Sketching for Large-Scale Learning of Mixture Models

Nicolas Keriven

Université Rennes 1
Ecole doctorale MATISSE

IRISA (CNRS/UMR 6074), Team PANAMA
Advisor: Rémi Gribonval

Thesis defense – 2017 October 12th
Context: machine learning

Database

Automatic task

- Clustering
- Classification
- etc...

= cat
Context: machine learning

**Large database**

Learning

**Automatic task**

- Clustering
- Classification
- etc...

= cat

12/10/2017

Nicolas Keriven
Context: machine learning

Large database

Learning

Automatic task

- Clustering
- Classification
- etc...

Large elements
Billions of elements

Learning

12/10/2017
Nicolas Keriven
Context: machine learning

**Large database**

**Large elements**

*Billions of elements*

**Learning**

**Slow, costly**

**Automatic task**

- Clustering
- Classification
- etc...

= cat
Context: machine learning

Large database

Large elements
Billions of elements

Distributed database

Learning
Slow, costly

Automatic task
- Clustering
- Classification
- etc...

12/10/2017
Nicolas Keriven
Context: machine learning

- Large database
  - Large elements
    - Billions of elements

- Distributed database

- Data Stream

Learning

- Automatic task
  - Clustering
  - Classification
  - etc...

12/10/2017

Nicolas Keriven
Context: machine learning

Large database

Learning

Slow, costly

Distributed database

Automatic task

- Clustering
- Classification
- etc...

Idea!

Small intermediate representation

Data Stream
Context: machine learning

**Large database**

- Large elements
- Billions of elements

**Distributed database**

**Data Stream**

**Automatic task**

- Clustering
- Classification
- etc...

**Learning**

- Slow, costly

**Idea!**

1: Compression

**Small** intermediate representation

12/10/2017
Nicolas Keriven
Context: machine learning

Large database
- Large elements
- Billions of elements
- Learning
  - Slow, costly

Distributed database

Data Stream

Large database

Automatic task
- Clustering
- Classification
- etc...

Small intermediate representation

1: Compression

2: Learning

Idea!

12/10/2017
Nicolas Keriven
Context: machine learning

**Large database**
- Large elements
- Billions of elements

**Distributed database**

**Data Stream**

**Automatic task**
- Clustering
- Classification
- etc...

**Learning**
- Slow, costly

**1: Compression**
- Small intermediate representation

**Desired properties**
- Fast to compute (distributed, streaming, GPU...)
- Preserve desired information
- Preserve data privacy

**2: Learning**

**Idea!**
Three compression schemes
Three compression schemes

Database

Feature extraction

Data = Collection of vectors

Compression?
Three compression schemes

**Database**

Data = Collection of vectors

Feature extraction

Compression?

\[ x_1 \ x_2 \ \ldots \ x_n \]

\[ x'_1 \ x'_2 \ \ldots \ x'_n \]

**Dimensionality reduction**

See eg [Calderbank 2009, Boutsidis 2010]

- Random Projection
- Feature selection
Three compression schemes

Database

Data = Collection of vectors

Feature extraction

Subsampling coresets
See eg [Feldman 2010]

- Random Projection
- Feature selection

Dimensionality reduction
See eg [Calderbank 2009, Boutsidis 2010]

Compression ?

Compression ?

Random Projection

- Uniform sampling (naive)
- Adaptive sampling...

Subsampling coresets

- Random Projection
- Feature selection

Dimensionality reduction

See eg [Calderbank 2009, Boutsidis 2010]
Three compression schemes

Database

Feature extraction

Data = Collection of vectors

Subsampling coresets
See eg [Feldman 2010]

Dimensionality reduction
See eg [Calderbank 2009, Boutsidis 2010]
- Random Projection
- Feature selection

Linear sketch
See [Thaper 2002]
[Cormode 2011]
- Distributed, streaming
- Hash tables, histograms
- Sketching for learning ?

