PLEASE

projection, learning and sparsity for efficient data processing

Random Feature Moments for Compressive Statistical Learning

Rémi Gribonval - Inria Rennes - Bretagne Atlantique

remi.gribonval@inria.fr

Joint work with: G. Blanchard (U. Potsdam)

N. Keriven, Y Traonmilin (Inria Rennes)

Main Contributors & Collaborators

- Anthony Bourrier

- Nicolas Keriven

- Yann Traonmilin

- Gilles Puy

- Nicolas Tremblay

- Gilles Blanchard

- Mike Davies

- Patrick Perez

Some unforeseen connections ?

- Signal processing & machine learning
  - inverse problems & generalized method of moments
  - embeddings with random projections & random features /kernels
  - image super-resolution, source localization & k-means
- Continuous *vs* discrete ?
  - wavelets (1990s): from continuous to discrete
  - compressive sensing (2000s): in the discrete world
  - compressive learning: continuous strikes back to circumvent curse of dimensionality !
  - links with off-the-grid compressive sensing, FRI, high-resolution methods

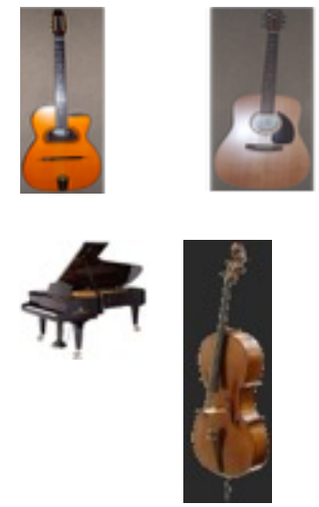
Statistical Learning

- Goal
  - **infer** parameters θ **from training data** to achieve a certain task
  - ensure **generalization to unseen data** of similar type (avoid overfitting)
  - often focus on "**model free**" assumption on data distribution
- Training collection = large point cloud $\mathcal{X}$
  - signals, images, …
  - feature vectors, labels, …

Digit recognition (MNIST)

Image classification

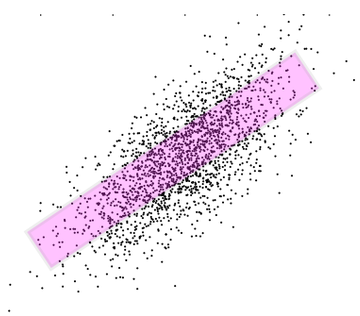
Sound classification



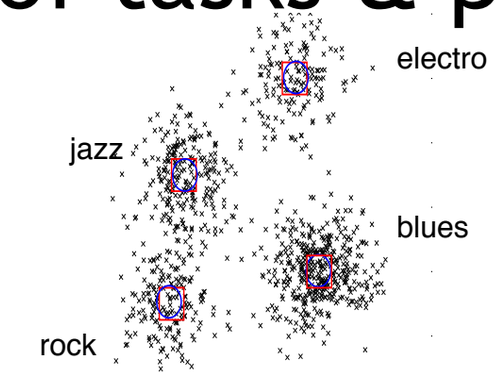
Statistical Learning

- Goal
  - **infer** parameters θ **from training data** to achieve a certain task
  - ensure **generalization to unseen data** of similar type (avoid overfitting)
  - often focus on "**model free**" assumption on data distribution
- Training collection = large point cloud $\mathcal{X}$
  - signals, images, …
  - feature vectors, labels, …

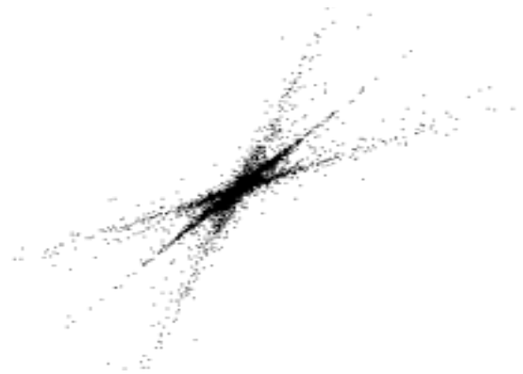
Examples of tasks & parameters



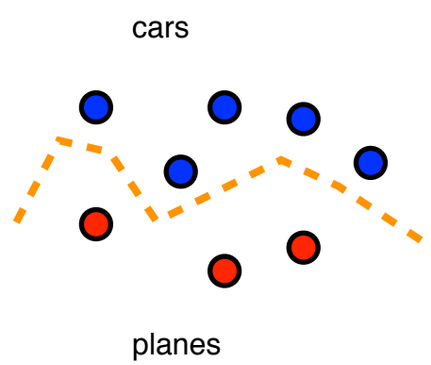
- PCA
  - θ = principal subspace



- Clustering
  - θ = centroids

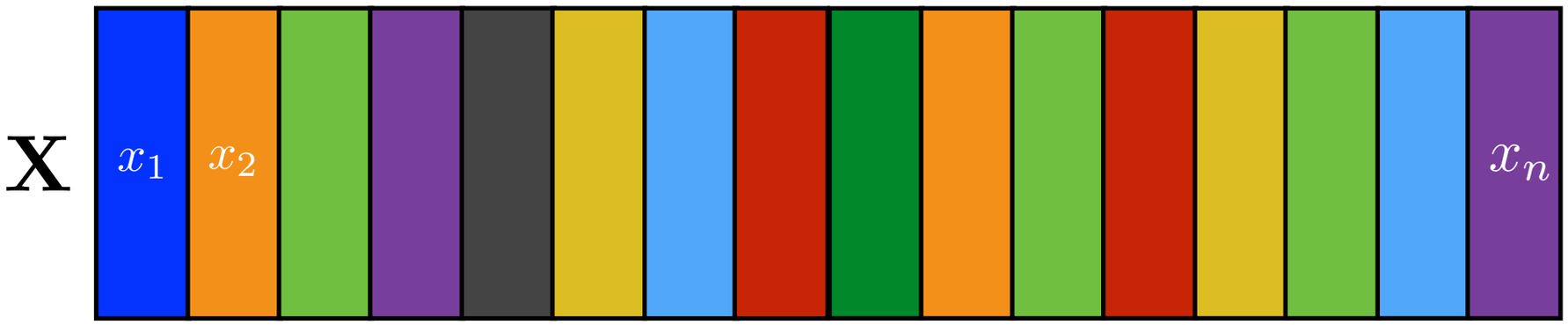


- Dictionary learning
  - θ = dictionary atoms

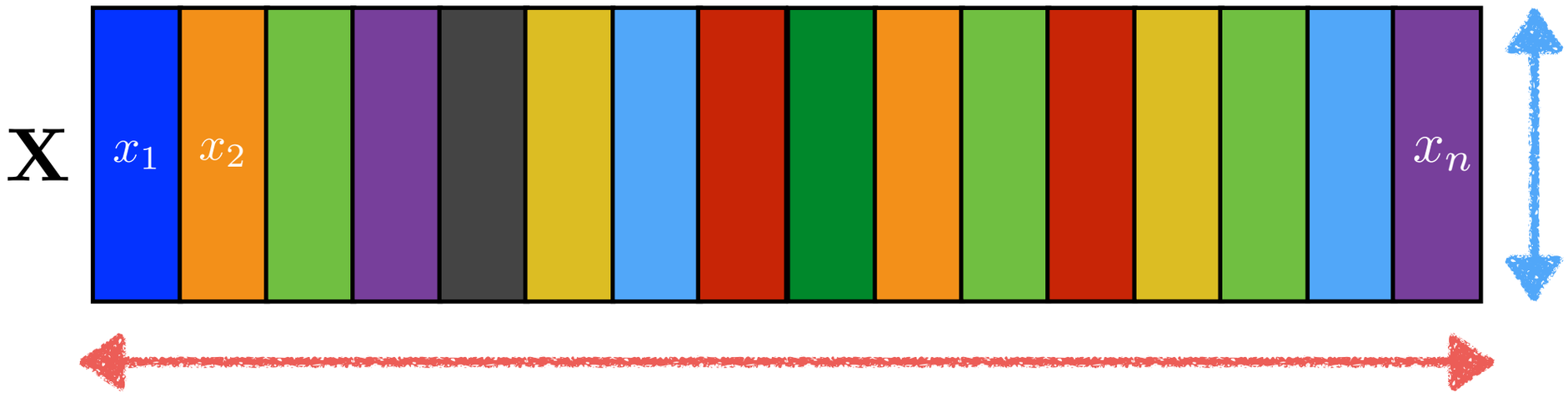


- Classification
  - θ = classifier parameters (e.g. support vectors)

Large-scale learning

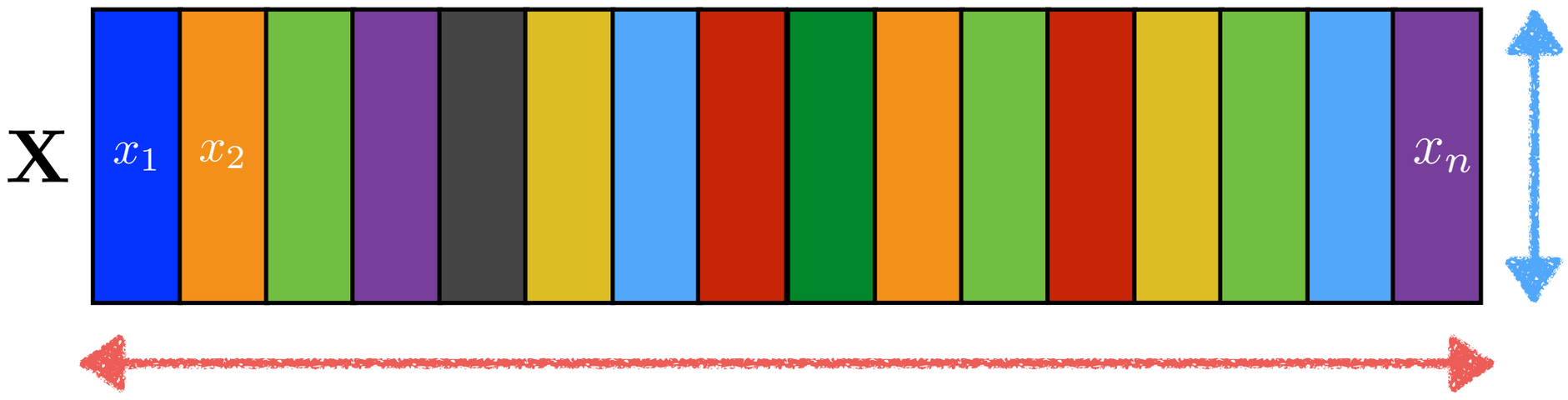


Large-scale learning



- High feature dimension *d*
- Large collection size *n* = *“volume”*

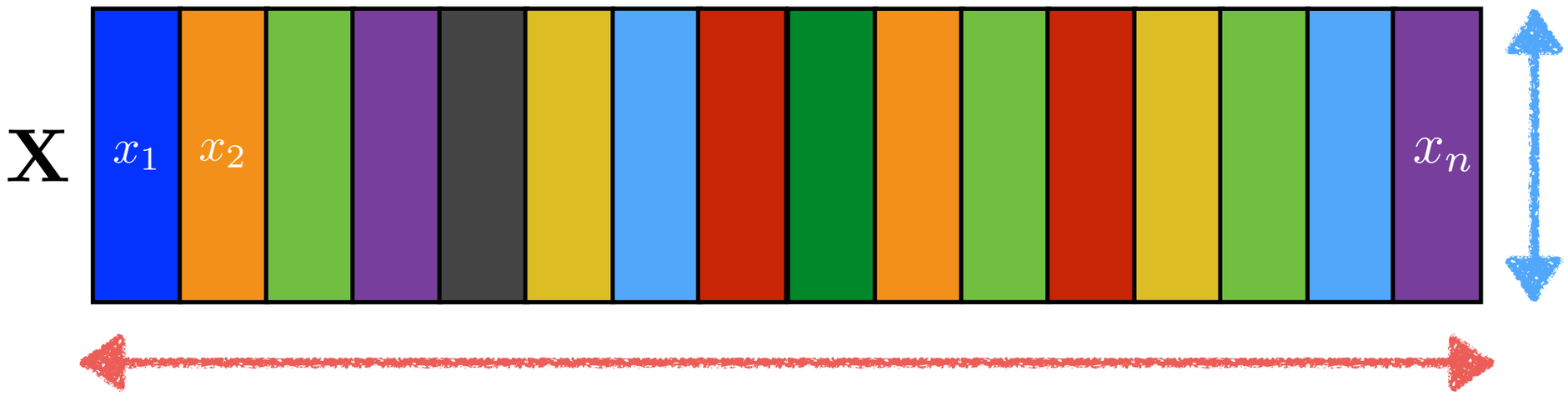
Large-scale learning



- High feature dimension *d*
- Large collection size *n* = *"volume"*

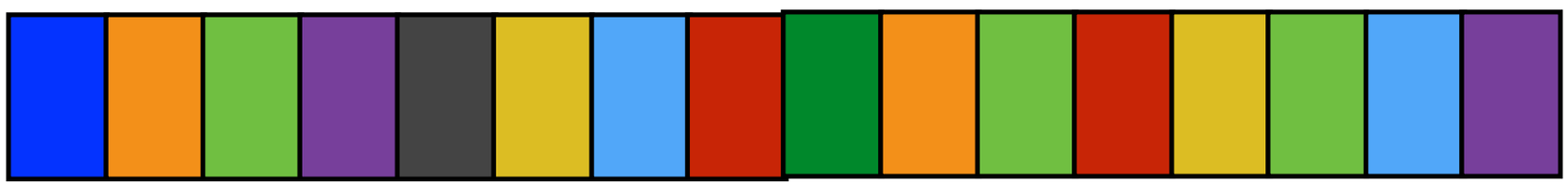
Challenge: compress $\mathcal{X}$ before learning ?

