

[Bourrier & al 2014] for arbitrary Σ

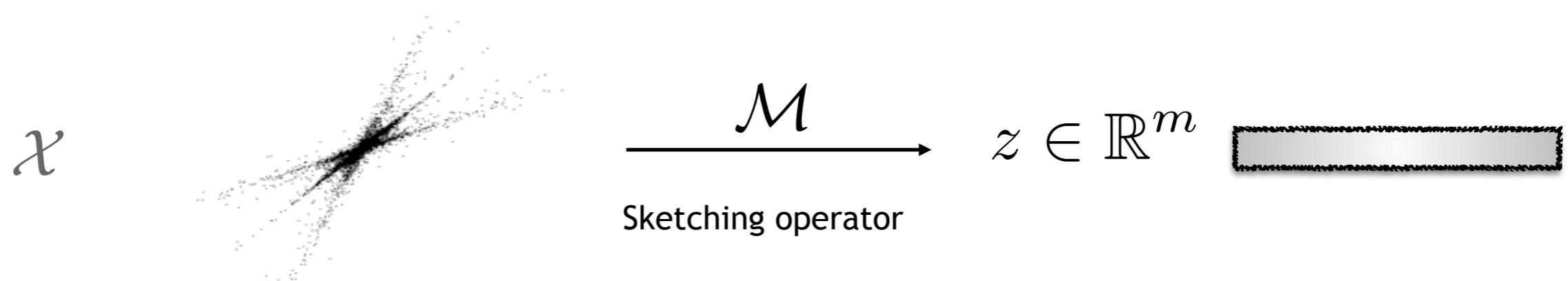
Distance to model set



Compressive Learning Examples

Compressive Machine Learning

- Point cloud = empirical probability distribution



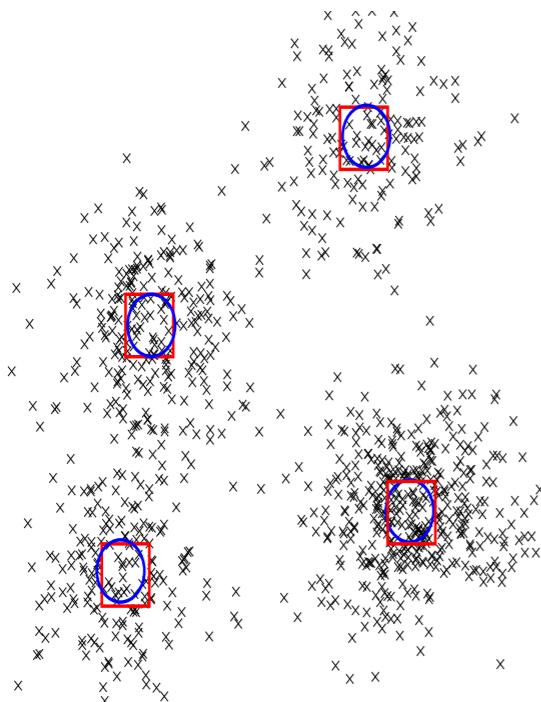
- Reduce collection dimension = **sketching**

$$z_\ell = \frac{1}{N} \sum_{i=1}^N h_\ell(x_i) \quad 1 \leq \ell \leq m$$

Choosing information preserving sketch ?

Example: Compressive Clustering

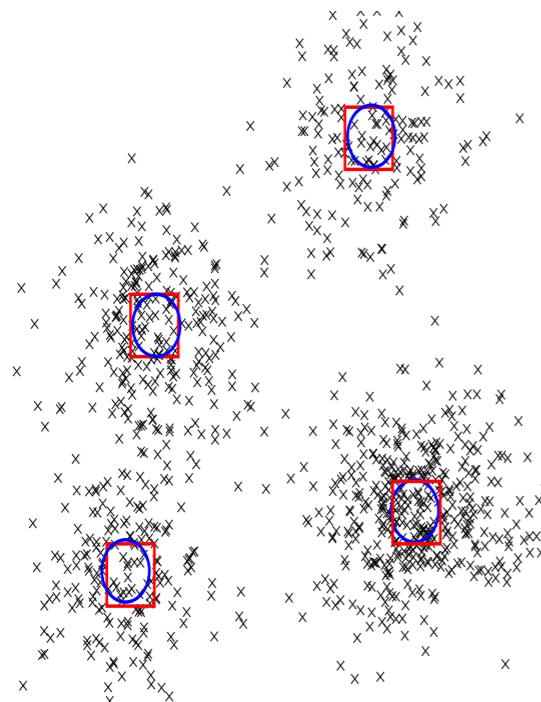
■ Goal: find k centroids



■ Standard approach = K-means

Example: Compressive Clustering

■ Goal: find k centroids



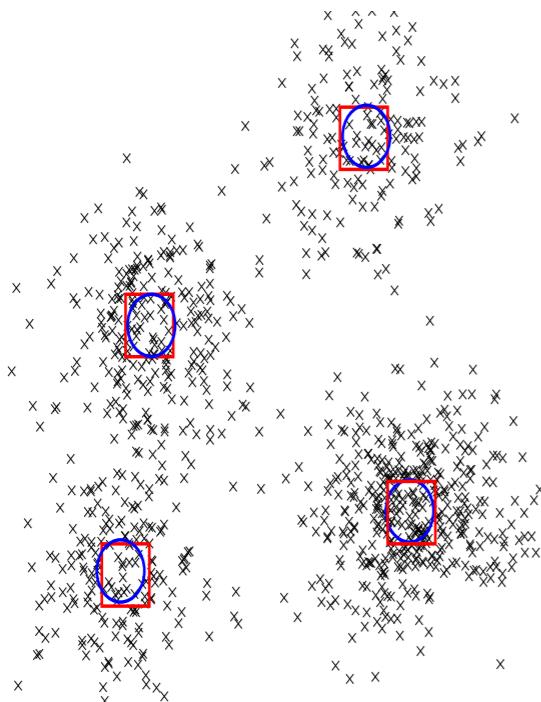
■ Sketching approach

- $p(x)$ is spatially localized
 - ▶ need “incoherent” sampling
 - ▶ choose Fourier sampling

■ Standard approach = K-means

Example: Compressive Clustering

■ Goal: find k centroids



■ Sketching approach

- $p(x)$ is spatially localized
 - ▶ need “incoherent” sampling
 - ▶ choose Fourier sampling
- sample characteristic function

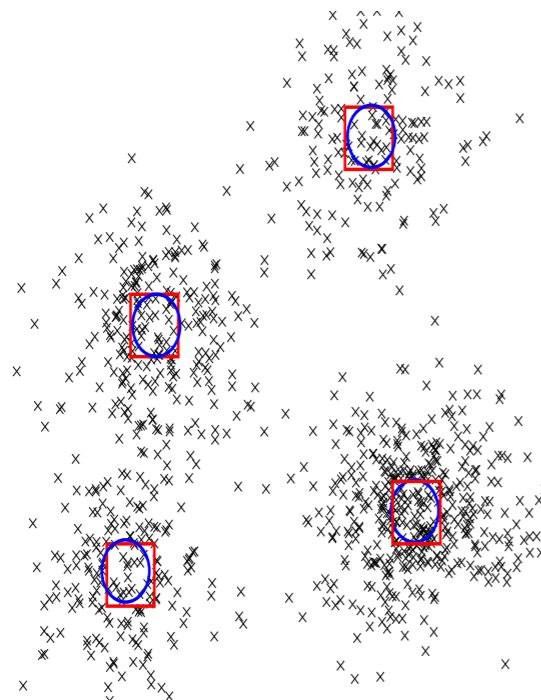
$$z_\ell = \frac{1}{N} \sum_{i=1}^N e^{j w_\ell^\top x_i}$$