- Uniform sampling (naive)
- Adaptive sampling...
Isotropic GMM estimation [Bourrier 2013]

Practical illustration: sketched Gaussian Mixture Model estimation with Id cov. [Bourrier 2013]

Data

\[ x_1, \ldots, x_n \in \mathbb{R}^d \]
Isotropic GMM estimation \cite{Bourrier2013}

**Practical illustration:** sketched Gaussian Mixture Model estimation with Id cov. \cite{Bourrier2013}

Data $x_1, \ldots, x_n \in \mathbb{R}^d$ → **Random Fourier moments** → Sketch
Isotropic GMM estimation [Bourrier 2013]

**Practical illustration:** sketched Gaussian Mixture Model estimation with Id cov. [Bourrier 2013]

Data: $x_1, \ldots, x_n \in \mathbb{R}^d$

Random Fourier moments → Sketch → Modified Iterative Hard Thresholding

$$\sum_{l=1}^k w_l \mathcal{N}(\mu_l, I)$$
Isotropic GMM estimation \cite{Bourrier2013}

**Practical illustration**: sketched Gaussian Mixture Model estimation with \text{Id cov.} \cite{Bourrier2013}

\[ x_1, \ldots, x_n \in \mathbb{R}^d \]

Random Fourier moments \rightarrow Sketch \rightarrow Modified Iterative Hard Thresholding

\[ \sum_{l=1}^{k} w_l \mathcal{N}(\mu_l, \text{I}) \]

**Observation**: necessarily...

Any \textit{linear} sketch = empirical moments

\[
\hat{z} = \hat{E} \Phi(X) = \frac{1}{n} \sum_{i} \Phi(x_i)
\]
Isotropic GMM estimation [Bourrier 2013]

**Practical illustration:** sketched Gaussian Mixture Model estimation with Id cov. [Bourrier 2013]

Data

$x_1, \ldots, x_n \in \mathbb{R}^d$

Random Fourier moments

Modified Iterative Hard Thresholding

Sketch

Observation: necessarily...

Any *linear* sketch = empirical moments

$$\hat{z} = \hat{E} \Phi(X) = \frac{1}{n} \sum_i \Phi(x_i)$$

$$\Phi : \mathbb{R}^d \rightarrow \mathbb{C}^m$$

Data 3/28

Observation: necessarily...
Isotropic GMM estimation [Bourrier 2013]

**Practical illustration:** sketched Gaussian Mixture Model estimation with \(\text{Id} \) cov. [Bourrier 2013]

---

**Data**

\(x_1, \ldots, x_n \in \mathbb{R}^d\)

---

**Random Fourier moments**

---

**Sketch**

\[\sum_{l=1}^k w_l \mathcal{N}(\mu_l, \mathbb{I})\]

---

**Modified Iterative Hard Thresholding**

---

**Observation:** necessarily...

**Any linear sketch** = empirical moments

\[
\hat{z} = \hat{E} \Phi(X) = \frac{1}{n} \sum_i \Phi(x_i)
\]

\(\Phi : \mathbb{R}^d \rightarrow \mathbb{C}^m\)

---

**... hence:**

**Sketch learning** = moment matching

\[
\min_{\theta} \| \hat{z} - \mathbb{E}_\theta \Phi(X) \|
\]

**True moments (param. \(\theta\))**
**Isotropic GMM estimation** [Bourrier 2013]

**Practical illustration:** sketched Gaussian Mixture Model estimation with Id cov. [Bourrier 2013]

Data $x_1, ..., x_n \in \mathbb{R}^d$

**Random Fourier moments**

**Modified Iterative Hard Thresholding**

Sketch

$\sum_{i=1}^{k} w_i \mathcal{N}(\mu_l, \mathbf{I})$

**Observation:** necessarily...

Any **linear** sketch = empirical moments

$$\hat{z} = \hat{E} \Phi(X) = \frac{1}{n} \sum_i \Phi(x_i)$$

$$\Phi : \mathbb{R}^d \rightarrow \mathbb{C}^m$$

**... hence:**

Sketch learning = moment matching

$$\min_\theta \| \hat{z} - \mathbb{E}_\theta \Phi(X) \|$$

**Good empirical properties of the « sketching » function $\Phi$**

- « Sufficient » dimension $m$ (size of the sketch)
- Randomly designed
Questions
Questions

- Generalize to other mixture models? New algorithm?
- Theoretical guarantees?
Contributions

Questions

- Generalize to other mixture models? New algorithm?
- Theoretical guarantees?