Compressive learning: three routes



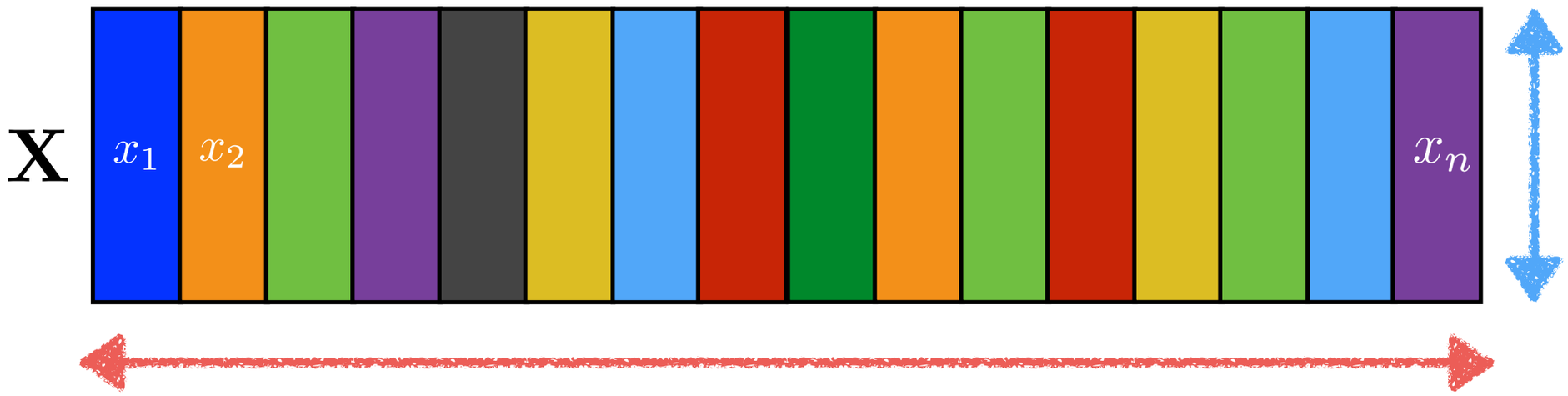
- dimension reduction
- subsampling
- **sketching**

$\mathbf{Y} = \mathbf{M}\mathbf{X}$

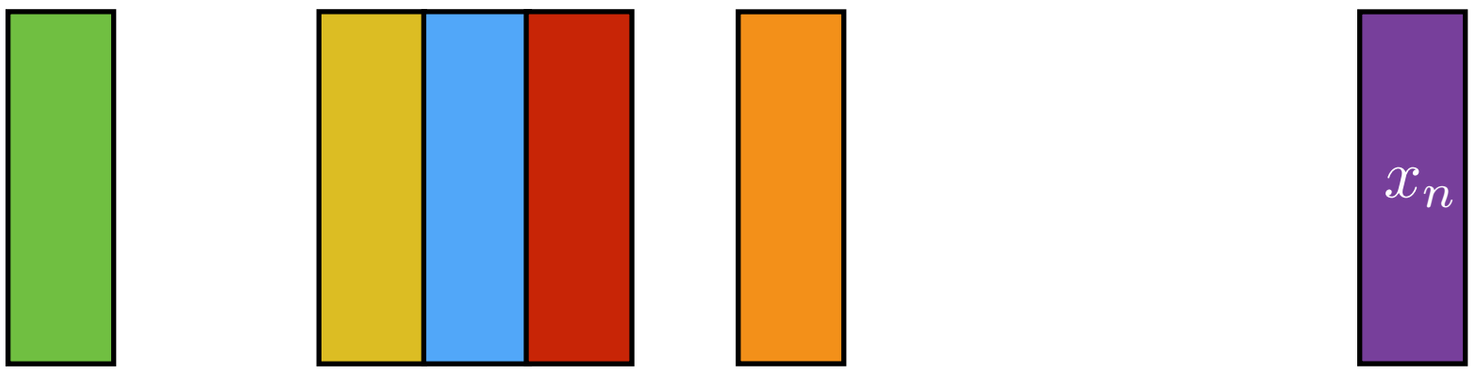


random projections - Johnson Lindenstrauss lemma
see e.g. [Calderbank & al 2009, Reboredo & al 2013]

Compressive learning: three routes



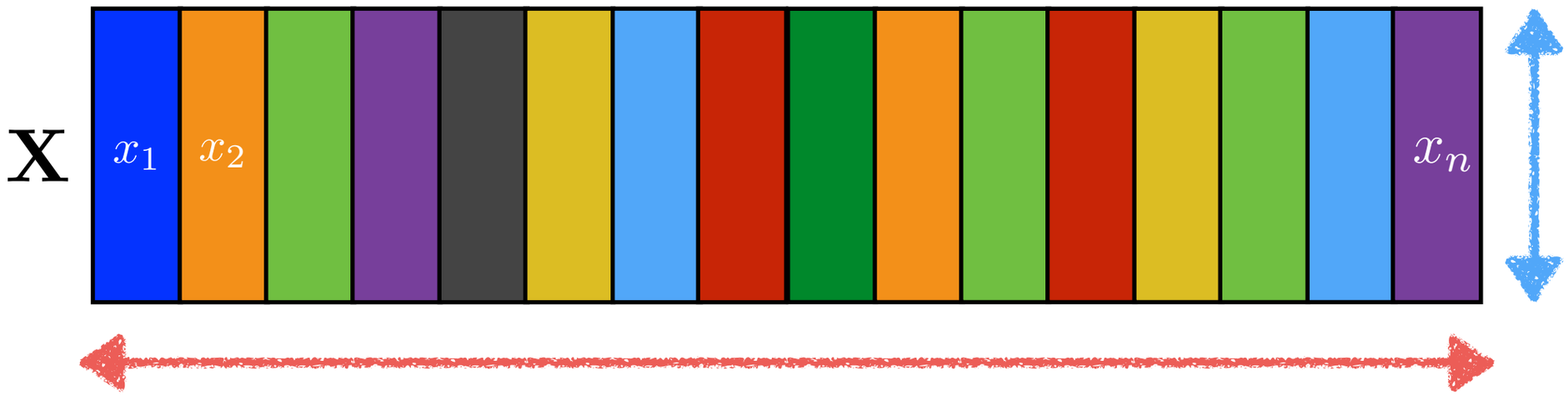
- dimension reduction
- subsampling
- **sketching**



Nyström method & coresets

see e.g. [Williams&Seeger 2000, Agarwal & al 2003, Felman 2010]

Compressive learning: three routes



- dimension reduction
- subsampling
- **random moments**

$$\begin{matrix}\mathbb{E}\Phi_1(X)\\ \dots \\ \mathbb{E}\Phi_m(X)\end{matrix} \quad z \in \mathbb{R}^m$$

Inspiration: *compressive sensing* *[Foucart & Rauhut 2013]*
sketching/hashing *[Thaper & al 2002, Cormode & al 2005]*
Connections with: *generalized method of moments* *[Hall 2005]*
kernel mean embeddings *[Smola & al 2007, Sriperimbudur & al 2010]*

Example: Compressive K-means on MNIST

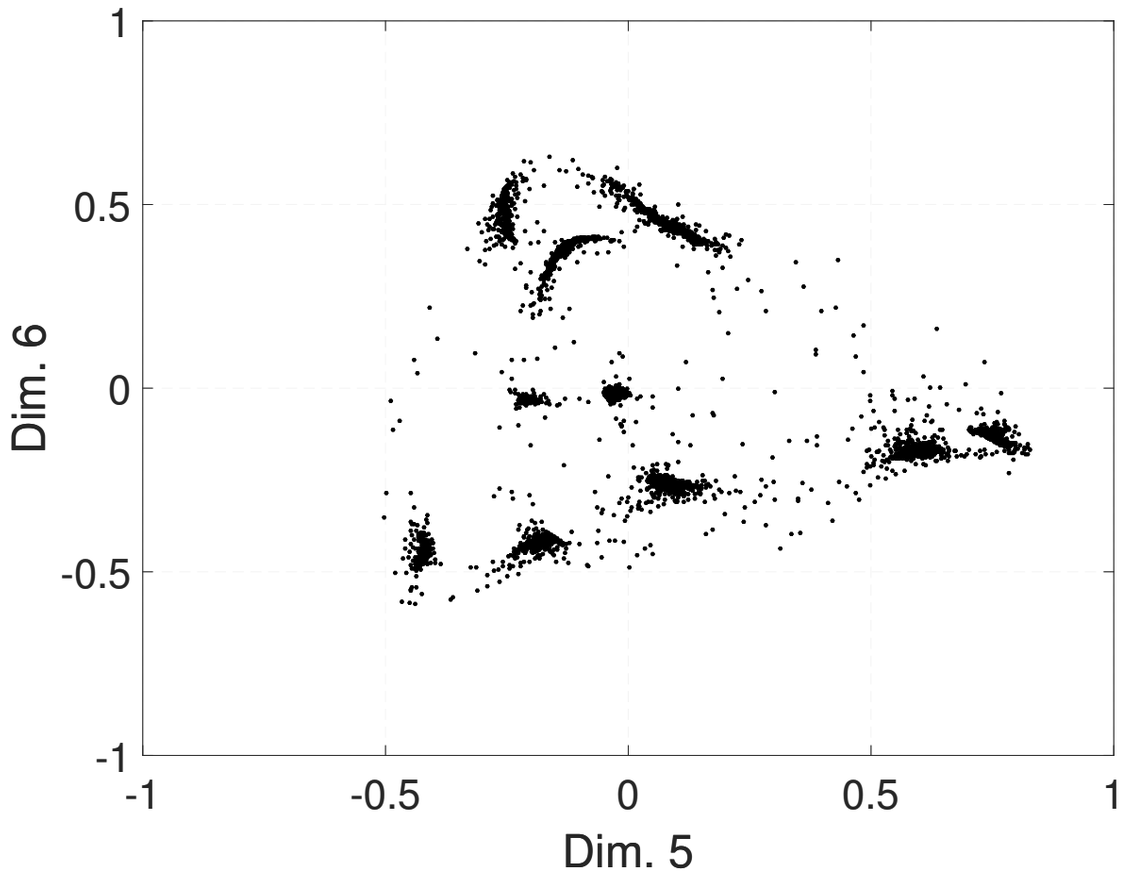
Training set $\mathcal{X}$

$n = 70000;\ d = 784;\ k = 10$

Example: Compressive K-means on MNIST

Training set $\mathcal{X}$

$n = 70000$; $d = k = 10$

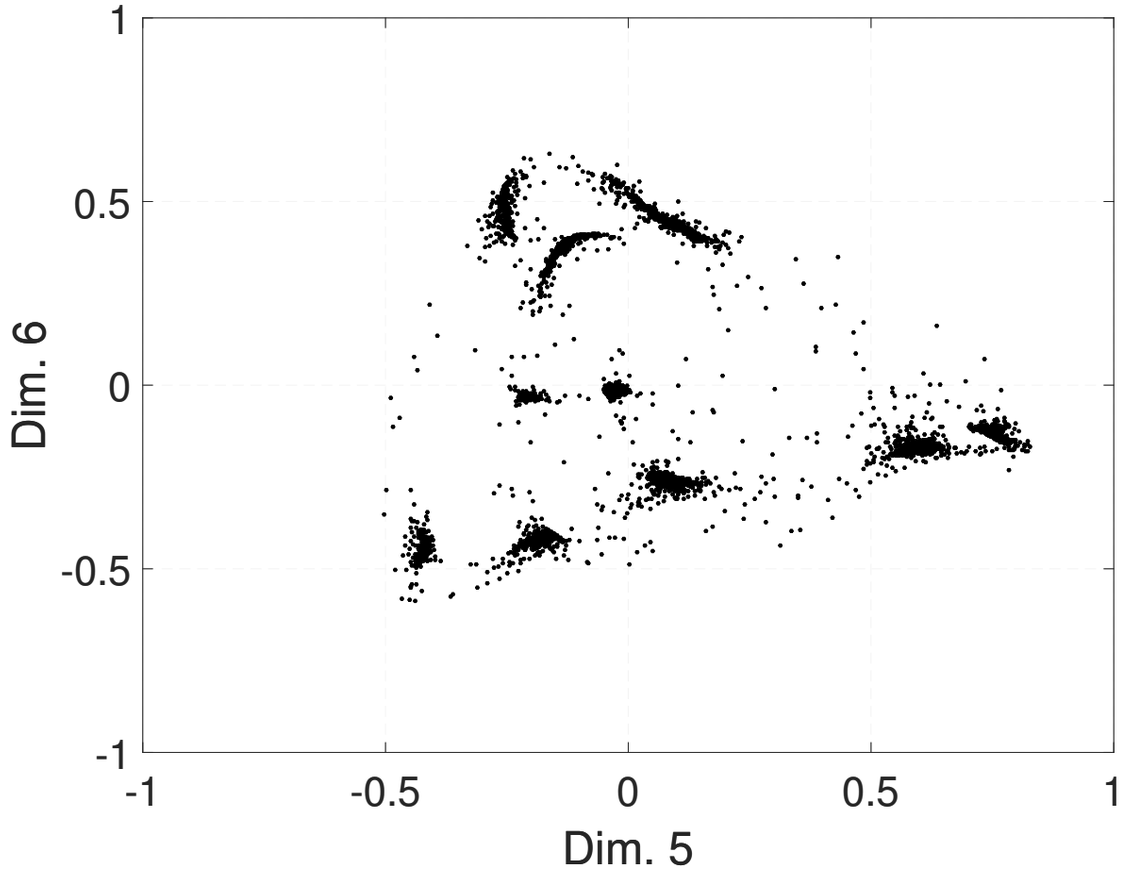


Spectral features

Example: Compressive K-means on MNIST

Training set $\mathcal{X}$

$$n = 70000;\ d = k = 10$$



Spectral features

$$\mathrm{Sketch}(\mathcal{X})$$

streaming / distributed computation

Sketch vector

memory size *independent* of *n*

$$m \gtrsim kd = 100$$

$$z \in \mathbb{R}^m = \frac{1}{n}\sum_{i=1}^{n} \Phi(x_i)$$

Example: Compressive K-means on MNIST

$$n = 70000;\ d = k = 10$$

Sketch vector

memory size *independent* of *n*

$$m \gtrsim kd = 100$$

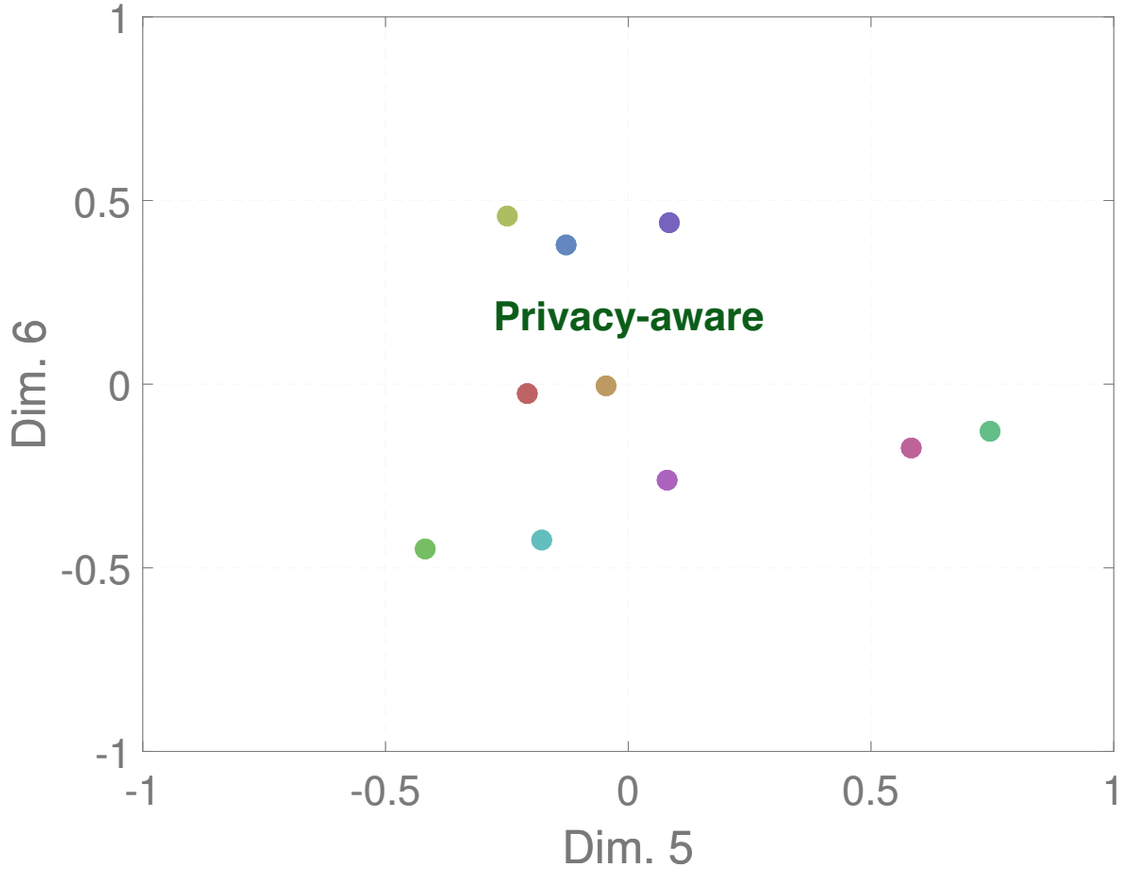
$$z \in \mathbb{R}^m = \frac{1}{n}\sum_{i=1}^{n} \Phi(x_i)$$