■ Standard approach = K-means

- ▶ choose *sampling frequencies*
 $w_\ell \in \mathbb{R}^n$

Example: Compressive Clustering

■ Goal: find k centroids



■ Sketching approach

- $p(x)$ is spatially localized
 - ▶ need “incoherent” sampling
 - ▶ choose Fourier sampling
- sample characteristic function

$$z_\ell = \frac{1}{N} \sum_{i=1}^N e^{j w_\ell^\top x_i}$$

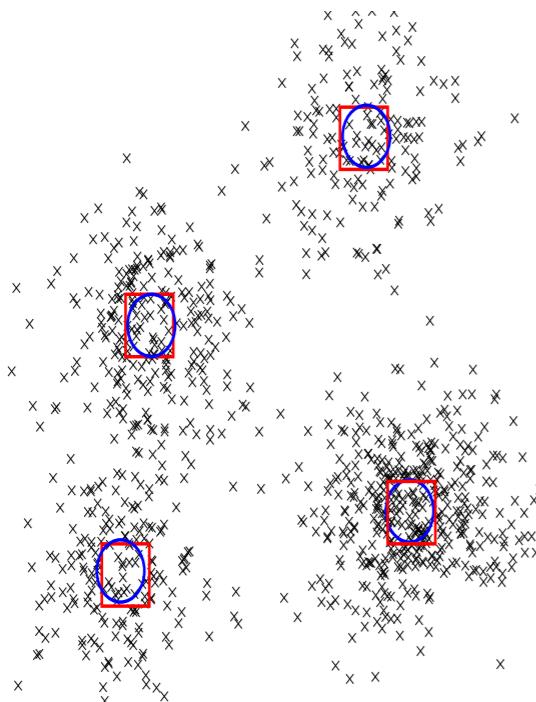
■ Standard approach = K-means

▶ choose *sampling frequencies*
 $w_\ell \in \mathbb{R}^n$

How ? see poster N. Keriven

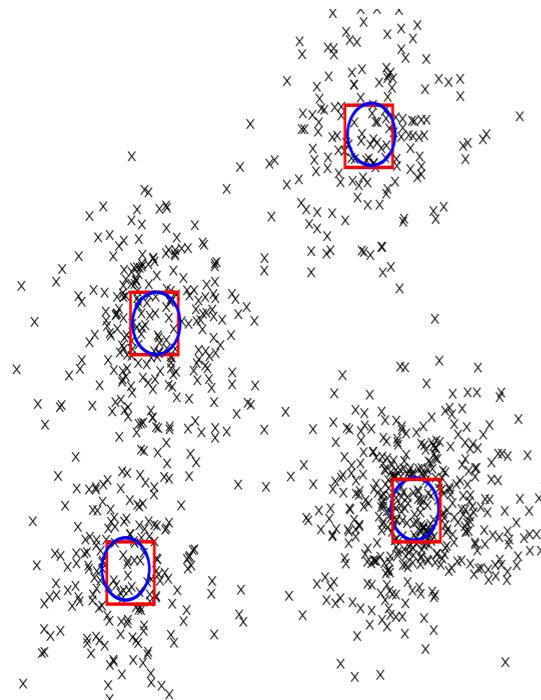
Example: Compressive Clustering

■ Goal: find k centroids



Example: Compressive Clustering

■ Goal: find k centroids



*Density model=GMM
with variance = identity*

$$p \approx \sum_{k=1}^K \alpha_k p_{\theta_k}$$

ground truth

$$\chi \xrightarrow[\substack{\text{Sampled} \\ \text{Characteristic} \\ \text{Function}}]{\mathcal{M}} z \in \mathbb{R}^m$$

$N = 1000; n = 2$

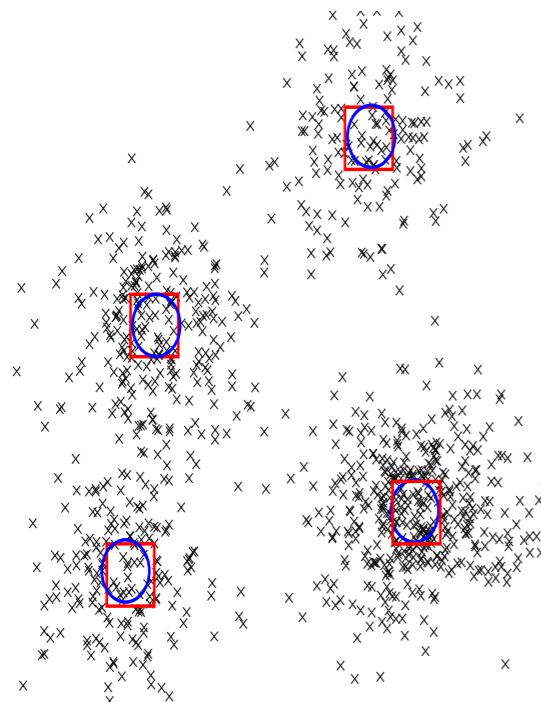
$m = 60$



$$z = \mathcal{M}p \approx \sum_{k=1}^K \alpha_k \mathcal{M}p_{\theta_k}$$

Example: Compressive Clustering

■ Goal: find k centroids



$\Delta \uparrow$
Recovery
algorithm
= “decoder”

$$\begin{matrix} \chi \\ N = 1000; n = 2 \end{matrix} \xrightarrow[\text{Sampled Characteristic Function}]{\mathcal{M}} z \in \mathbb{R}^m \quad m = 60$$

*Density model=GMM
with variance = identity*

$$p \approx \sum_{k=1}^K \alpha_k p_{\theta_k}$$

ground truth

□ estimated centroids

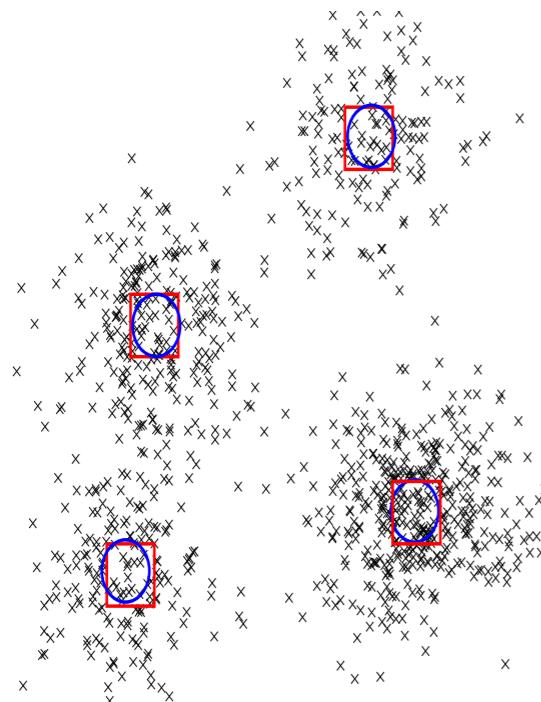
*inspired by
Iterative Hard Thresholding*



$$z = \mathcal{M}p \approx \sum_{k=1}^K \alpha_k \mathcal{M}p_{\theta_k}$$

Example: Compressive Clustering

■ Goal: find k centroids



Δ

Recovery
algorithm
= “decoder”

*Density model=GMM
with variance = identity*

$$p \approx \sum_{k=1}^K \alpha_k p_{\theta_k}$$

ground truth

□ estimated centroids

*inspired by
Iterative Hard Thresholding*

Compressive Hierarchical Splitting (CHS)
= extension to general GMM
similar to OMP with Replacement

$$\chi \xrightarrow[\text{Sampled Characteristic Function}]{\mathcal{M}} z \in \mathbb{R}^m$$

$N = 1000; n = 2$

$m = 60$



$$z = \mathcal{M}p \approx \sum_{k=1}^K \alpha_k \mathcal{M}p_{\theta_k}$$

Application: Speaker Verification Results (DET-curves)



~ 50 Gbytes
~ 1000 hours of speech

■ MFCC coefficients $x_i \in \mathbb{R}^{12}$

$N = 300\ 000\ 000$

Application: Speaker Verification Results (DET-curves)



~ 50 Gbytes
~ 1000 hours of speech

■ MFCC coefficients $x_i \in \mathbb{R}^{12}$

$$N = 300\ 000\ 000$$

■ After silence detection

$$N = 60\ 000\ 000$$

Application: Speaker Verification Results (DET-curves)



~ 50 Gbytes
~ 1000 hours of speech

■ MFCC coefficients $x_i \in \mathbb{R}^{12}$

$$N = 300\ 000\ 000$$

■ After silence detection

$$N = 60\ 000\ 000$$

■ Maximum size manageable by EM

$$N = 300\ 000$$

Application: Speaker Verification Results (DET-curves)