Contributions of this thesis
Questions

- Generalize to other mixture models? New algorithm?
- Theoretical guarantees?

Contributions of this thesis

- **Algorithmic**: heuristic greedy algorithm for any sketched mixture model estimation
  - General GMM estimation
  - Sketched k-means
  - Mixture of multivariate elliptic $\alpha$-stable distributions estimation
Contributions

Questions

- Generalize to other mixture models? New algorithm?
- Theoretical guarantees?

Contributions of this thesis

- **Algorithmic**: heuristic greedy algorithm for any sketched mixture model estimation
  - General GMM estimation
  - Sketched k-means
  - Mixture of multivariate elliptic $\alpha$-stable distributions estimation

- **Theoretical**: Information-preservation guarantees
  - Recovery conditions for generic models
  - Additional focus on mixture models
1. Sketched Mixture Model Estimation
   1.1 A flexible greedy algorithm
   1.2 Experiments

2. Information-preservation guarantees
   2.1 Generic analysis
   2.2 Statistical Learning with sketches of limited size

3. Conclusion
Outline

1. Sketched Mixture Model Estimation
   1.1 A flexible greedy algorithm
   1.2 Experiments

2. Information-preservation guarantees
   2.1 Generic analysis
   2.2 Statistical Learning with sketches of limited size

3. Conclusion
Sketched mixture model estimation

\[ \hat{z} = \hat{E}(\Phi(X)) \]
Sketched mixture model estimation

Goal

- Estimate mixture model:

\[ x_i \sim \sum_{l=1}^{k} w_l \pi_{\theta_l} \]

\[ w_l \geq 0, \quad \sum_l w_l = 1 \]

Ex: \( \pi_{\theta} = \mathcal{N}(\mu, \Sigma) \)
Estimate mixture model:

from sketch

Ex: \( \pi_\theta = \mathcal{N}(\mu, \Sigma) \)

Goal

- Estimate mixture model:

\[
x_i \sim \sum_{l=1}^{k} w_l \pi_{\theta_l}
\]

\[
w_l \geq 0, \quad \sum_l w_l = 1
\]

from sketch \( \hat{z} = \hat{E} \Phi(X) \)
**Goal**

- Estimate mixture model:

\[
\mathbf{x}_i \sim \sum_{l=1}^{k} w_l \pi_{\theta_l} \\
\text{subject to } w_l \geq 0, \sum_l w_l = 1
\]

from sketch \( \hat{\mathbf{z}} = \hat{\mathbf{E}} \Phi(\mathbf{X}) \)

**Ex:** \( \pi_{\theta} = \mathcal{N}(\mu, \Sigma) \)

**Method: moment matching**
Sketched mixture model estimation

**Goal**
- Estimate mixture model:
  \[ x_i \sim \sum_{l=1}^{k} w_l \pi_{\theta_l} \]
  \[ w_l \geq 0, \sum_l w_l = 1 \]

  from sketch \[ \hat{z} = \hat{E} \Phi (X) \]

  Ex: \[ \pi_{\theta} = \mathcal{N}(\mu, \Sigma) \]

**Method: moment matching**

Written as

\[ \min_{\theta_l, w} \| \hat{z} - \sum_{l=1}^{k} w_l f(\theta_l) \|_2 \]

where

\[ f(\theta) := \mathbb{E}_{X \sim \pi_{\theta}} \Phi (X) \]
Estimate mixture model:

\[ x_i \sim \sum_{l=1}^{k} w_l \pi_{\theta_l} \]

\[ \sum_l w_l = 1 \]

\[ w_l \geq 0 \]

\[ \hat{z} = \hat{E} \Phi(X) \]