streaming / distributed computation

Privacy-aware

Example: Compressive K-means on MNIST

$$n = 70000;\ d = k = 10$$



Sketch vector

memory size *independent* of *n*

$$m \gtrsim kd = 100$$

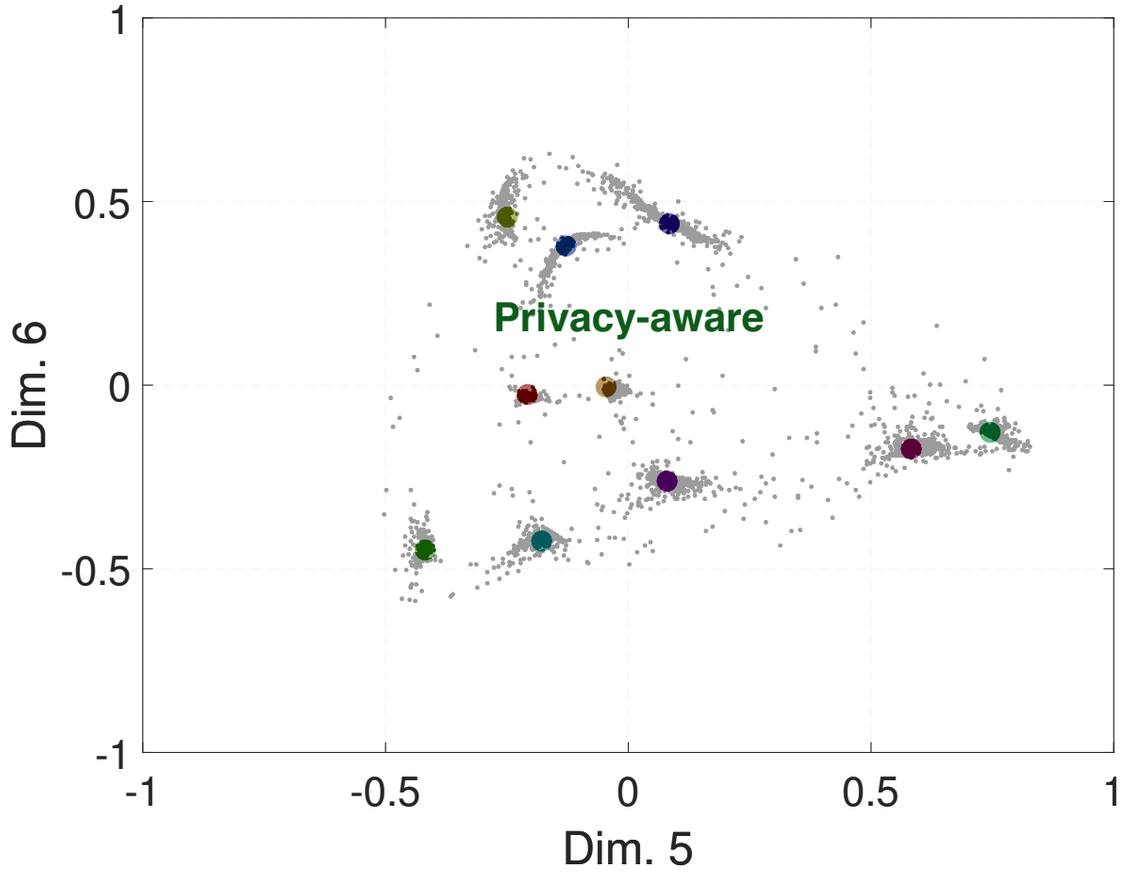
$$z \in \mathbb{R}^m = \frac{1}{n}\sum_{i=1}^{n} \Phi(x_i)$$

streaming / distributed computation

Learn centroids
from sketch
= moment *fitting*

Example: Compressive K-means on MNIST

$$n = 70000;\ d = k = 10$$



Sketch vector

memory size *independent* of *n*

$$m \gtrsim kd = 100$$

$$z \in \mathbb{R}^m = \frac{1}{n}\sum_{i=1}^{n} \Phi(x_i)$$

streaming / distributed computation

Learn centroids
from sketch
= moment *fitting*

Agenda

- From Statistical Learning to Compressive Learning
  - ***Principle***
  - *Guarantees*
- Worked Examples:
  - *Compressive PCA*
  - *Compressive K-means*
  - *Compressive Gaussian Mixture Modeling*
- Conclusion and perspectives

Moments & kernel mean embeddings

- Data distribution $X \sim p(x)$

- Sketch = vector of generalized moments

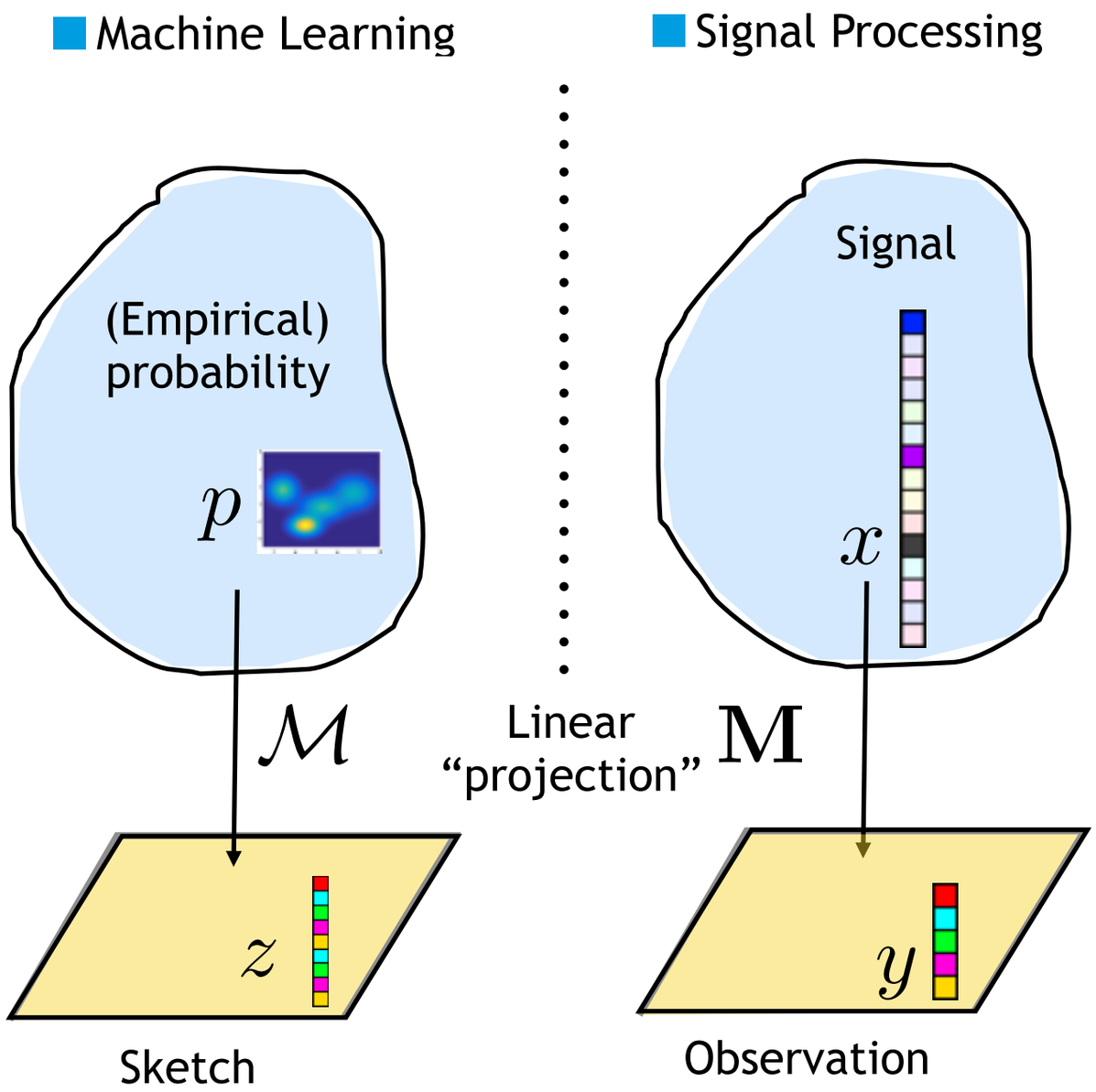
$$z = \frac{1}{n} \sum_{i=1}^{n} \Phi(x_i) \approx \mathbb{E}\ \Phi(X) = \int \Phi(x) p(x) dx$$

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

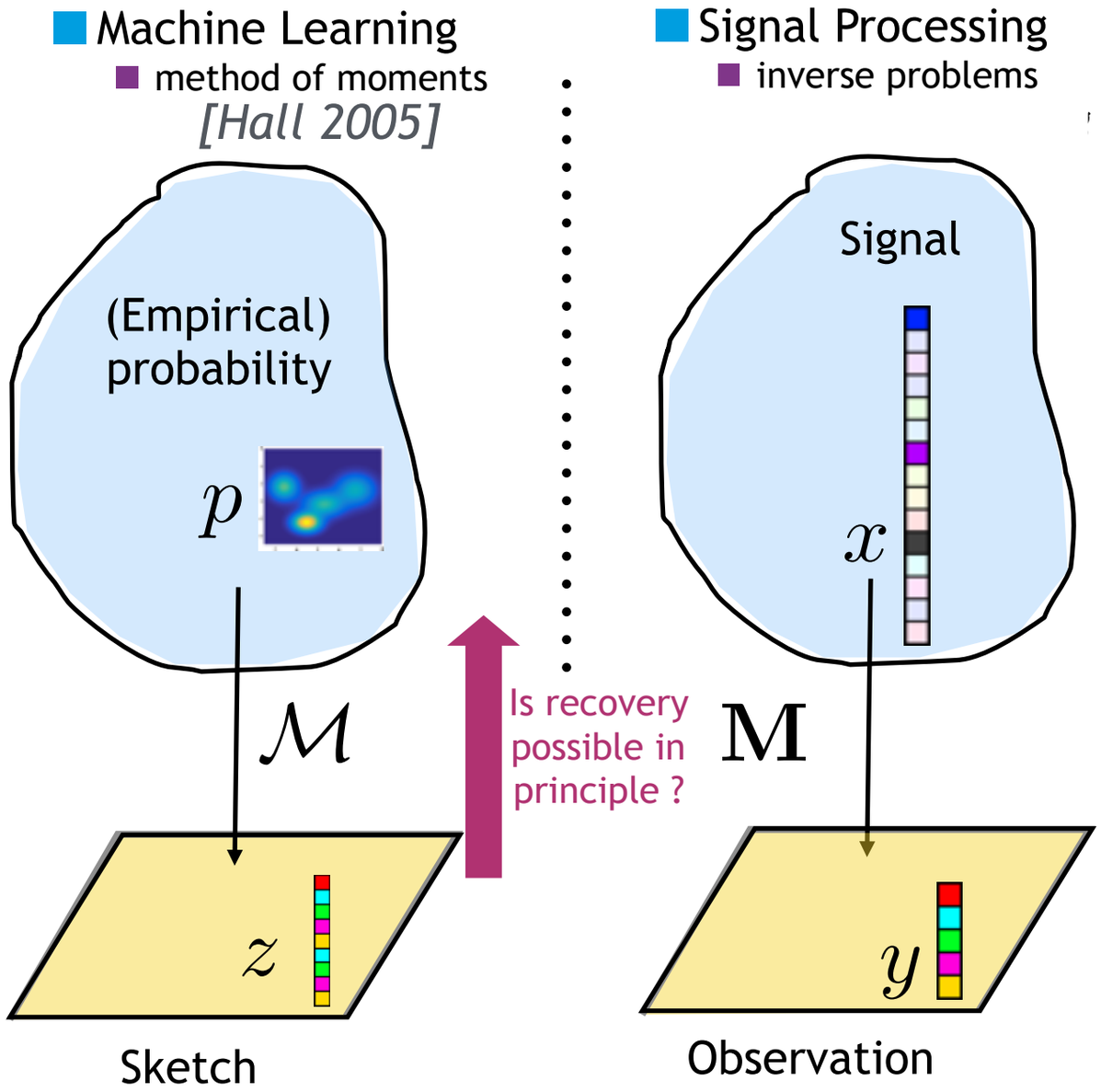
- finite-dimensional **Mean Map Embedding**, *[cf Smola & al 2007, Sriperumbudur & al 2010]*

$$\mathcal{M}(p) := \mathbb{E}_{X \sim p} \Phi(X)$$

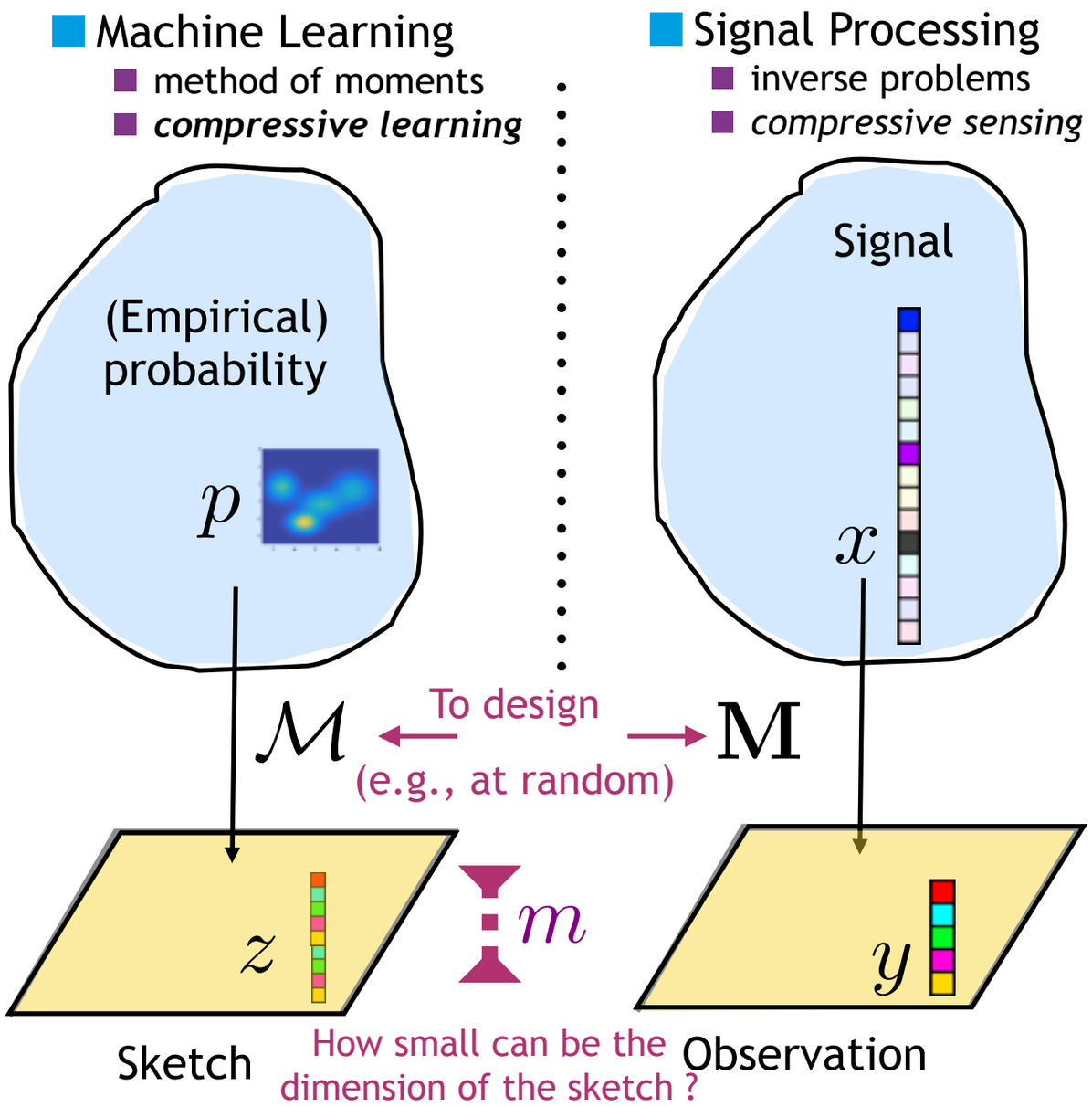
Question 1: Information preservation ?