~ 50 Gbytes
~ 1000 hours of speech

■ MFCC coefficients $x_i \in \mathbb{R}^{12}$

$$N = 300\,000\,000$$

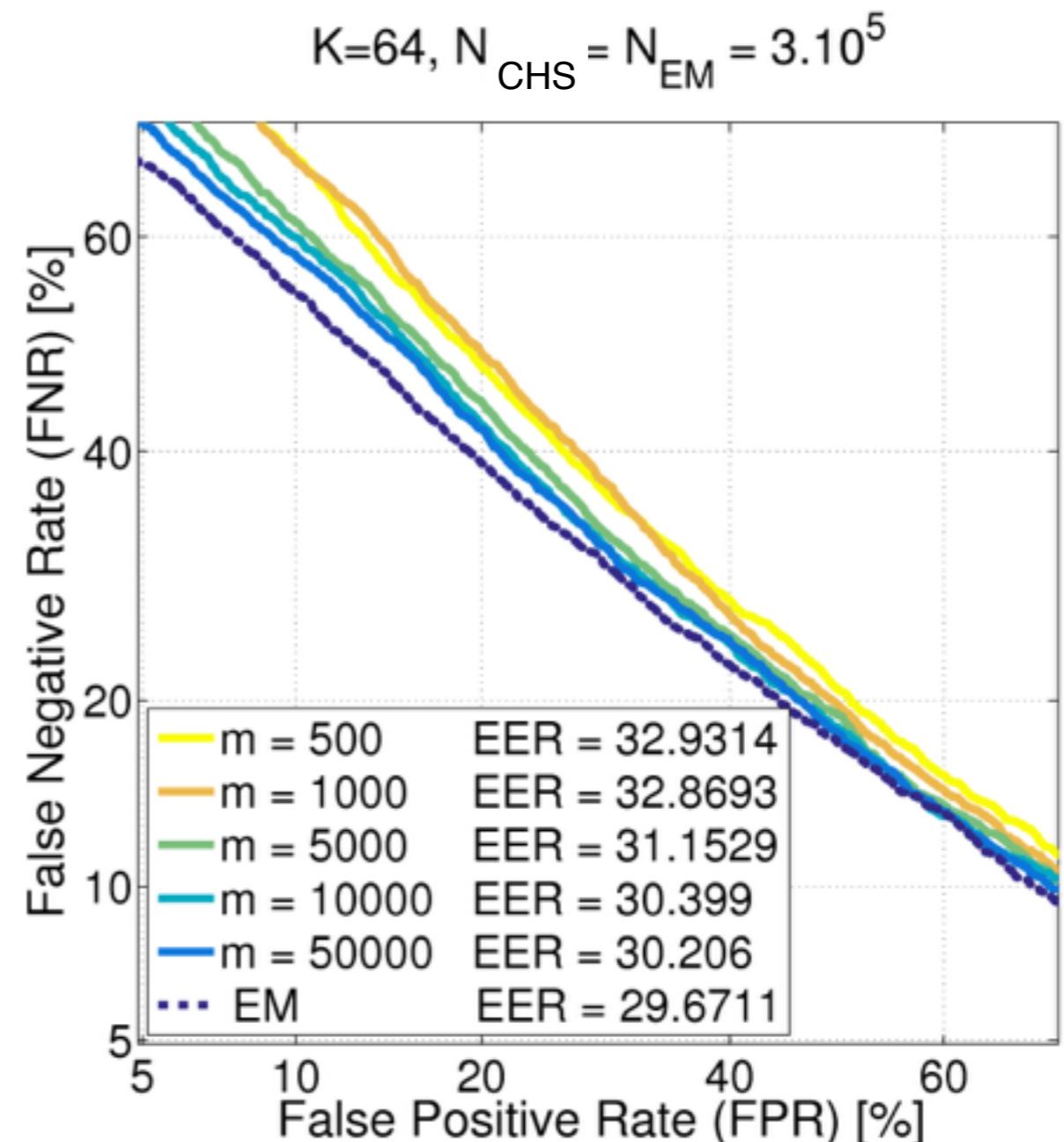
■ After silence detection

$$N = 60\,000\,000$$

■ Maximum size manageable by EM

$$N = 300\,000$$

for EM
for CHS



Application: Speaker Verification Results (DET-curves)



~ 50 Gbytes
~ 1000 hours of speech

■ MFCC coefficients $x_i \in \mathbb{R}^{12}$

$$N = 300\,000\,000$$

■ After silence detection

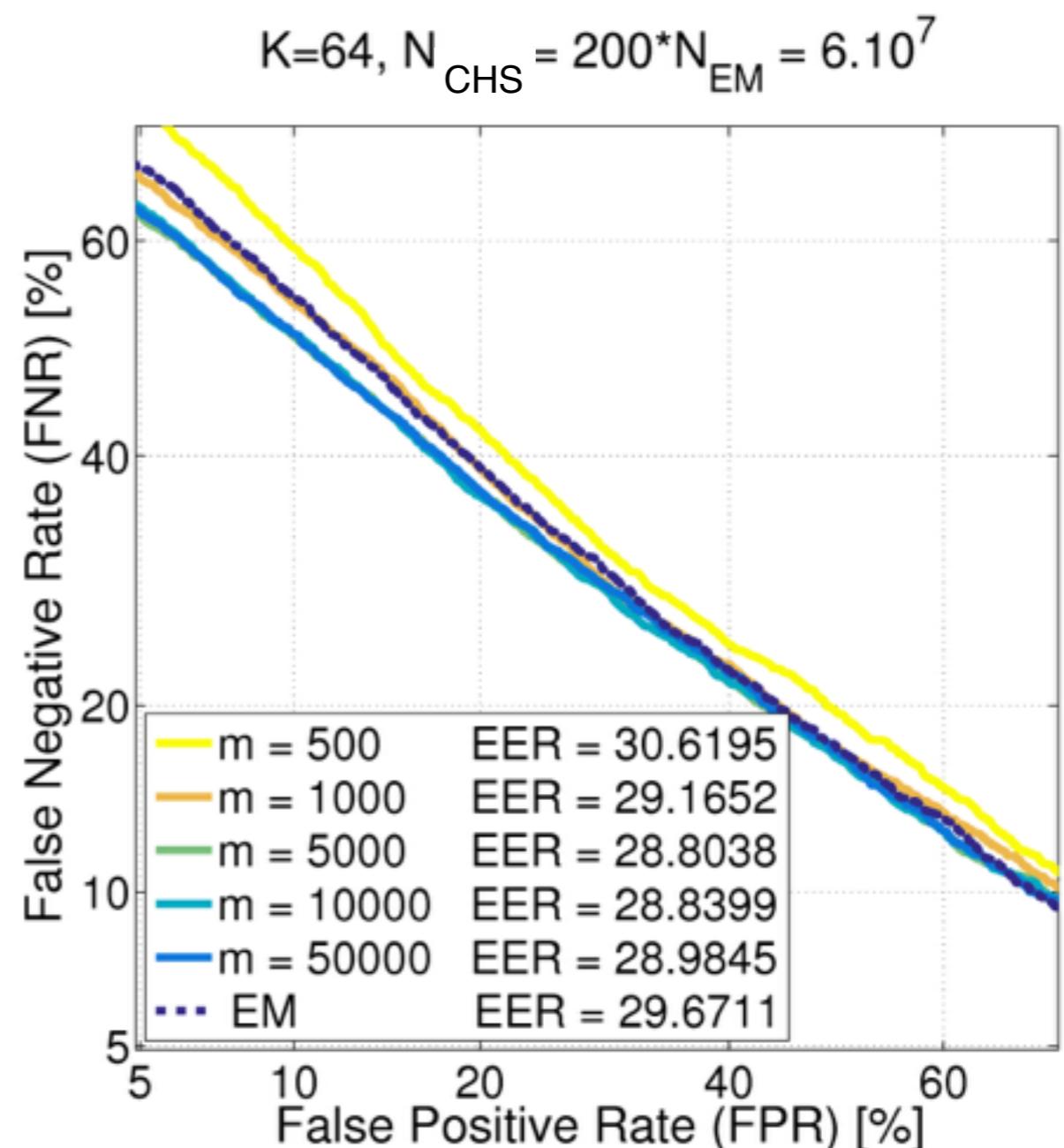
$$N = 60\,000\,000$$

for CHS

■ Maximum size manageable by EM

$$N = 300\,000$$

for EM

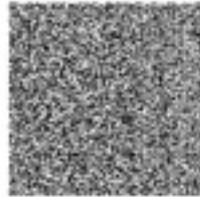


Application: Speaker Verification Results (DET-curves)