Question 1: Information preservation ?

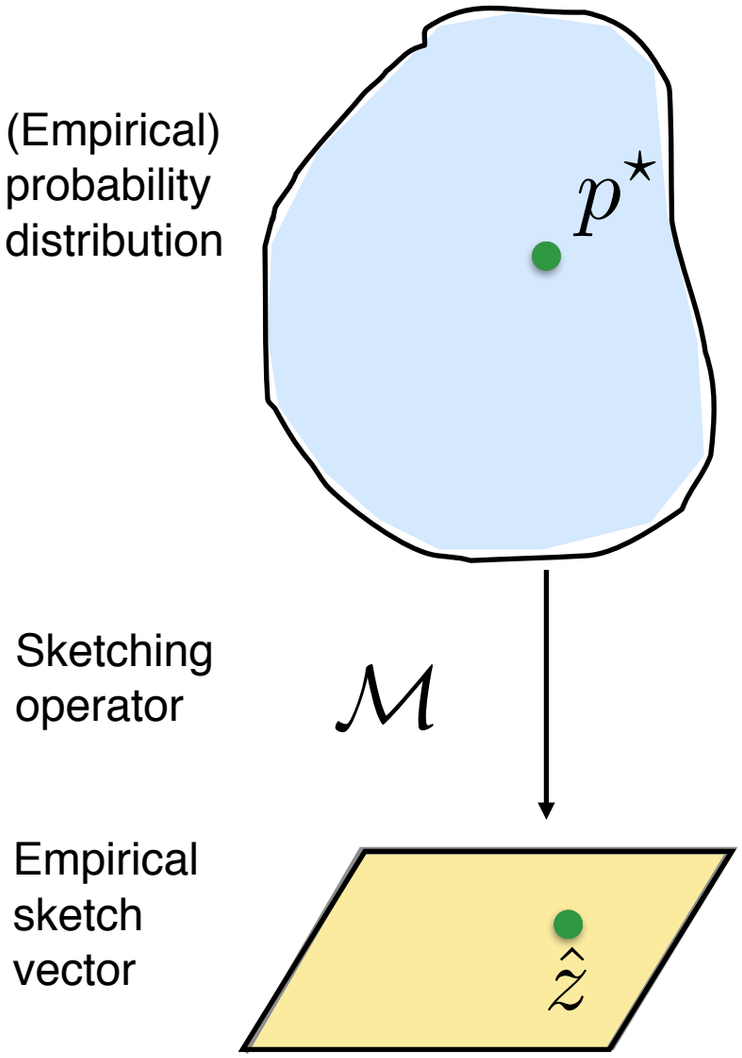


Question 2: Dimension reduction ?



Compressive Statistical Learning

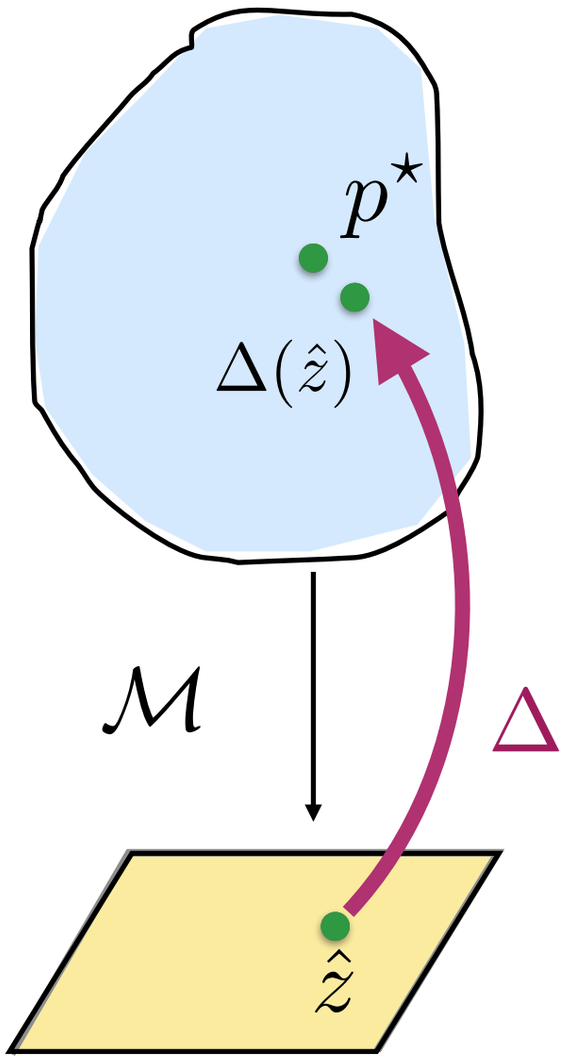
Existence of good ***decoder ?***



Compressive Statistical Learning

Existence of good ***decoder ?***

(Empirical) probability distribution



Sketching operator

Empirical sketch vector

Compressive Statistical Learning

Existence of good ***decoder ?***

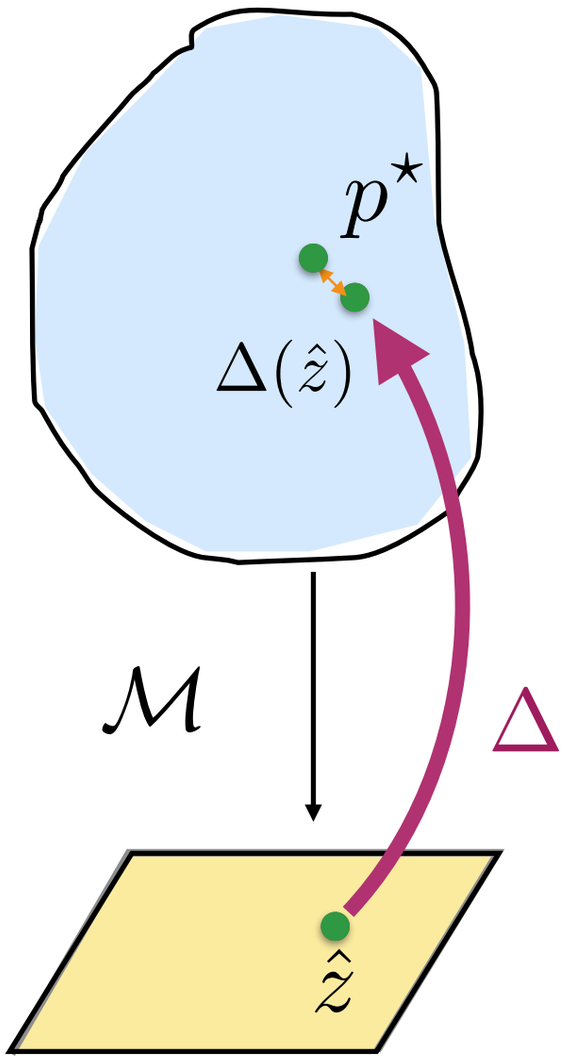
some quality metric

$$\|p^\star - \Delta(\hat{z})\|_\ell \lesssim \text{???}$$

(Empirical) probability distribution

Sketching operator

Empirical sketch vector



Compressive Statistical Learning

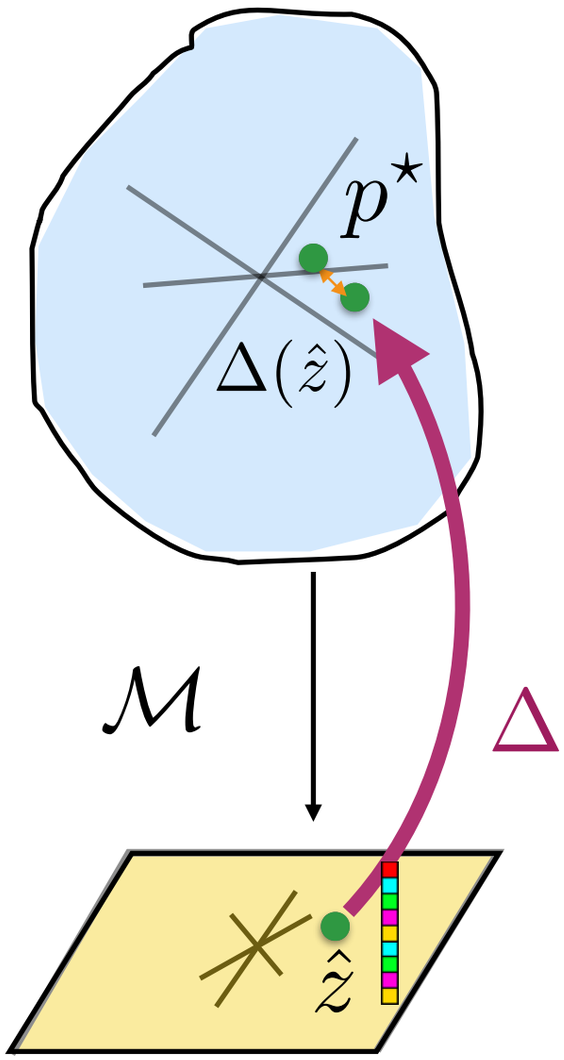
Existence of good ***decoder ?***

some quality metric

$$\forall p^\star \in \Sigma$$

$$\|p^\star - \Delta(\hat{z})\|_\ell \lesssim \|\mathcal{M}(\hat{p}_n - p^\star)\|_2$$

(Empirical) probability distribution



On a model set Σ

= distributions "of interest"

ex: Gaussian Mixture Model

Compressive Statistical Learning

Existence of good ***decoder ?***

Lower Restricted Isometry Property (LRIP)

some quality metric

$$\forall p^\star \in \Sigma$$

$$\|p^\star - \Delta(\hat{z})\|_\ell \lesssim \|\mathcal{M}(\hat{p}_n - p^\star)\|_2$$

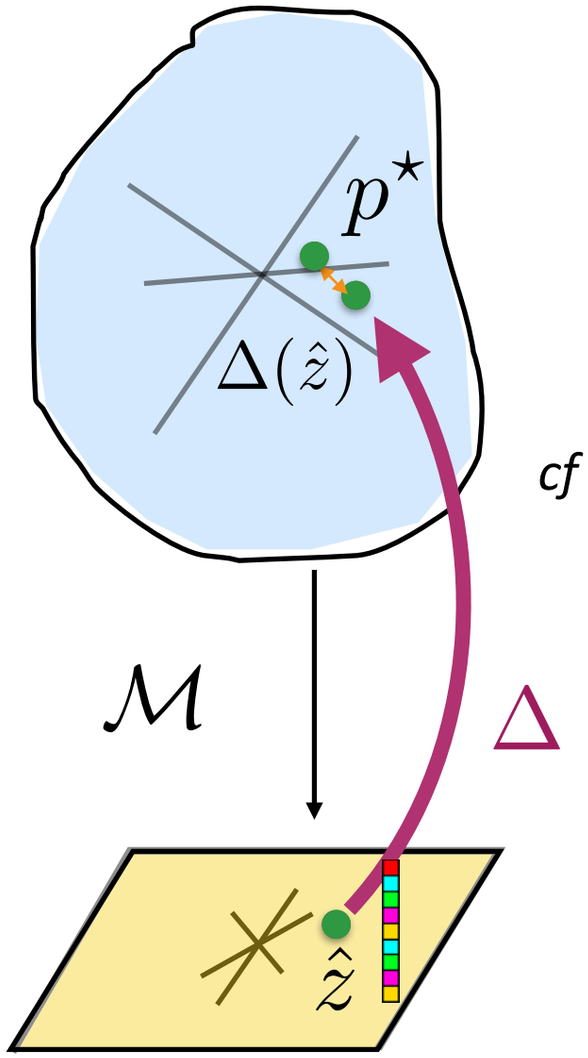
$$\forall q, q' \in \Sigma$$

$$\|q - q'\|_\ell \lesssim \|\mathcal{M}(q - q')\|_2$$

(Empirical) probability distribution

Sketching operator

Empirical sketch vector

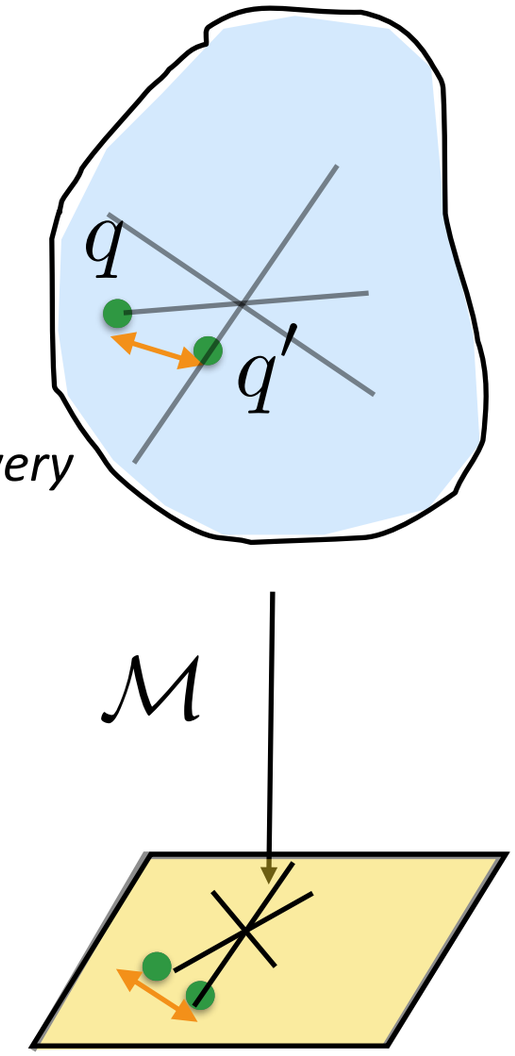


On a model set Σ

= distributions "of interest"

ex: Gaussian Mixture Model

cf [Cohen & al 2009] for sparse recovery



Compressive Statistical Learning

Existence of good ***decoder ?***

Lower Restricted Isometry Property (LRIP)

some quality metric

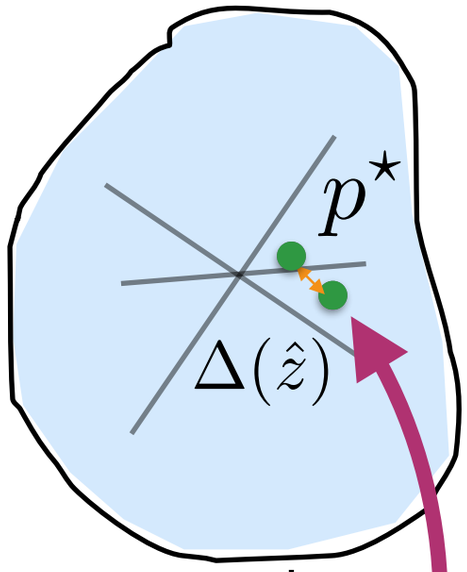
$$\forall p^\star \in \Sigma$$

$$\|p^\star - \Delta(\hat{z})\|_\ell \lesssim \|\mathcal{M}(\hat{p}_n - p^\star)\|_2$$

$$\forall q, q' \in \Sigma$$

$$\|q - q'\|_\ell \lesssim \|\mathcal{M}(q - q')\|_2$$

(Empirical) probability distribution

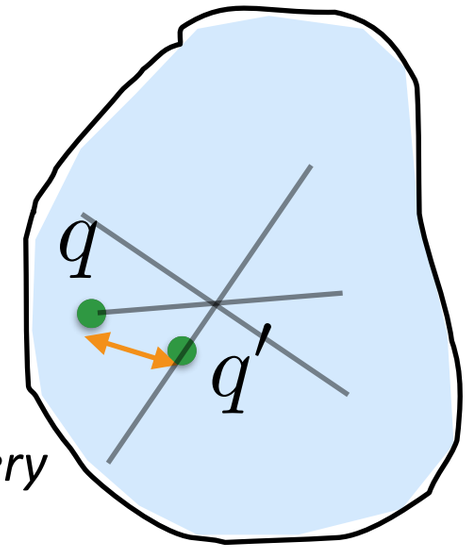


On a model set Σ

= distributions "of interest"

ex: Gaussian Mixture Model

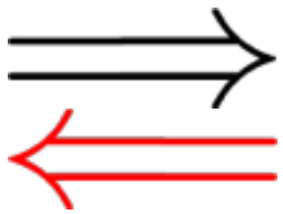
cf [Cohen & al 2009] for sparse recovery