~ 50 Gbytes

~ 1000 hours of speech



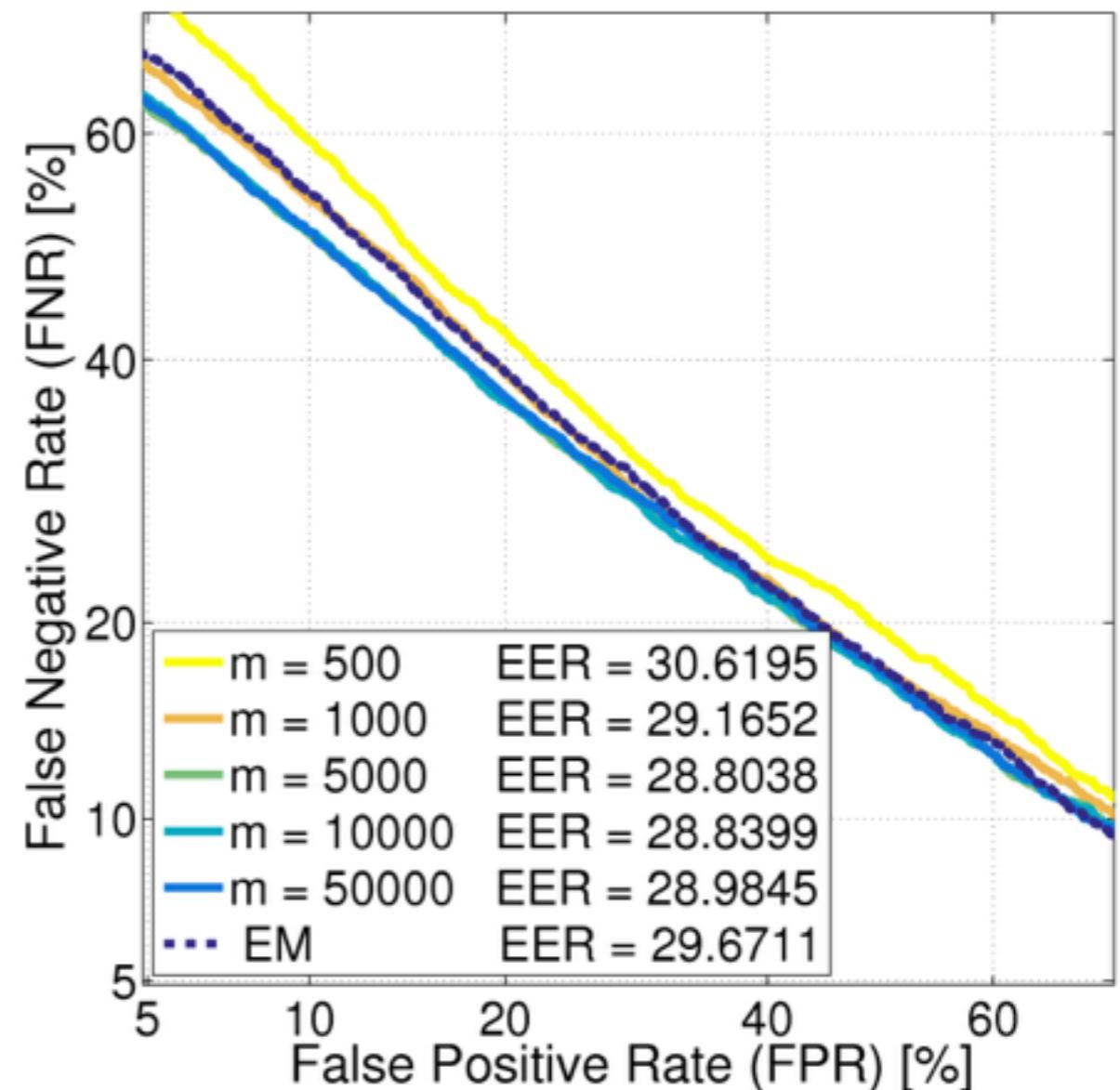
■ **m = 500**

■ close to EM

■ 7 200 000-fold compression

► one QR code 40-L

$$K=64, N_{CHS} = 200 * N_{EM} = 6 \cdot 10^7$$

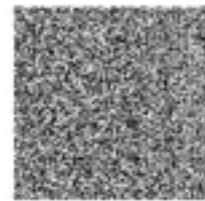


Application: Speaker Verification Results (DET-curves)



~ 50 Gbytes

~ 1000 hours of speech

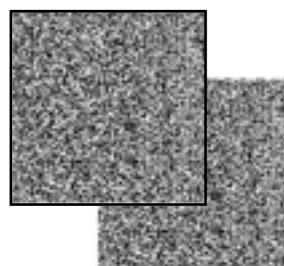


m = 500

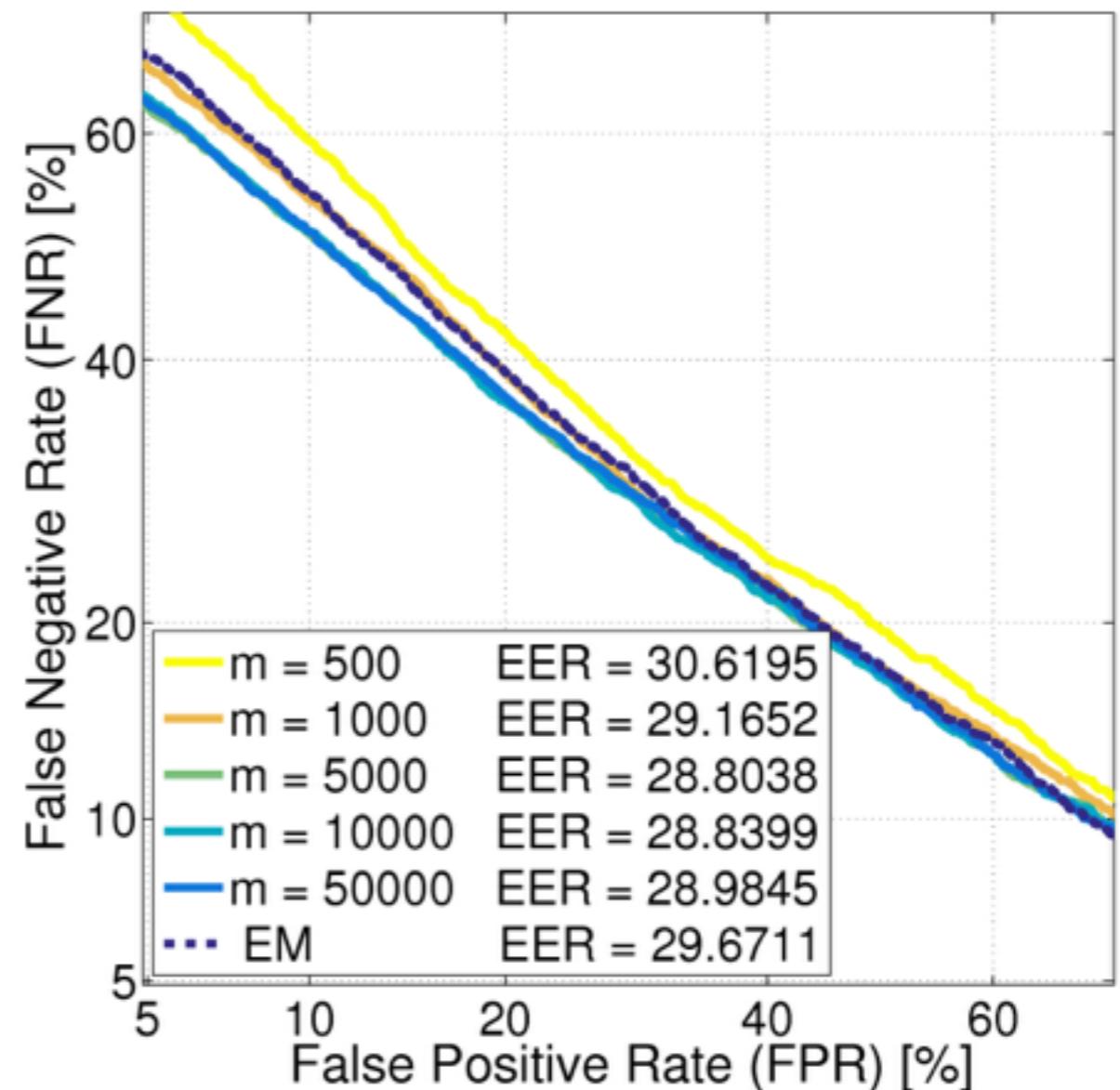
- close to EM
- 7 200 000-fold compression
- one QR code 40-L

m = 1000

- same as EM
- 3 600 000-fold compression
- two QR codes 40-L



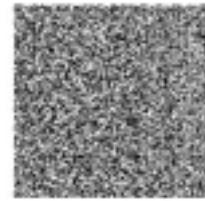
$$K=64, N_{\text{CHS}} = 200 \cdot N_{\text{EM}} = 6 \cdot 10^7$$



Application: Speaker Verification Results (DET-curves)



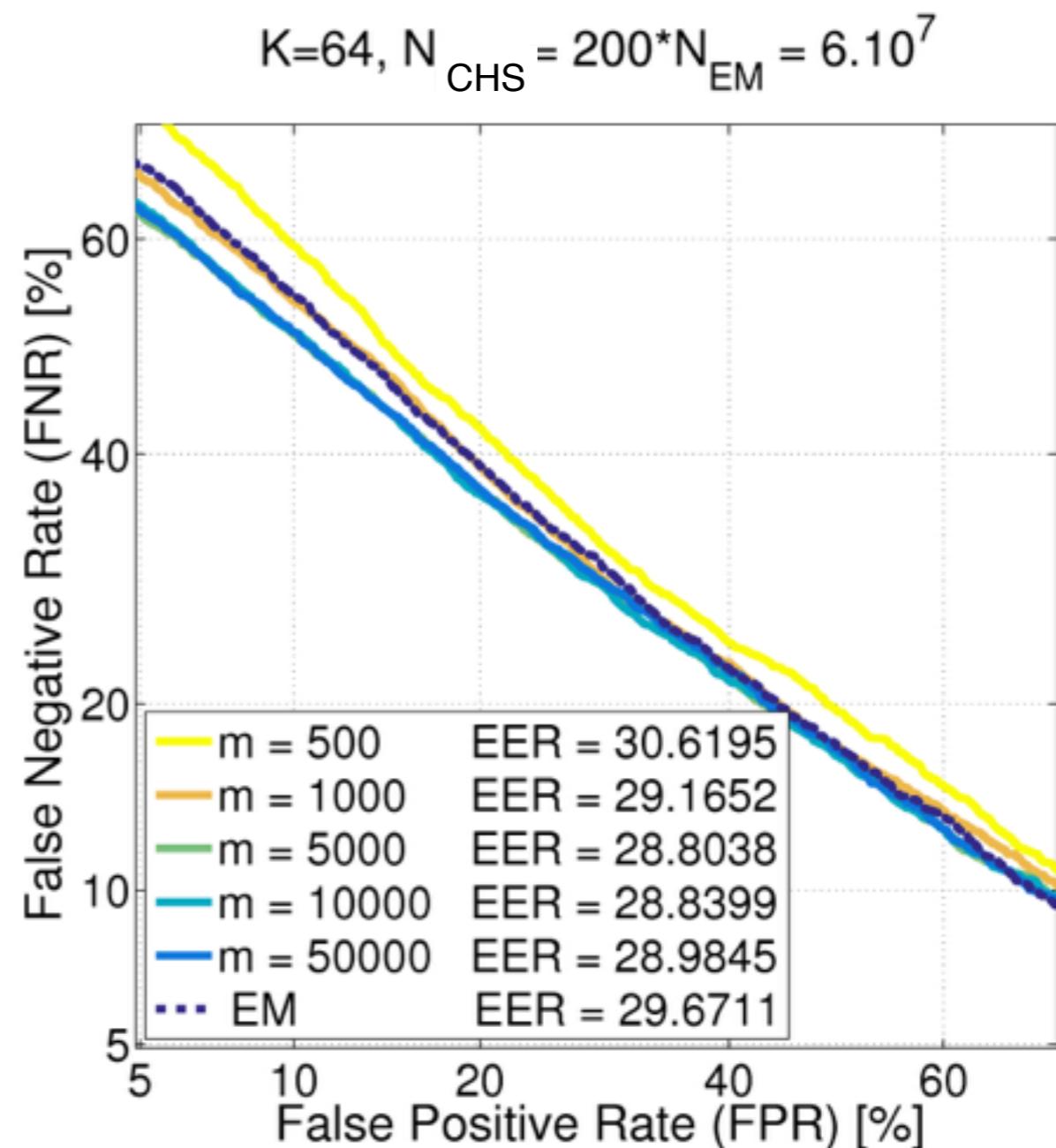
~ 50 Gbytes
~ 1000 hours of speech



m= 500
close to EM
7 200 000-fold compression
one QR code 40-L

m= 1000
same as EM
3 600 000-fold compression
two QR codes 40-L

m= 5 000
better than EM
exploit whole collection
720 000-fold compression
fit 80 on 3"1/2 floppy disk