Sketching operator

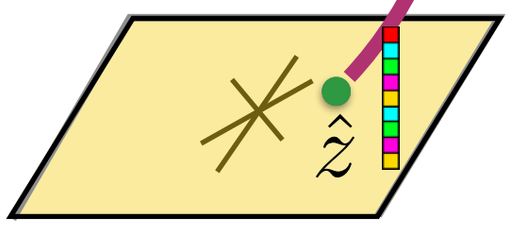
$\mathcal{M}$

Δ



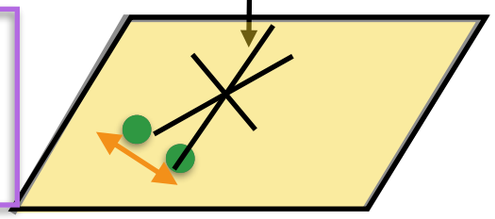
$\mathcal{M}$

Empirical sketch vector



$$\Delta(z) := \arg\min_{p \in \Sigma} \|z - \mathcal{M}p\|_2$$

= Generalized method of moments



Compressive Statistical Learning

Existence of good ***decoder ?***

Lower Restricted Isometry Property (LRIP)

some quality metric

$$\forall p^\star$$

$$\|p^\star - \Delta(\hat{z})\|_\ell \lesssim \|\mathcal{M}(\hat{p}_n - p^\star)\|_2 + d(p^\star, \Sigma)$$

$$\forall q, q' \in \Sigma$$

$$\|q - q'\|_\ell \lesssim \|\mathcal{M}(q - q')\|_2$$

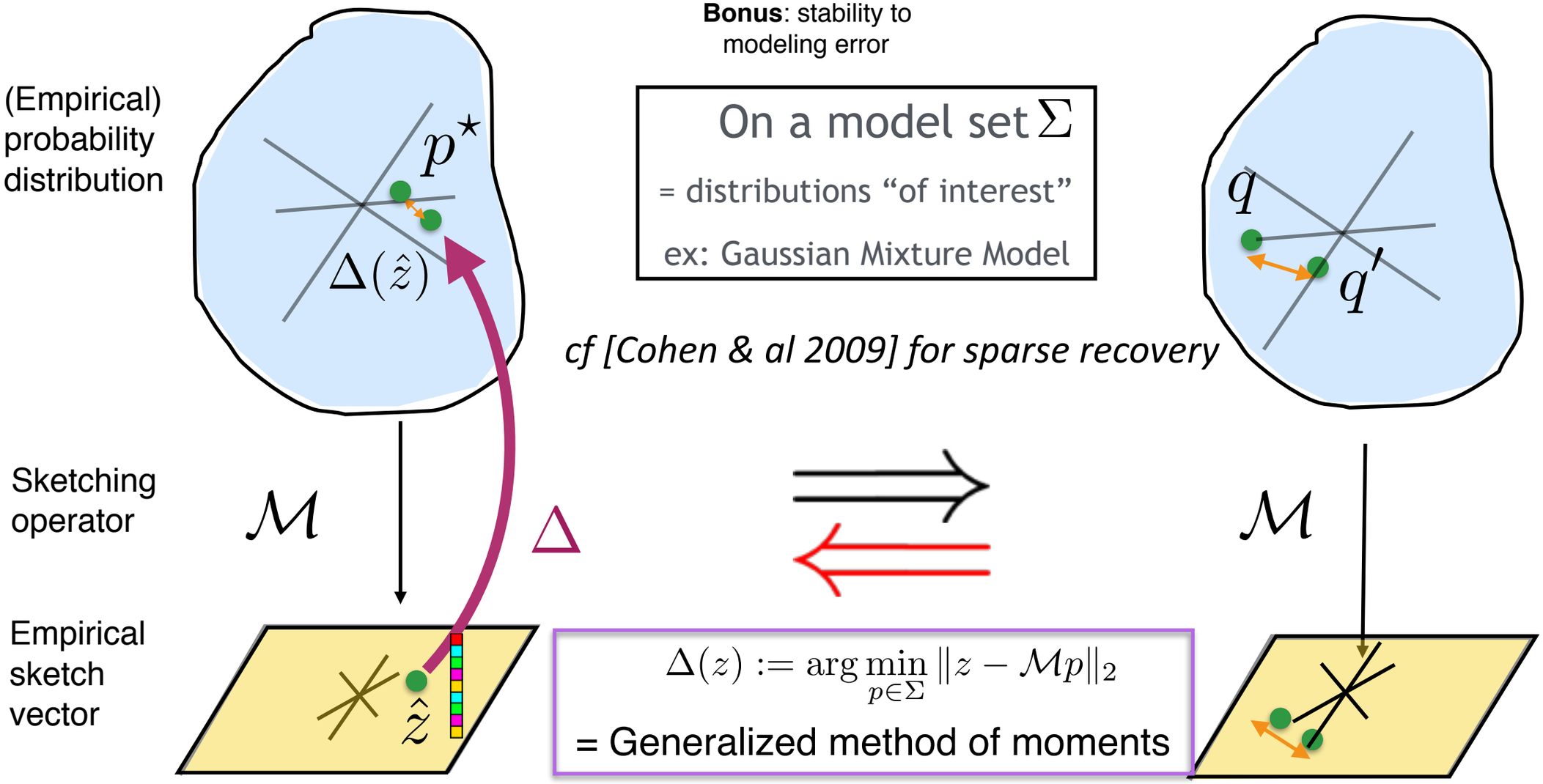
Bonus: stability to modeling error

On a model set Σ

= distributions "of interest"

ex: Gaussian Mixture Model

cf [Cohen & al 2009] for sparse recovery



$$\Delta(z) := \arg\min_{p \in \Sigma} \|z - \mathcal{M}p\|_2$$

= Generalized method of moments

Which metric ?

Statistical learning

- **task** given by **loss function** $\ell(x, \theta)$
- **learning** = minimizing *risk*

$$R(p, \theta) = \mathbb{E}_{x \sim p} \ell(x, \theta)$$

- *given* ***"arbitrary"*** $p^\star$ *(e.g.: bounded)*

$$\theta^\star \in \arg\min_\theta R(p^\star, \theta)$$

Examples

- k-means $\quad \ell(x, \theta) = \min_i \|x - \theta_i\|^2$
- k-medians $\quad \ell(x, \theta) = \min_i \|x - \theta_i\|$
- PCA $\quad \ell(x, \theta) = \|x - P_\theta x\|^2$
- Max Likelihood $\quad \ell(x, \theta) = -\log p_\theta(x)$
- …

Which metric ?

Statistical learning

- **task** given by **loss function** $\ell(x, \theta)$
- **learning** = minimizing *risk*

$$R(p, \theta) = \mathbb{E}_{x \sim p} \ell(x, \theta)$$

- *given* ***"arbitrary"*** $p^\star$ *(e.g.: bounded)*

$$\theta^\star \in \arg\min_\theta R(p^\star, \theta)$$

Empirical Risk Minimization(ERM)

- given training collection $x_i \sim p^\star$, i.i.d.
- minimize *empirical risk*

$$R(\hat{p}_n, \theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(x_i, \theta)$$

- *empirical estimate*

$$\hat{\theta}_n \in \arg\min_\theta R(\hat{p}_n, \theta)$$

Examples

- k-means $\ell(x, \theta) = \min_i \|x - \theta_i\|^2$
- k-medians $\ell(x, \theta) = \min_i \|x - \theta_i\|$
- PCA $\ell(x, \theta) = \|x - P_\theta x\|^2$
- Max Likelihood $\ell(x, \theta) = -\log p_\theta(x)$
- …

Which metric ?

Statistical learning

- **task** given by **loss function** $\ell(x, \theta)$
- **learning** = minimizing *risk*

$$R(p, \theta) = \mathbb{E}_{x \sim p} \ell(x, \theta)$$

- *given* ***"arbitrary"*** $p^\star$ *(e.g.: bounded)*

$$\theta^\star \in \arg\min_\theta R(p^\star, \theta)$$

Empirical Risk Minimization(ERM)

- given training collection $x_i \sim p^\star$, i.i.d.
- minimize *empirical risk*

$$R(\hat{p}_n, \theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(x_i, \theta)$$

- *empirical estimate*

$$\hat{\theta}_n \in \arg\min_\theta R(\hat{p}_n, \theta)$$

Guarantees for ERM

- **Goal:** control *excess risk*

$$R(p^\star, \hat{\theta}_n) \leq R(p^\star, \theta^\star) + \eta_n$$

- **Sufficient:** show that w.h.p.

$$\sup_\theta |R(\hat{p}_n, \theta) - R(p^\star, \theta)| \leq \eta_n / 2$$

Which metric ?

Statistical learning

- **task** given by **loss function** $\ell(x, \theta)$
- **learning** = minimizing *risk*

$$R(p, \theta) = \mathbb{E}_{x \sim p} \ell(x, \theta)$$

- *given* ***"arbitrary"*** $p^\star$ *(e.g.: bounded)*

$$\theta^\star \in \arg\min_\theta R(p^\star, \theta)$$

Empirical Risk Minimization(ERM)

- given training collection $x_i \sim p^\star$, i.i.d.
- minimize *empirical risk*

$$R(\hat{p}_n, \theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(x_i, \theta)$$

- *empirical estimate*

$$\hat{\theta}_n \in \arg\min_\theta R(\hat{p}_n, \theta)$$

Guarantees for ERM

- **Goal:** control *excess risk*

$$R(p^\star, \hat{\theta}_n) \leq R(p^\star, \theta^\star) + \eta_n$$

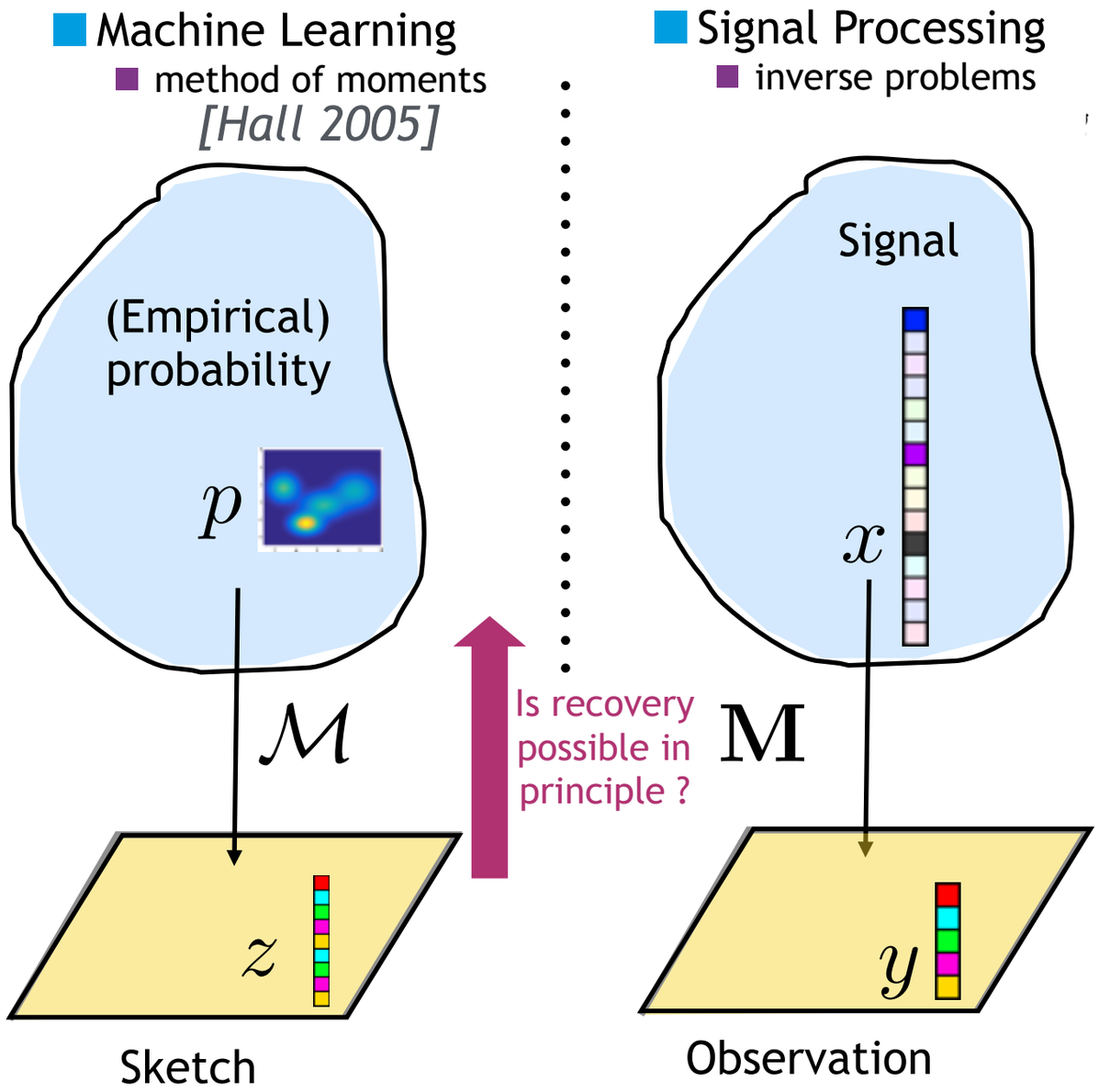
- **Sufficient:** show that w.h.p.