Computational Efficiency

Computational Aspects

■ Sketching

- empirical characteristic function

$$z_\ell = \frac{1}{N} \sum_{i=1}^N e^{j w_\ell^\top x_i}$$

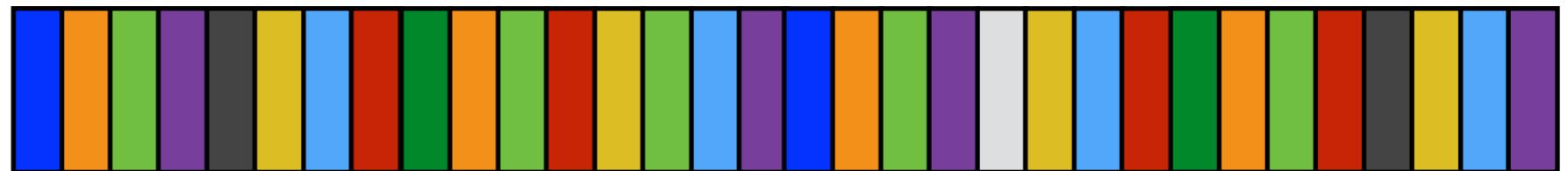
Computational Aspects

■ Sketching

- empirical characteristic function

$$z_\ell = \frac{1}{N} \sum_{i=1}^N e^{j w_\ell^\top x_i}$$

X

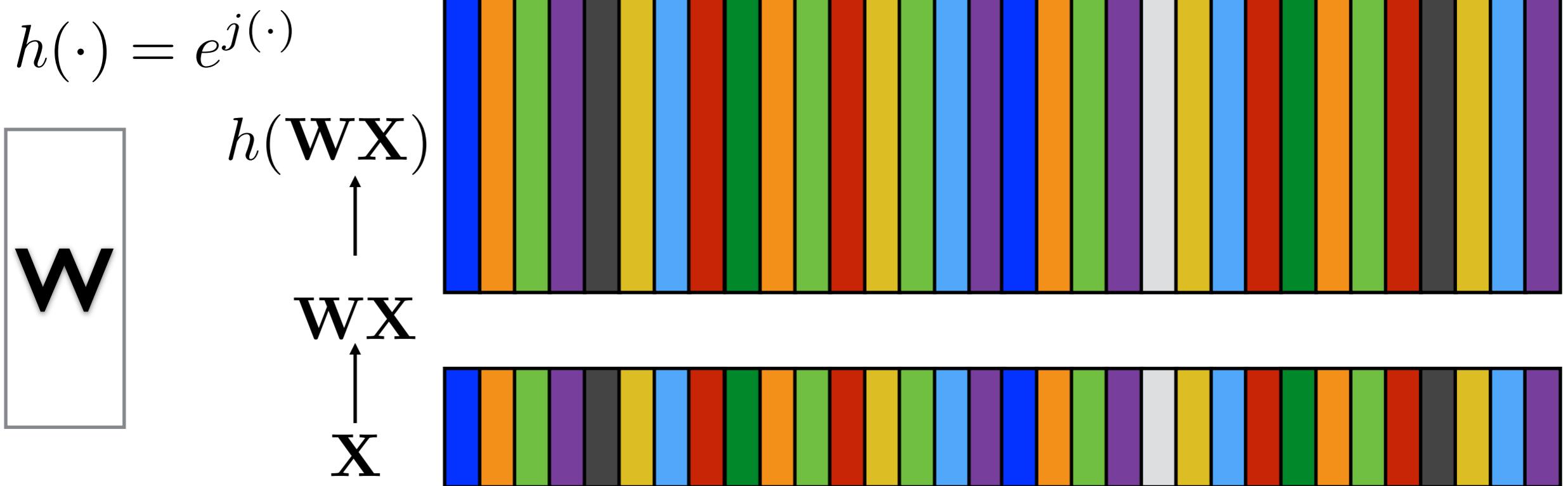


Computational Aspects

■ Sketching

- empirical characteristic function

$$z_\ell = \frac{1}{N} \sum_{i=1}^N e^{j w_\ell^\top x_i}$$

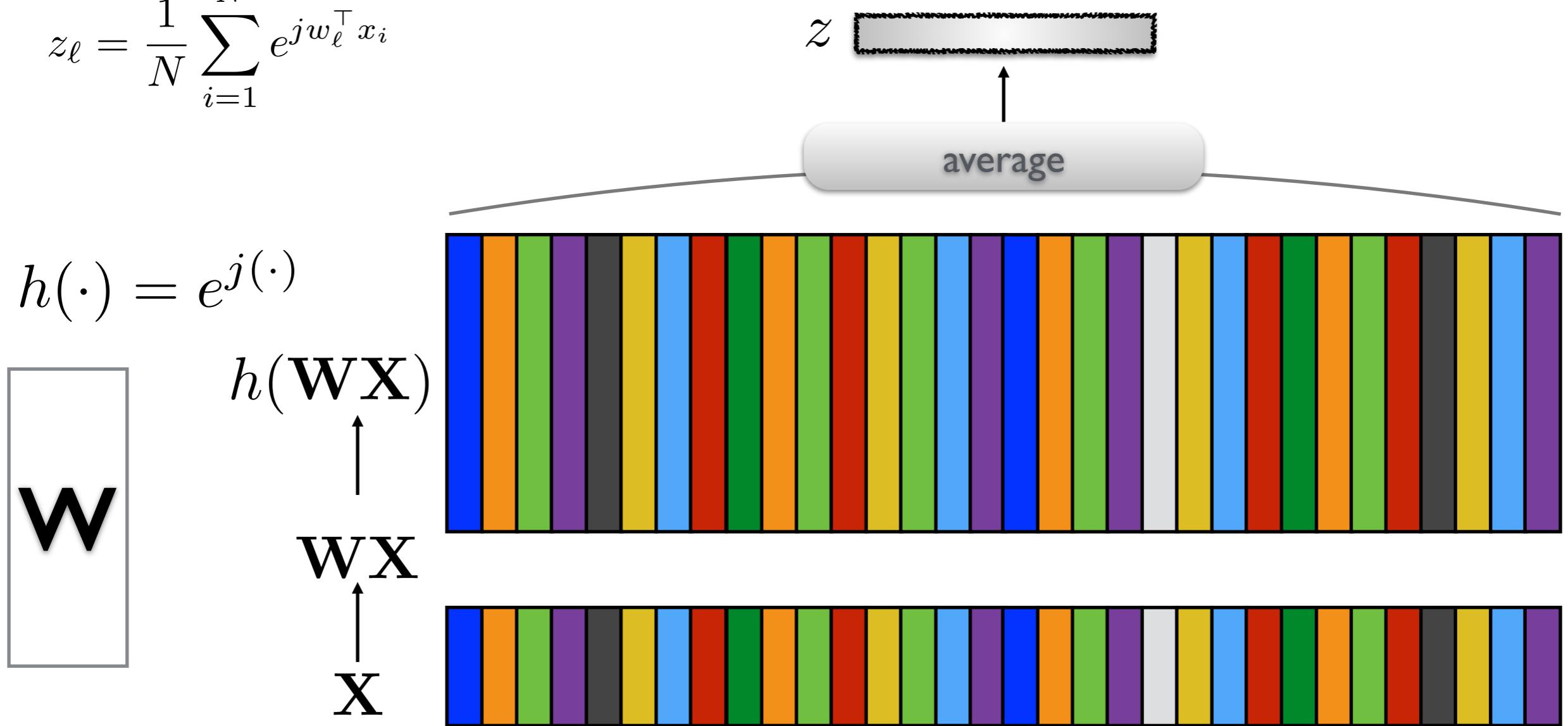


Computational Aspects

Sketching

- ## ■ empirical characteristic function

$$z_\ell = \frac{1}{N} \sum_{i=1}^N e^{j w_\ell^\top x_i}$$



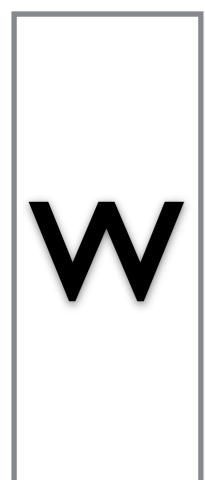
Computational Aspects

■ Sketching

- empirical characteristic function

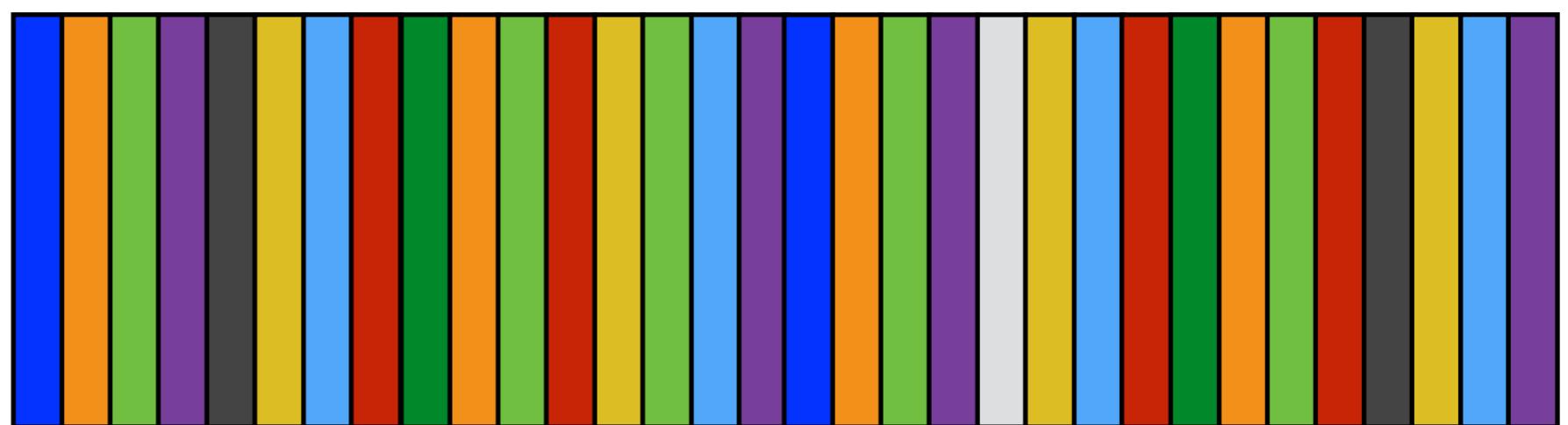
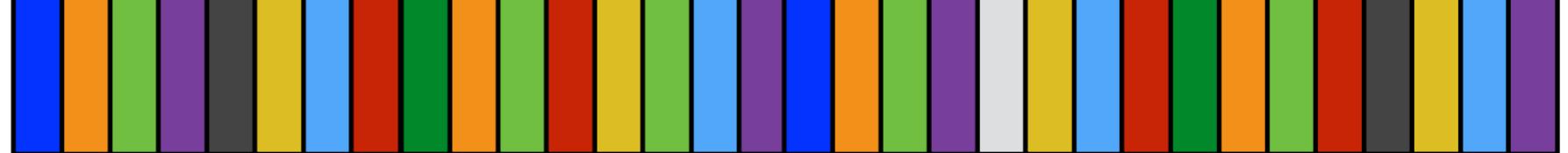
$$z_\ell = \frac{1}{N} \sum_{i=1}^N e^{j w_\ell^\top x_i}$$

$$h(\cdot) = e^{j(\cdot)}$$



$$h(\mathbf{WX})$$

$$\mathbf{WX}$$



■ ~ One-layer random neural net

- DNN ~ hierarchical sketching ?

see also [Bruna & al 2013, Giryes & al 2015]

$$z$$



average



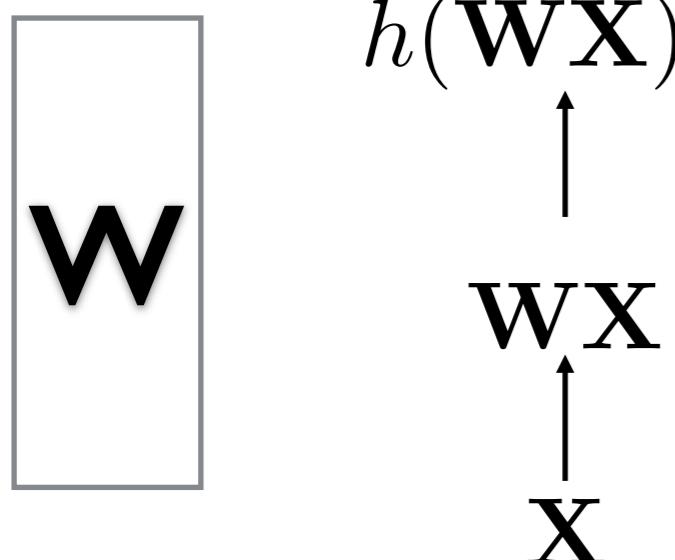
Computational Aspects

■ Sketching

- empirical characteristic function

$$z_\ell = \frac{1}{N} \sum_{i=1}^N e^{j w_\ell^\top x_i}$$

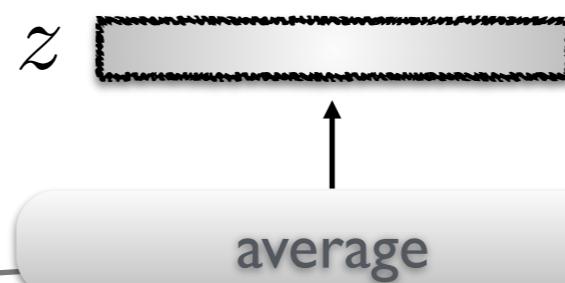
$$h(\cdot) = e^{j(\cdot)}$$



■ ~ One-layer random neural net

- DNN ~ hierarchical sketching ?

see also [Bruna & al 2013, Giryes & al 2015]



■ Privacy-preserving

- sketch and forget

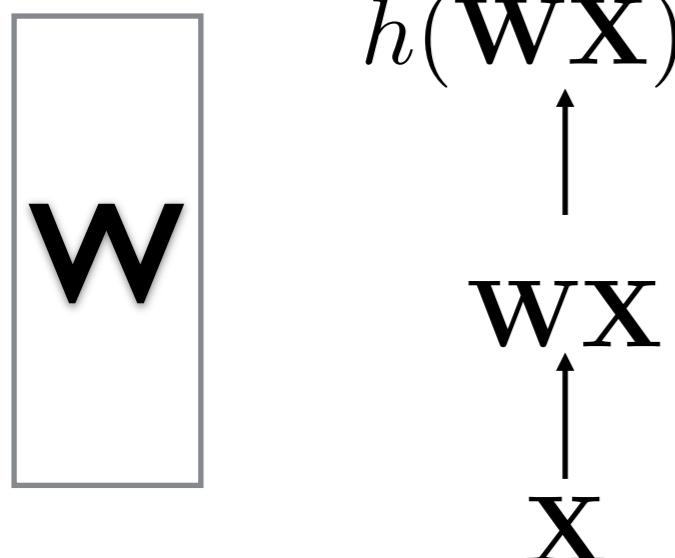
Computational Aspects

■ Sketching

- empirical characteristic function

$$z_\ell = \frac{1}{N} \sum_{i=1}^N e^{j w_\ell^\top x_i}$$

$$h(\cdot) = e^{j(\cdot)}$$



■ Streaming algorithms

- One pass; online update

z 

↑
average

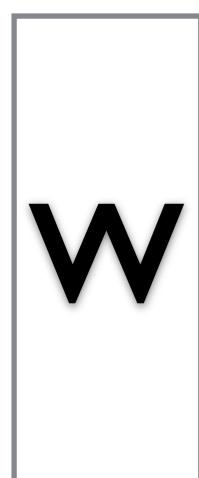
Computational Aspects

Sketching

- empirical characteristic function

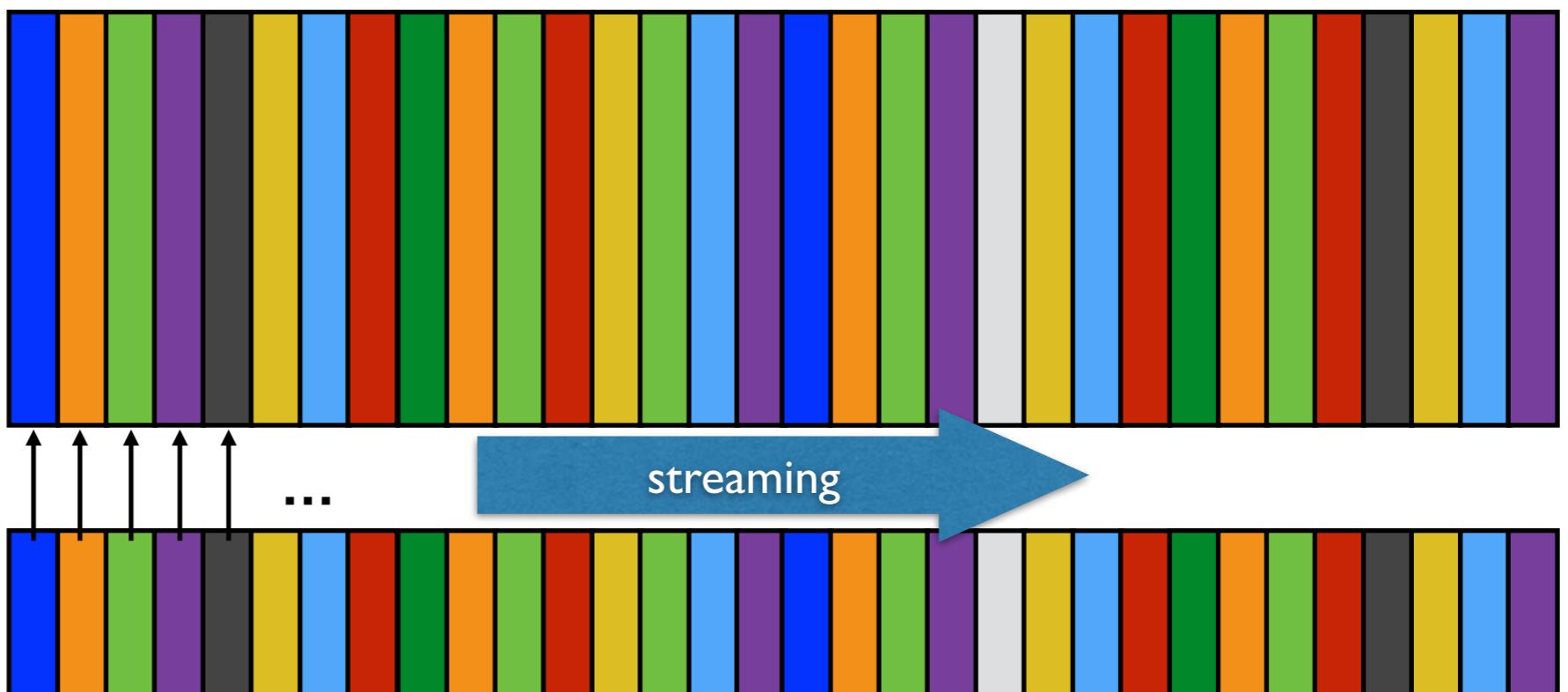
$$z_\ell = \frac{1}{N} \sum_{i=1}^N e^{j w_\ell^\top x_i}$$