$$\sup_\theta |R(\hat{p}_n, \theta) - R(p^\star, \theta)| \leq \eta_n / 2$$

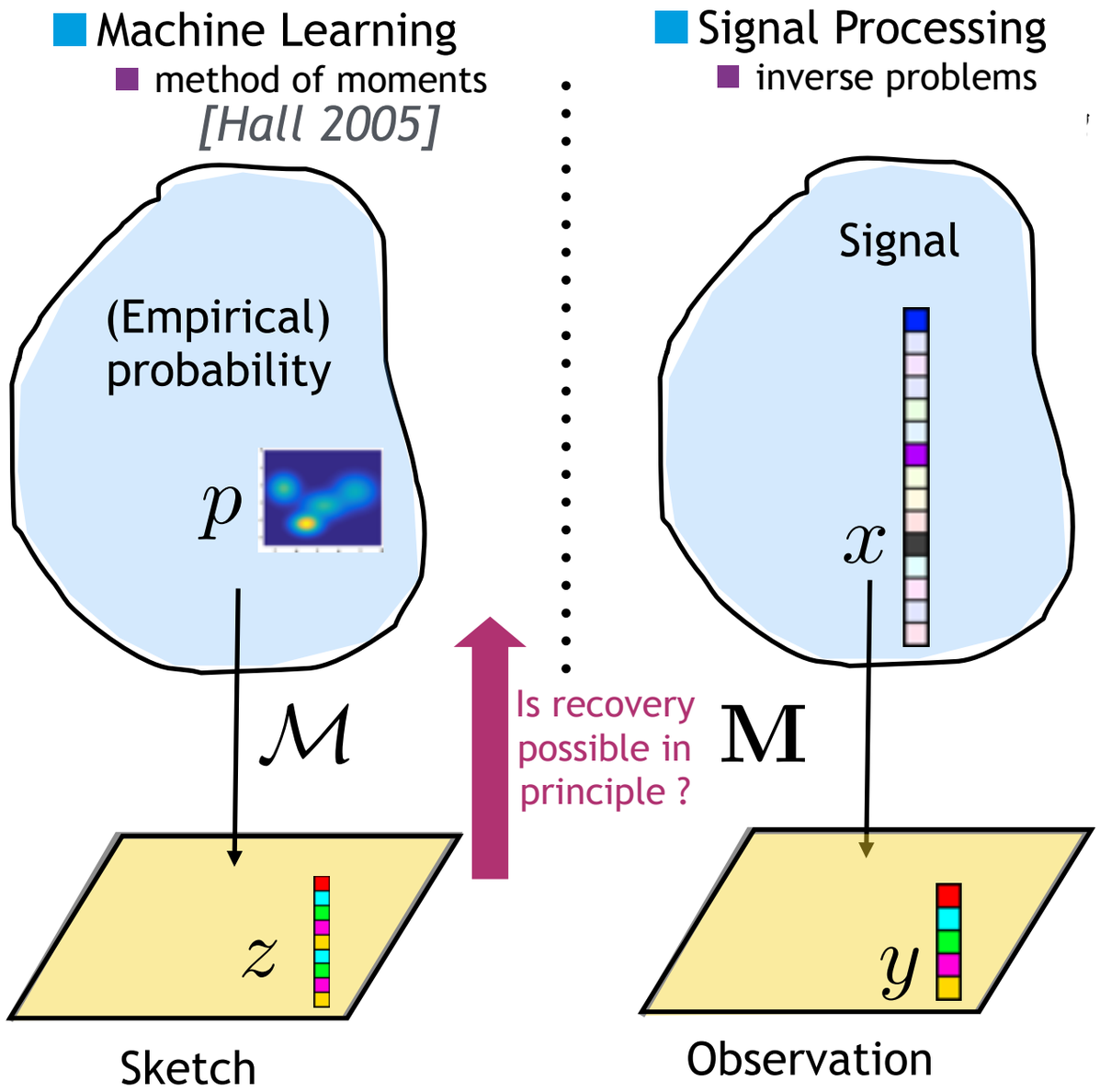
Definition: task-based metric

$$\|p' - p\|_\ell := \sup_\theta |R(p', \theta) - R(p, \theta)|$$

Question 1: Information preservation ?



Question 1: Information preservation ?



✓ Lower RIP with respect to task-based metric

Compressive Statistical Learning

- **Given a learning task** $\ell(x, \theta)$
  - Design a sketching function $\Phi(x) \in \mathbb{R}^m$ and operator $\mathcal{M}(p) := \mathbb{E}_{X \sim p} \Phi(X)$
  - Choose a model set Σ

Compressive Statistical Learning

- **Given a learning task** $\ell(x, \theta)$
  - Design a sketching function $\Phi(x) \in \mathbb{R}^m$ and operator $\mathcal{M}(p) := \mathbb{E}_{X \sim p} \Phi(X)$
  - Choose a model set Σ
- **Given a training collection**
  - Compute a sketch vector

$$z = \frac{1}{n} \sum_{i=1}^{n} \Phi(x_i) \approx \mathbb{E}\ \Phi(X)$$

Compressive Statistical Learning

- **Given a learning task** $\ell(x, \theta)$
  - Design a sketching function $\Phi(x) \in \mathbb{R}^m$ and operator $\mathcal{M}(p) := \mathbb{E}_{X \sim p} \Phi(X)$
  - Choose a model set Σ
- **Given a training collection**
  - Compute a sketch vector

$$z = \frac{1}{n} \sum_{i=1}^{n} \Phi(x_i) \approx \mathbb{E}\ \Phi(X)$$

- **Given a sketch vector**
  - Step 1: $\tilde{p} = \Delta(\hat{z}) = \arg\min_{p \in \Sigma} \|\hat{z} - \mathcal{M}p\|_2$
  - Step 2: $\hat{\theta} \in \arg\min_{\theta} R(\tilde{p}, \theta)$

Compressive Statistical Learning

Given a learning task $\ell(x, \theta)$

- Design a sketching function $\Phi(x) \in \mathbb{R}^m$ and operator $\mathcal{M}(p) := \mathbb{E}_{X \sim p} \Phi(X)$
- Choose a model set Σ

Given a training collection

- Compute a sketch vector

$$z = \frac{1}{n} \sum_{i=1}^{n} \Phi(x_i) \approx \mathbb{E}\ \Phi(X)$$

Given a sketch vector

- Step 1: $\tilde{p} = \Delta(\hat{z}) = \arg\min_{p \in \Sigma} \|\hat{z} - \mathcal{M}p\|_2$
- Step 2: $\hat{\theta} \in \arg\min_{\theta} R(\tilde{p}, \theta)$

Guarantees?

- Assume: LRIP

$$\|p - p'\|_\ell \lesssim \|\mathcal{M}(p) - \mathcal{M}(p')\|_2, \forall p, p' \in \Sigma$$

- Then: excess risk control $R(p^\star, \hat{\theta}) - R(p^\star, \theta^\star) \lesssim \|\mathcal{M}(\hat{p}_n - p^\star)\|_2 + d(p^\star, \Sigma)$

Compressive Statistical Learning

Given a learning task $\ell(x, \theta)$

- Design a sketching function $\Phi(x) \in \mathbb{R}^m$ and operator $\mathcal{M}(p) := \mathbb{E}_{X \sim p} \Phi(X)$
- Choose a model set Σ

Given a training collection

- Compute a sketch vector

$$z = \frac{1}{n} \sum_{i=1}^{n} \Phi(x_i) \approx \mathbb{E}\ \Phi(X)$$

Given a sketch vector

- Step 1: $\tilde{p} = \Delta(\hat{z}) = \arg\min_{p \in \Sigma} \|\hat{z} - \mathcal{M}p\|_2$
- Step 2: $\hat{\theta} \in \arg\min_{\theta} R(\tilde{p}, \theta)$

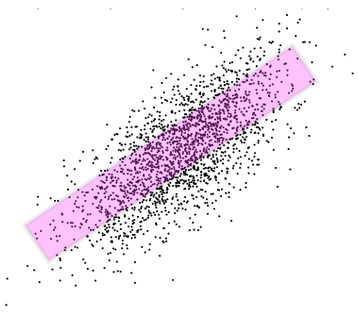
Guarantees?

- Assume: LRIP $\quad \|p - p'\|_\ell \lesssim \|\mathcal{M}(p) - \mathcal{M}(p')\|_2, \forall p, p' \in \Sigma$
- Then: excess risk control $R(p^\star, \hat{\theta}) - R(p^\star, \theta^\star) \lesssim \|\mathcal{M}(\hat{p}_n - p^\star)\|_2 + d(p^\star, \Sigma)$

Agenda

- From Statistical Learning to Compressive Learning
- Worked Examples:
  - *Compressive PCA*
  - *Compressive K-means*
  - *Compressive Gaussian Mixture Modeling*
- Conclusion and perspectives

Compressive PCA



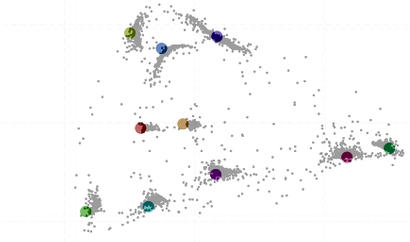
- **Loss function:** $\ell(x, \theta) = \|x - P_\theta x\|^2 \qquad x \in \mathbb{R}^d$
- **An hypothesis** = a *k*-dimensional subspace θ
- **Model set** Σ **:**
  - distributions supported on a *k*-dimensional subspace
- **Sketch:**
  - **naive** = full covariance matrix, $\mathbb{E}_{X \sim p} X X^T$ $\quad m = \mathcal{O}(d^2)$
  - using compressive matrix sensing $\quad m = \mathcal{O}(kd)$

  LRIP proved
- **Learn from sketch:**
  - low-rank matrix recovery

Compressive K-Means (clustering)

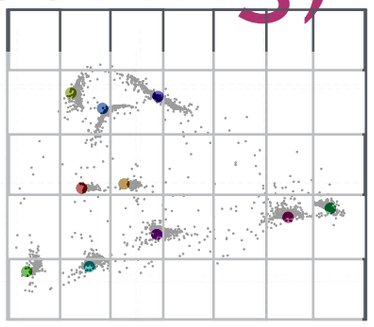
- **Loss function:** $\ell(x, \theta) = \min_i \|x - \theta_i\|^2$
- **An hypothesis = *k* centroids** $\theta = \{\theta_1, \ldots, \theta_k\},\ \theta_i \in \mathbb{R}^d$
- **Standard approach: “K-means algorithm”**
  - *aka* Lloyd-Max algorithm [Steinhaus 1956, Lloyd 1957 (publ. 1982)]
    - several passes on the training set

Compressive K-Means (clustering)



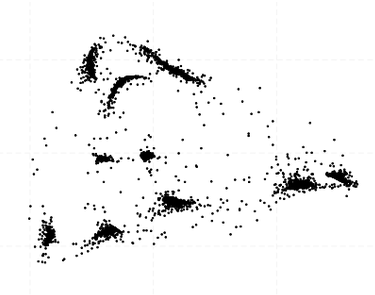
- **Loss function:** $\ell(x, \theta) = \min_i \|x - \theta_i\|^2$
- **An hypothesis = *k* centroids** $\theta = \{\theta_1, \ldots, \theta_k\},\ \theta_i \in \mathbb{R}^d$
- **Standard approach: "K-means algorithm"**
  - *aka* Lloyd-Max algorithm [Steinhaus 1956, Lloyd 1957 (publ. 1982)]
    - several passes on the training set
- **Model set** Σ
  - mixtures of *k Diracs*

Compressive K-Means (clustering)



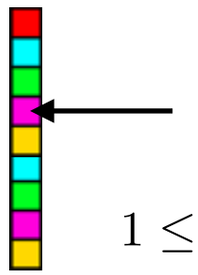
- **Loss function:** $\ell(x, \theta) = \min_i \|x - \theta_i\|^2$
- **An hypothesis = *k* centroids** $\theta = \{\theta_1, \ldots, \theta_k\},\ \theta_i \in \mathbb{R}^d$
- **Standard approach: "K-means algorithm"**
  - *aka* Lloyd-Max algorithm [Steinhaus 1956, Lloyd 1957 (publ. 1982)]
    - several passes on the training set
- **Model set** Σ
  - mixtures of *k Diracs*
- **Naive sketching =histograms**
  - $N = \mathcal{O}((R/\epsilon)^d)$ bins of size ϵ within domain of size *R*
    - sketch size $m = N$
    - compressive sensing suggests $m = \mathcal{O}(k \log N) = \mathcal{O}(kd \log(R/\epsilon))$

Compressive K-means: How to Sketch ?



- Which moments to choose ?

  - **Observation**: distribution *p(x)* is ***spatially localized***
  - **Intuition (from compressive sensing)**
    - need "**incoherent**" sampling
    - choose **Fourier** measurements
    - = **characteristic function of** $\hat{p}_n$



$$z_\ell \approx \mathbb{E}_{X\sim p} e^{j w_\ell^\top X} \qquad \omega_\ell \in \mathbb{R}^d$$

$$1 \le \ell \le m$$

- *Sketching function* $\Phi(x) = \frac{1}{\sqrt{m}} \left(e^{j\omega_\ell^\top x} \right)_{\ell=1}^{m}$ **Random Fourier Features** *[Rahimi & Recht 2007]*
- *Sketch vector* z = **Random Fourier *Moments***

Learning centroids from a sketch ?

From theory:

- Learning from sketch vector
  - Step 1:

$$\tilde{p} = \Delta(\hat{z}) = \arg\min_{p \in \Sigma} \|\hat{z} - \mathcal{M}p\|_2$$

  - Step 2:

$$\hat{\theta} \in \arg\min_{\theta} R(\tilde{p}, \theta)$$

- Model set
  - mixtures of *k Diracs* $p = \sum_{j=1}^{k} \alpha_j \delta_{\theta_j}$

$$\mathcal{M}(p) = \sum_{j=1}^{k} \alpha_j \Phi(\theta_j)$$

Instantiation

- Optimization problem
  - Step 1:

$$(\tilde{\alpha}_j, \tilde{\theta}_j) = \arg\min_{\alpha_j, \theta_j} \|\hat{z} - \sum_{j=1}^{k} \alpha_j \Phi(\theta_j)\|_2$$

  - Step 2:

$$(\hat{\theta}_j) = (\tilde{\theta}_j)$$

Learning centroids from a sketch ?

From theory:

- Learning from sketch vector
  - Step 1:

$$\tilde{p} = \Delta(\hat{z}) = \arg\min_{p \in \Sigma} \|\hat{z} - \mathcal{M}p\|_2$$

  - Step 2:

$$\hat{\theta} \in \arg\min_{\theta} R(\tilde{p}, \theta)$$

- Model set
  - mixtures of *k Diracs* $p = \sum_{j=1}^{k} \alpha_j \delta_{\theta_j}$

$$\mathcal{M}(p) = \sum_{j=1}^{k} \alpha_j \Phi(\theta_j)$$

Instantiation

- Optimization problem
  - Step 1:

$$(\tilde{\alpha}_j, \tilde{\theta}_j) = \arg\min_{\alpha_j, \theta_j} \|\hat{z} - \sum_{j=1}^{k} \alpha_j \Phi(\theta_j)\|_2$$

  - Step 2:

$$(\hat{\theta}_j) = (\tilde{\theta}_j)$$

Ideal decoding scheme

- "Highly" non-convex
- Two approaches:
  - Discretize + convex relaxation *[Bunea & al 2010]*
  - **Greedy & gridless**

Learning centroids from a sketch ?

From theory:

- Learning from sketch vector
  - Step 1:

$$\tilde{p} = \Delta(\hat{z}) = \arg\min_{p \in \Sigma} \|\hat{z} - \mathcal{M}p\|_2$$

  - Step 2:

$$\hat{\theta} \in \arg\min_{\theta} R(\tilde{p}, \theta)$$

- Model set
  - mixtures of *k Diracs* $p = \sum_{j=1}^{k} \alpha_j \delta_{\theta_j}$

$$\mathcal{M}(p) = \sum_{j=1}^{k} \alpha_j \Phi(\theta_j)$$

Instantiation

- Optimization problem
  - Step 1:

$$(\tilde{\alpha}_j, \tilde{\theta}_j) = \arg\min_{\alpha_j, \theta_j} \|\hat{z} - \sum_{j=1}^{k} \alpha_j \Phi(\theta_j)\|_2$$

  - Step 2:

$$(\hat{\theta}_j) = (\tilde{\theta}_j)$$

Ideal decoding scheme

- "Highly" non-convex
- Two approaches:
  - Discretize + convex relaxation *[Bunea & al 2010]*
  - **Greedy & gridless**

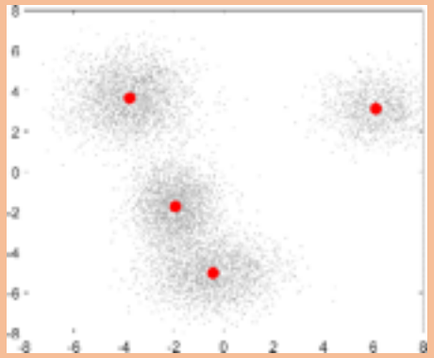
`MP (Mallat & Zhang 93) > OMP (Pati & al 93) > OMPR (Jain 2011) > CL-OMPR`

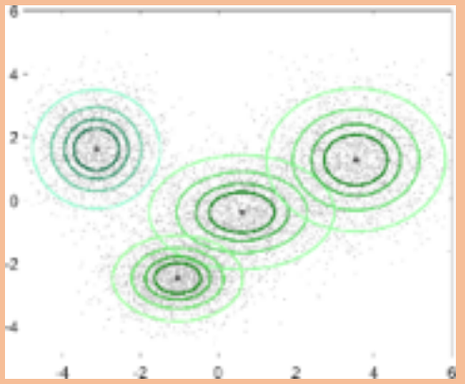
Analogous to Frank-Wolfe, see e.g. [Bredies & al 2013]

The SketchMLbox

SketchMLbox (**`sketchml.gforge.inria.fr`**)

- Mixture of Diracs (« K-means »)
- GMMs with known covariance



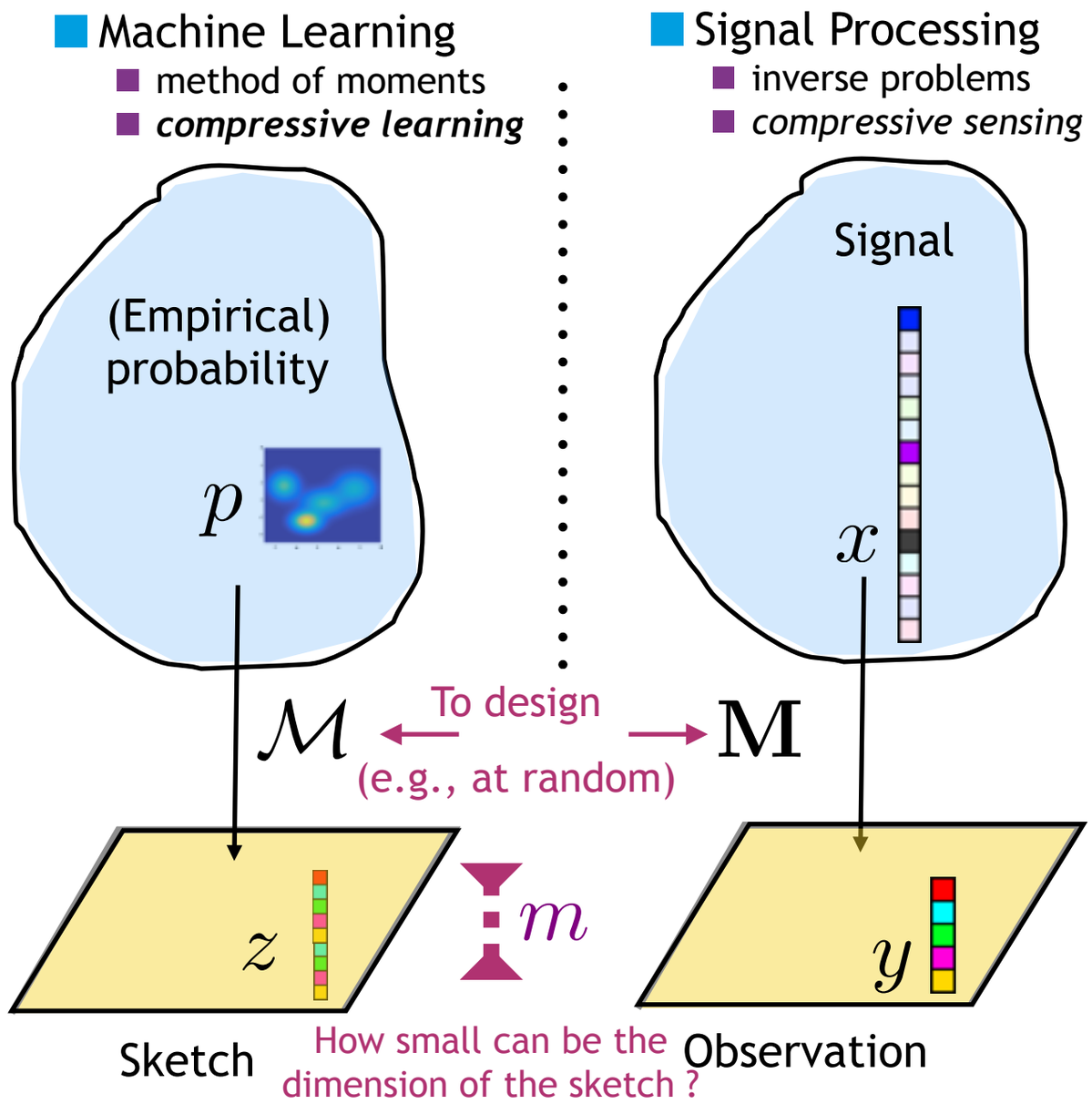


- GMMs with unknown diagonal covariance
- **Soon**:
  - Mixtures of alpha-stable
  - Gaussian Locally Linear Mapping *[Deleforge 2014]*

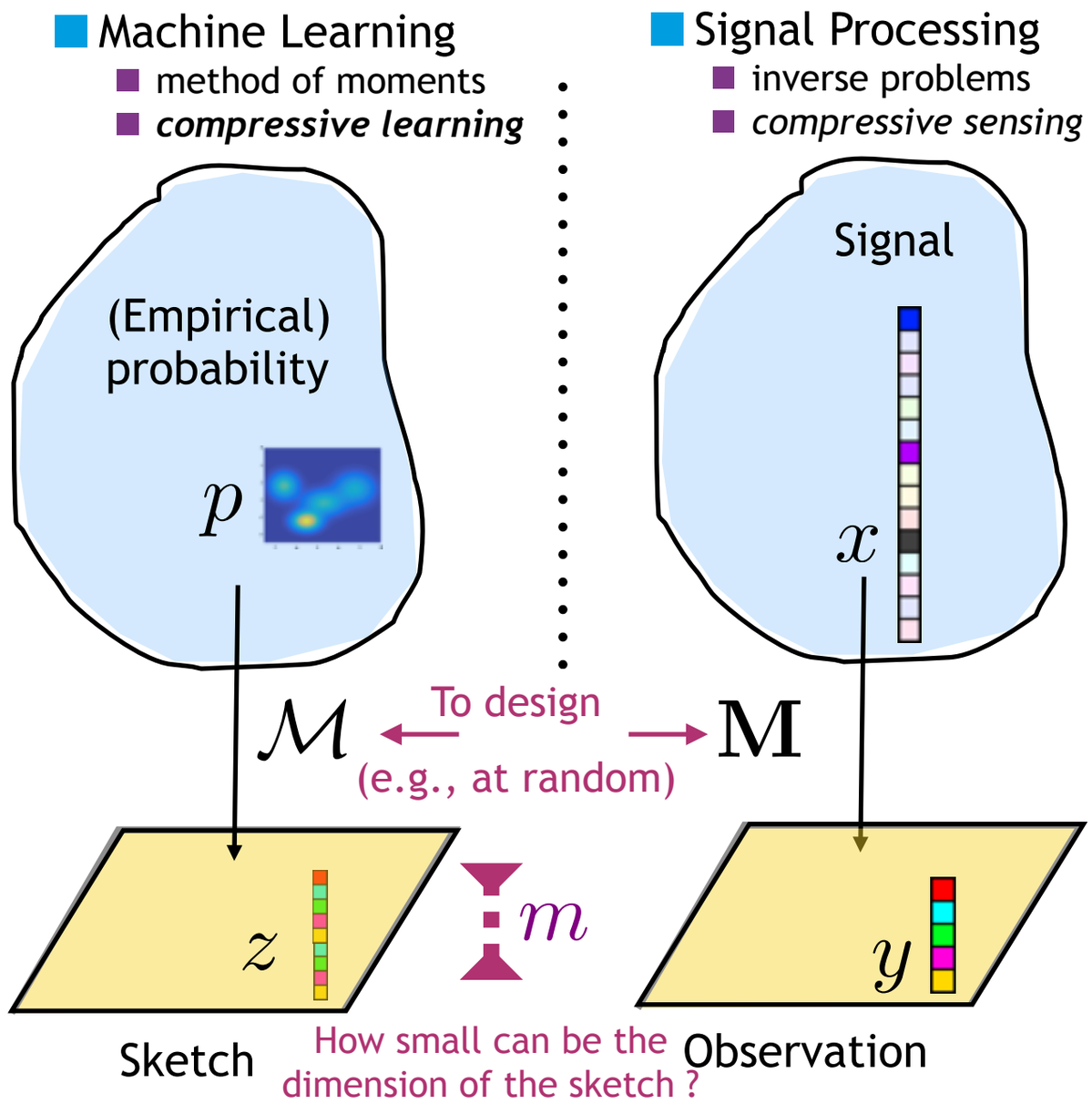
- **Handles generic mixtures** $p = \sum_{j=1}^{k} \alpha_j p_{\theta_j}$

  with user-defined $\mathcal{M}p_\theta, \nabla_\theta \mathcal{M} p_\theta$

Back to Question 2: Dimension reduction ?



Back to Question 2: Dimension reduction ?



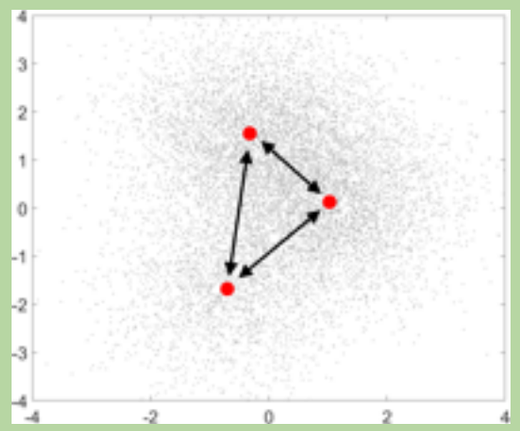
Theorem: K-medians

Goal $\underset{\Theta \in \mathbf{\Theta}}{\text{minimize}}$ $R(p^\star, \Theta) = \mathbb{E}_{X \sim p^\star} \left[\min_{j=1}^{k} \|X - \theta_j\|_2 \right]$ *(expected risk)*

Theorem: K-medians

Goal $\underset{\Theta \in \boldsymbol{\Theta}}{\text{minimize}}$ $R(p^\star, \Theta) = \mathbb{E}_{X \sim p^\star} \left[\min_{j=1}^{k} \|X - \theta_j\|_2 \right]$ *(expected risk)*

Hyp. $\boldsymbol{\Theta}_{\epsilon, M}$



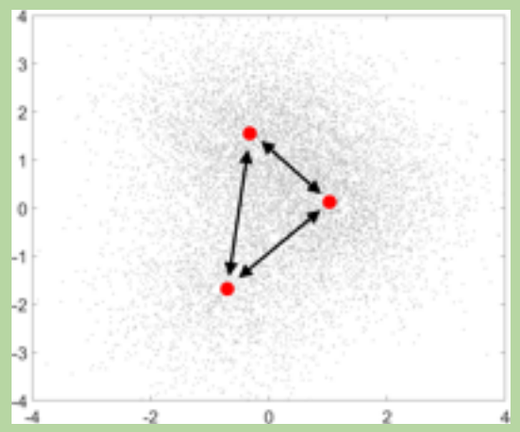
- M - bounded domain in $\mathbb{R}^d$
- ε - separation

  *(= assumptions on the k centroids, **not samples**)*

Theorem: K-medians

Goal $\underset{\Theta \in \boldsymbol{\Theta}}{\text{minimize}}$ $R(p^\star, \Theta) = \mathbb{E}_{X \sim p^\star} \left[\min_{j=1}^{k} \|X - \theta_j\|_2 \right]$ *(expected risk)*

Hyp. $\boldsymbol{\Theta}_{\epsilon, M}$



- M - bounded domain in $\mathbb{R}^d$
- $\mathcal{E}$ - separation

*(= assumptions on the k centroids, **not samples**)*

Sketching $\mathcal{M}$

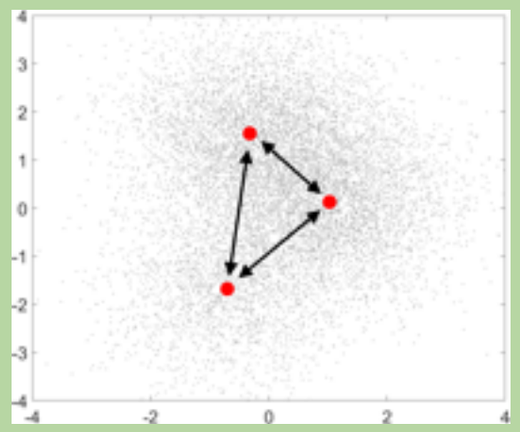
Reweighted random Fourier moments

(needed for theory, no effect in practice)

Theorem: K-medians

Goal $\underset{\Theta \in \mathbf{\Theta}}{\text{minimize}}$ $R(p^\star, \Theta) = \mathbb{E}_{X \sim p^\star} \left[\min_{j=1}^{k} \|X - \theta_j\|_2 \right]$ *(expected risk)*

Hyp. $\mathbf{\Theta}_{\epsilon, M}$



- M - bounded domain in $\mathbb{R}^d$
- $\mathcal{E}$ - separation

*(= assumptions on the k centroids, **not samples**)*

Sketching $\mathcal{M}$

Reweighted random Fourier moments
(needed for theory, no effect in practice)

If $m \geq \mathcal{O}(k^2 d^2 \log^2 k (\log(kd) + \log(M/\epsilon)))$

then w.h.p., for *any* $p^\star$

$$R(\hat{\Theta}) - R(\Theta^\star) \lesssim d(p^\star, \Sigma) + O(\sqrt{1/n})$$

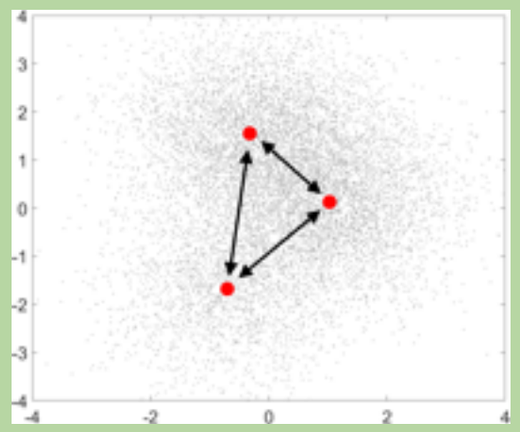
Minimize $\|z - \mathcal{M} p_{\Theta,\alpha}\|_2$ (with hyp.)

Minimize expected risk (with hyp.)