$$h(\cdot) = e^{j(\cdot)}$$



$$h(\mathbf{W}\mathbf{X})$$

$$\mathbf{W}\mathbf{X}$$



Streaming algorithms

- One pass; online update

$$z$$

average

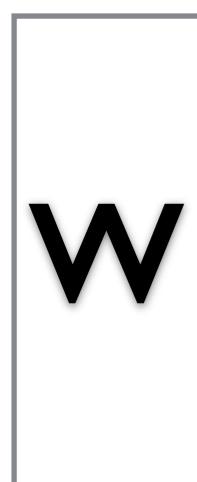
Computational Aspects

Sketching

- empirical characteristic function

$$z_\ell = \frac{1}{N} \sum_{i=1}^N e^{j w_\ell^\top x_i}$$

$$h(\cdot) = e^{j(\cdot)}$$

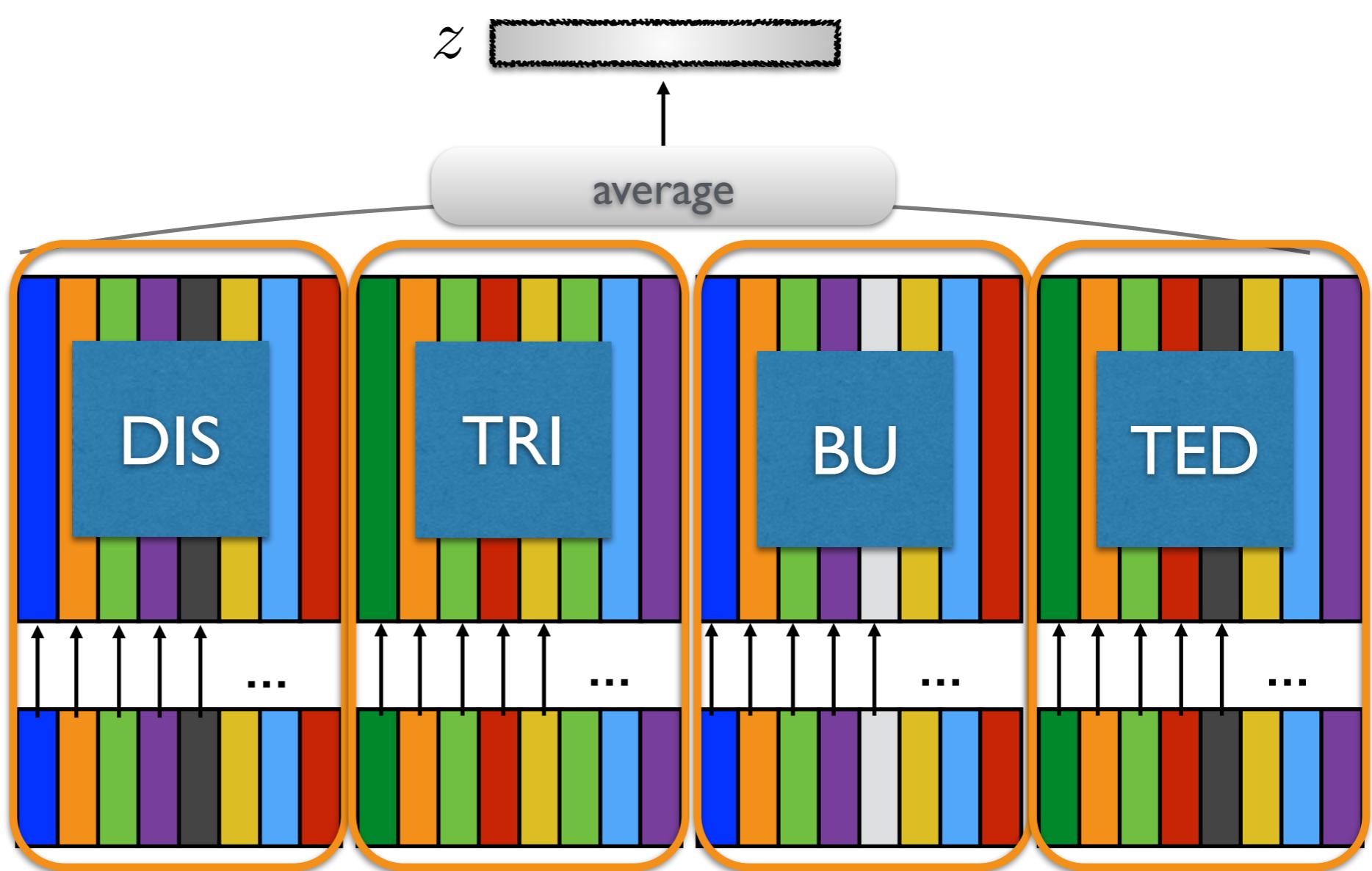


$$h(WX)$$

$$\begin{matrix} W \\ \times \\ X \end{matrix}$$

Distributed computing

- Decentralized (HADOOP) / parallel (GPU)

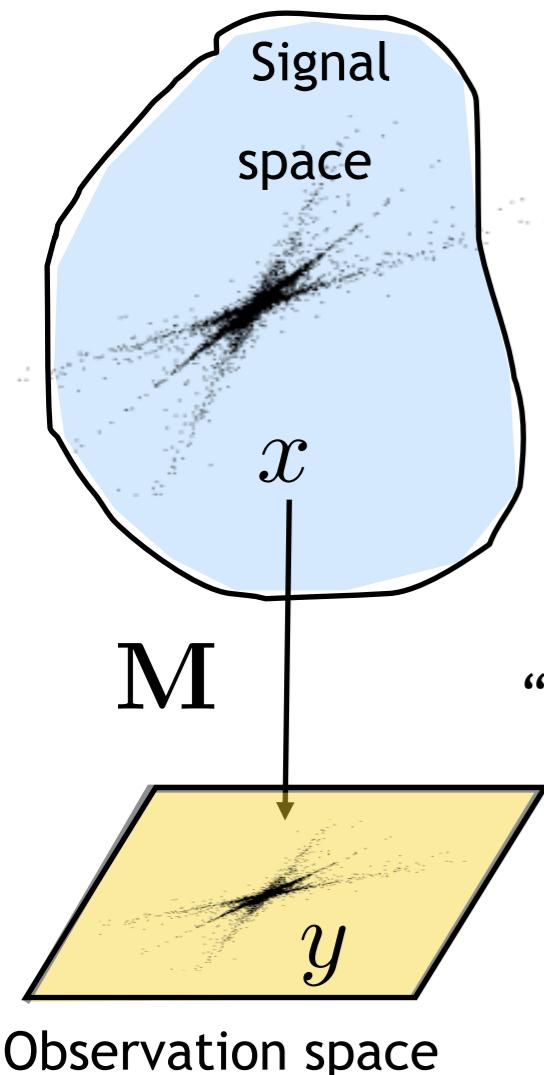




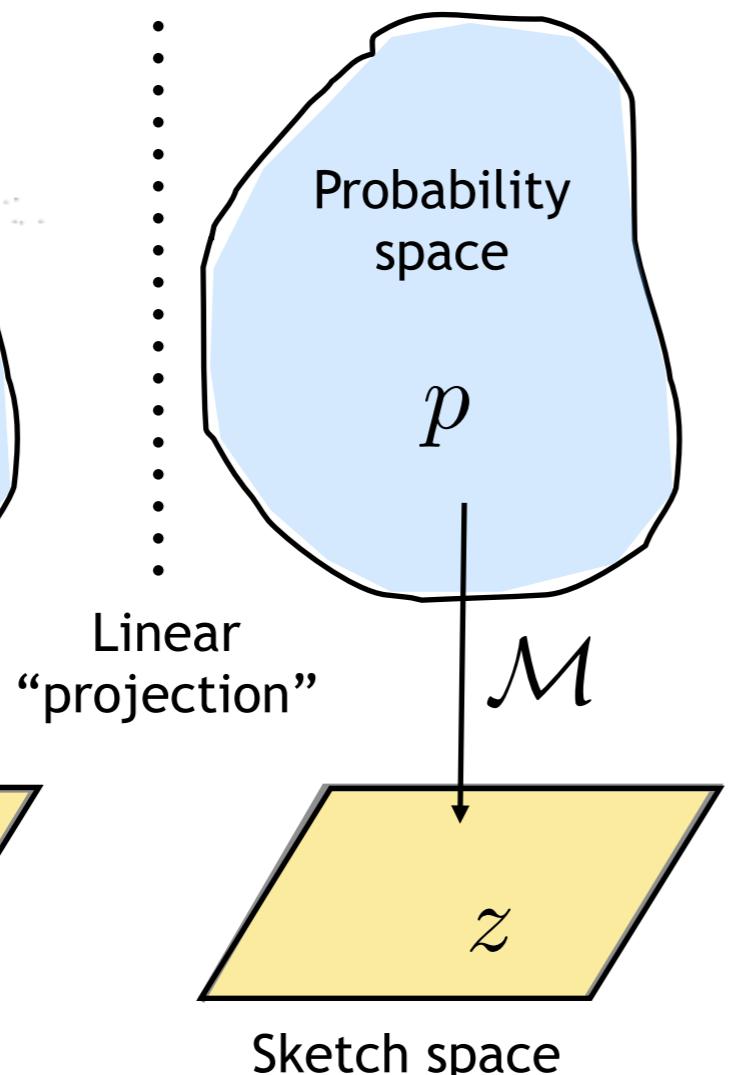
Conclusion

Projections & Learning

■ Signal Processing ■ compressive sensing



► Machine Learning ■ compressive learning



- Reduce dimension of data items
- Reduce **size of collection**

- Compressive sensing
random projections of data items
- Compressive learning with **sketches**
random projections of collections
 - nonlinear in the feature vectors
 - linear in their probability distribution

Summary

Challenge: compress \mathcal{X} before learning ?

- Compressive clustering & Compressive GMM
 - Bourrier, G., Perez, *Compressive Gaussian Mixture Estimation*. ICASSP 2013
 - Keriven & G.. *Compressive Gaussian Mixture Estimation by Orthogonal Matching Pursuit with Replacement*. SPARS 2015, Cambridge, United Kingdom
 - Keriven & al, *Sketching for Large-Scale Learning of Mixture Models* (draft)

Details: poster N. Keriven

Information preservation ?

- Unified framework covering projections & sketches
- Instance Optimal Decoders
- Restricted Isometry Property
 - Bourrier & al, *Fundamental performance limits for ideal decoders in high-dimensional linear inverse problems*. IEEE Transactions on Information Theory, 2014

Recent / ongoing work / challenges

■ Dimension reduction ?

- Sufficient dimension for RIP $m = O(d_B(\Sigma - \Sigma))$
 - Puy, Davies & G., *Recipes for stable linear embeddings from Hilbert spaces to \mathbb{R}^m* , hal-01203614, see also EUSIPCO 2015 and [Dirksen 2014]

Details: poster G. Puy

- RIP for sketches in RKHS applied to compressive GMM
 - upcoming, Keriven, Bourrier, Perez & G.
- Compressive statistical learning: intrinsic dimension of PCA and other related learning tasks
 - work in progress, Blanchard & G.

■ Decoders?

- RIP-based guarantees for general (convex & nonconvex) regularizers
 - Traonmilin & G, *Stable recovery of low-dimensional cones in Hilbert spaces - One RIP to rule them all*, arXiv:1510.00504
 - extends sharp RIP $1/\sqrt{2}$ [Cai & Zhang 2014] beyond sparsity (low-rank; block/structured ...)

Interested ? Joint the team

- Postdoc / R&D engineer positions @ IRISA

- ✓ theoretical and algorithmic foundations of large-scale machine learning & signal processing
- ✓ funded by ERC project PLEASE



PLEASE

projection, learning and sparsity for efficient data processing

THANKS

erc