Theorem: K-medians

Goal minimize $\Theta \in \mathbf{\Theta}$ $R(p^\star, \Theta) = \mathbb{E}_{X \sim p^\star} \left[\min_{j=1}^{k} \|X - \theta_j\|_2 \right]$ *(expected risk)*

Hyp. $\mathbf{\Theta}_{\epsilon, M}$



- M - bounded domain in $\mathbb{R}^d$
- $\mathcal{E}$ - separation

*(= assumptions on the k centroids, **not samples**)*

Sketching $\mathcal{M}$

Reweighted random Fourier moments
(needed for theory, no effect in practice)

If $m \geq \mathcal{O}(k^2 d \ \log^2 k(\log(kd) + \log(M/\epsilon)))$

then w.h.p., for *any* $p^\star$

$$R(\hat{\Theta}) - R(\Theta^\star) \lesssim d(p^\star, \Sigma) + O(\sqrt{1/n})$$

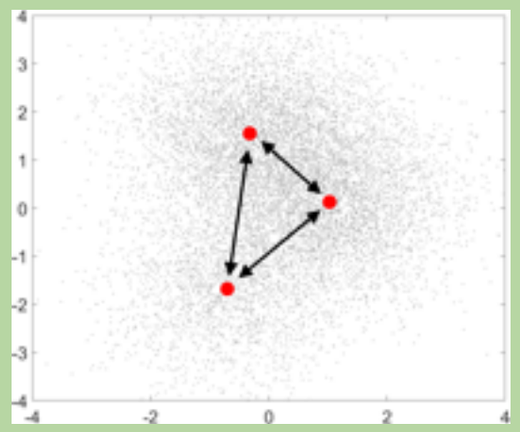
Minimize $\|z - \mathcal{M} p_{\Theta,\alpha}\|_2$ (with hyp.)

Minimize expected risk (with hyp.)

Theorem: K-medians

Goal minimize $\Theta \in \mathbf{\Theta}$ $R(p^\star, \Theta) = \mathbb{E}_{X \sim p^\star} \left[\min_{j=1}^{k} \|X - \theta_j\|_2 \right]$ *(expected risk)*

Hyp. $\mathbf{\Theta}_{\epsilon, M}$



- M - bounded domain in $\mathbb{R}^d$
- $\mathcal{E}$ - separation

*(= assumptions on the k centroids, **not samples**)*

Sketching $\mathcal{M}$

Reweighted random Fourier moments
(needed for theory, no effect in practice)

If $m \geq \mathcal{O}(k^2 d \ \log^2 k(\log(kd) + \log(M/\epsilon)))$

then w.h.p., for *any* $p^\star$

by proving the LRIP

$$R(\hat{\Theta}) - R(\Theta^\star) \lesssim d(p^\star, \Sigma) + O(\sqrt{1/n})$$

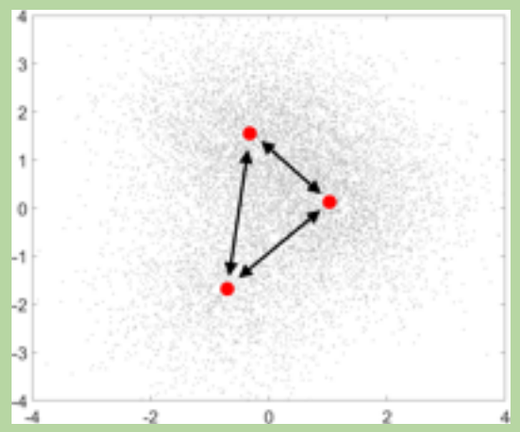
Minimize $\|z - \mathcal{M} p_{\Theta,\alpha}\|_2$ (with hyp.)

Minimize expected risk (with hyp.)

Theorem: K-medians

Goal minimize $\Theta \in \mathbf{\Theta}$ $R(p^\star, \Theta) = \mathbb{E}_{X \sim p^\star} \left[\min_{j=1}^{k} \|X - \theta_j\|_2 \right]$ *(expected risk)*

Hyp. $\mathbf{\Theta}_{\epsilon, M}$



- M - bounded domain in $\mathbb{R}^d$
- ε - separation

*(= assumptions on the k centroids, **not samples**)*

Sketching $\mathcal{M}$

Reweighted random Fourier moments
(needed for theory, no effect in practice)

If $m \geq \mathcal{O}(k^2 d \ \log^2 k(\log(kd) + \log(M/\epsilon)))$

then w.h.p., for *any* $p^\star$

$$R(\hat{\Theta}) - R(\Theta^\star) \lesssim d(p^\star, \Sigma) + O(\sqrt{1/n}) \lesssim R(\Theta^\star)$$

Using optimal transport

by proving the LRIP

Minimize $\|z - \mathcal{M} p_{\Theta,\alpha}\|_2$ (with hyp.)

Minimize expected risk (with hyp.)

#measurements: phase transitions ?

In theory, sufficient to have

$$m \gtrsim \mathcal{O}(k^2 d)$$

Empirically ?

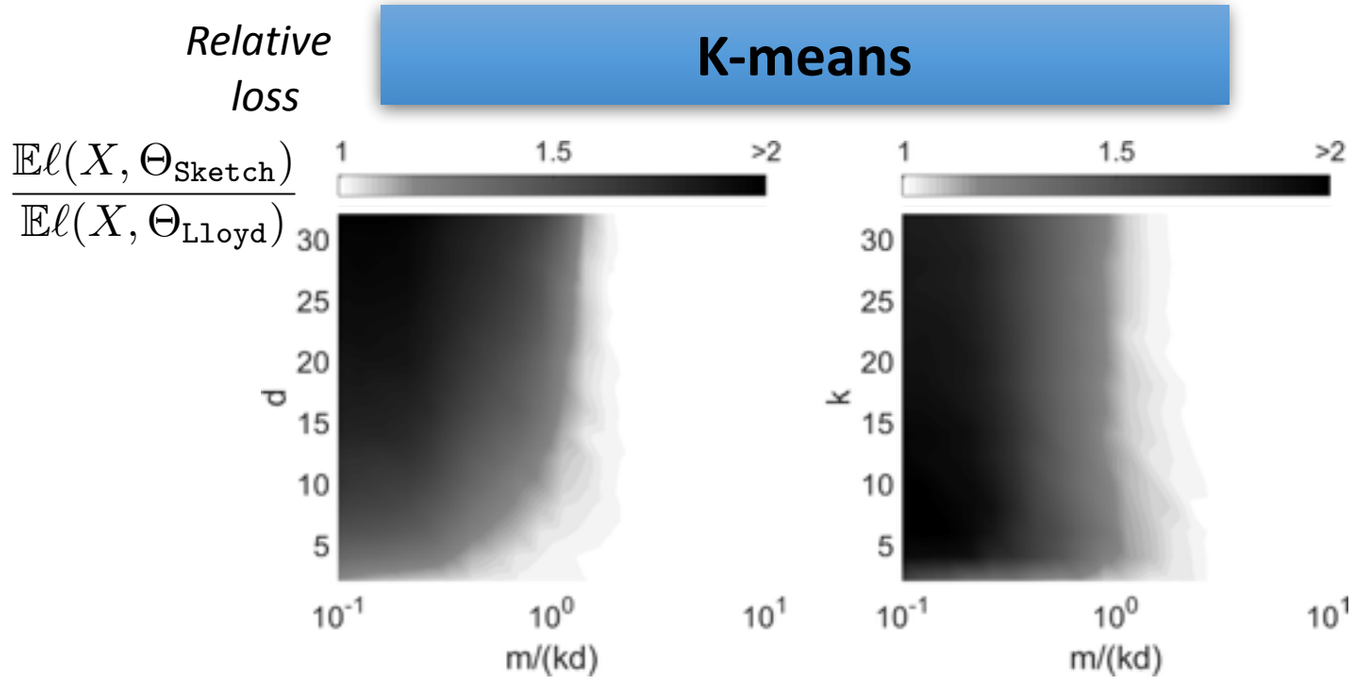
#measurements: phase transitions ?

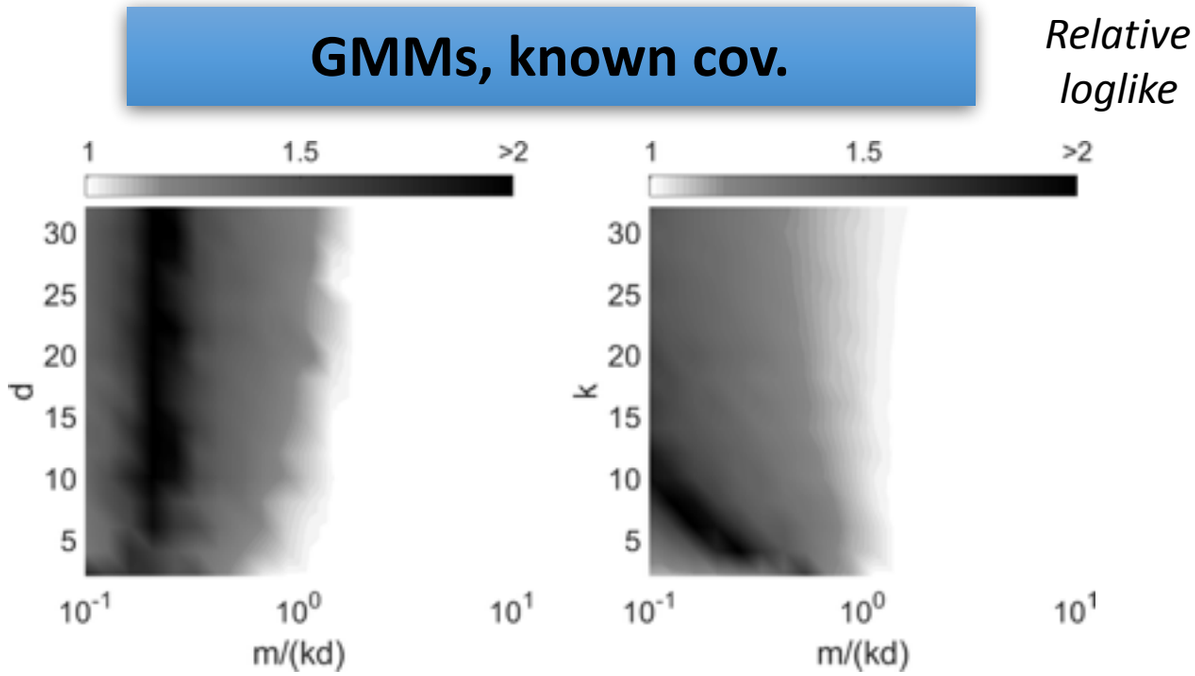
In theory, sufficient to have

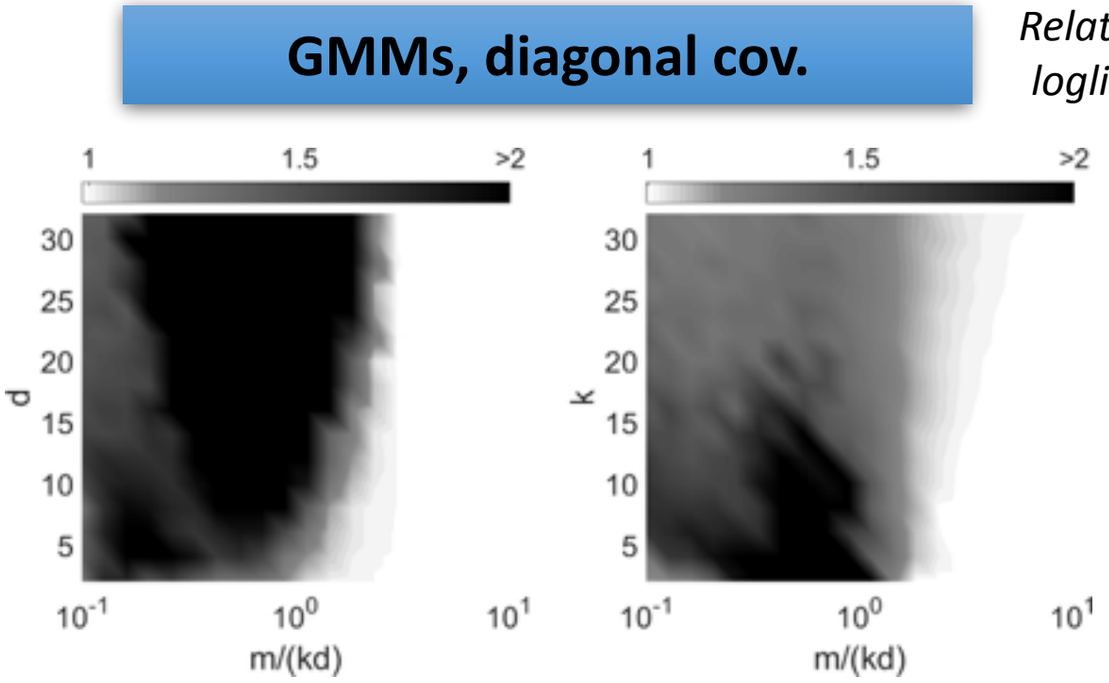
$$m \gtrsim \mathcal{O}(k^2 d)$$

Empirically ?

$$m \approx \mathcal{O}(kd)$$







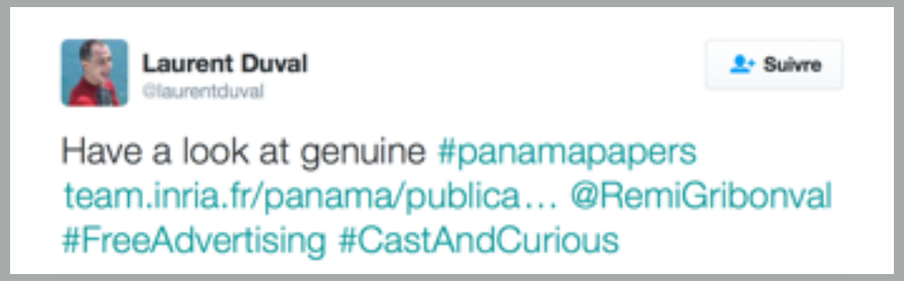
PLEASE

projection, learning and sparsity for efficient data processing

Conclusion

Compressive learning with random moments

- **Promises**
  - controlled resources (memory, flops)
  - distributed, streamed computations
  - information preservation
  - … privacy preservation ?
  - … provably good & efficient learning algorithms ?
- **Applications**
  - compressive PCA
  - compressive clustering (k-means, k-medians)
    - surprising links with line spectral estimation / “super-resolution” / off-the-grid methods
  - compressive mixture modeling (Gaussians, alpha-stable, …)
  - … dictionary learning, supervised classification ?
  - … universal sketching functions ?
- **Details**
  - toolbox `sketchml.gforge.inria.fr`
  - preprint `arxiv.org/abs/1706.07180`



PLEASE

projection, learning and sparsity for efficient data processing

PANAMA

Parsimony and New Algorithms for Audio and Signal Modeling

Presentation | Team members | Papers | Events | Projects | News | Job offers | Software | Contact | Intranet

Papers

2016

Journal articles

Flexible Multi-layer Sparse Approximations of Matrices and Applications

Luc Le Magoarou, Rémi Gribonval

IEEE Journal of Selected Topics in Signal Processing, IEEE, 2016, <10.1109/JSTSP.2016.2543461>

Random sampling of bandlimited signals on graphs

Gilles Puy, Nicolas Tremblay, Rémi Gribonval, Pierre Vandergheynst

Applied and Computational Harmonic Analysis, Elsevier, 2016, <10.1016/j.acha.2016.05.005>

Fast Robust PCA on Graphs

Nauman Shahid, Nathanael Perraudin, Vassilis Kalofolias, Gilles Puy, Pierre Vandergheynst

IEEE Journal of Selected Topics in Signal Processing, IEEE, 2016, 10 (4), pp.740 – 756, <10.1109/JSTSP.2016.2555239>

English | Français

News

(closed) PhD offer – Interactive Navigation for a Video Audio True Experience @ PANAMA, Inria Rennes 2016/07/08

Paper + code on random sampling of bandlimited signals on graphs 2016/06/16

2016 Award for Outstanding Contributions in Neural Systems 2016/05/25

(closed) PhD offer – Estimating the Geometry of Audio Scenes Using Virtually-Supervised Learning @ Inria Rennes 2016/05/24

Paper + code on Compressive Spectral Clustering 2016/05/22

PLEASE

projection, learning and sparsity for efficient data processing

THANKS

erc