**Goal**

- Estimate mixture model:

**Method: moment matching**

Written as

\[ \min_{\theta_l, w} \| \hat{z} - \sum_{l=1}^{k} w_l f(\theta_l) \|_2 \]

where

\[ f(\theta) := E_{X \sim \pi_{\theta}} \Phi(X) \]

**Non-convex minimization**

**Ex: \( \pi_{\theta} = N(\mu, \Sigma) \)**
Sketched mixture model estimation

Goal

- Estimate mixture model:

\[ x_i \sim \sum_{l=1}^{k} w_l \pi_{\theta_l} \]

\[ w_l \geq 0, \quad \sum_l w_l = 1 \]

from sketch \( \hat{z} = \hat{E}\Phi(X) \)

Ex: \( \pi_{\theta} = \mathcal{N}(\mu, \Sigma) \)

Method: moment matching

Written as

\[ \min_{\theta_l, w} \| \hat{z} - \sum_{l=1}^{k} w_l f(\theta_l) \|_2 \]

where

\[ f(\theta) := \mathbb{E}_{X \sim \pi_{\theta}} \Phi(X) \]

- Non-convex minimization
- Convex relaxation? (super-resolution)
Estimate mixture model:

\[
\mathbf{x}_i \sim \sum_{l=1}^{k} w_l \pi_{\theta_l}
\]

\[w_l \geq 0, \quad \sum_l w_l = 1\]

from sketch \(\hat{\mathbf{z}} = \hat{\mathbb{E}} \Phi (\mathbf{X})\)

Ex: \(\pi_{\theta} = \mathcal{N}(\mu, \Sigma)\)

**Goal**

**Method: moment matching**

Written as

\[
\min_{\theta_l, w} \| \hat{\mathbf{z}} - \sum_{l=1}^{k} w_l f(\theta_l) \|_2
\]

where

\[
f(\theta) := \mathbb{E}_{\mathbf{X} \sim \pi_{\theta}} \Phi (\mathbf{X})
\]

- Non-convex minimization
- Convex relaxation? (super-resolution)
- Proposed approach: greedy heuristic
Algorithm: Compressive Learning OMPR (CL-OMPR)

Continuous (off-the-grid) adaptation of Orthogonal Matching Pursuit with Replacement

[Jain 2011]
Algorithm: Compressive Learning OMPR (CL-OMPR)

Continuous (off-the-grid) adaptation of *Orthogonal Matching Pursuit with Replacement*  

\[ \hat{z} = \hat{E} \Phi(X) \]  

[Jain 2011]
Algorithm: Compressive Learning OMPR (CL-OMPR)

Continuous (off-the-grid) adaptation of Orthogonal Matching Pursuit with Replacement

[Jain 2011]
Algorithm: Compressive Learning OMPR (CL-OMPR)
Continuous (off-the-grid) adaptation of Orthogonal Matching Pursuit with Replacement
[Jain 2011]
Algorithm: Compressive Learning OMPR (CL-OMPR)

Continuous (off-the-grid) adaptation of Orthogonal Matching Pursuit with Replacement

[Jain 2011]
Algorithm: Compressive Learning OMPR (CL-OMPR)

Continuous (off-the-grid) adaptation of Orthogonal Matching Pursuit with Replacement

[Jain 2011]

Can be applied if: \( f(\theta) = \mathbb{E}_{\pi_{\theta}} \Phi(X) \) has a closed-form, differentiable expression
Algorithm: Compressive Learning OMPR (CL-OMPR)
Continuous (off-the-grid) adaptation of Orthogonal Matching Pursuit with Replacement [Jain 2011]

Can be applied if: \( f(\theta) = \mathbb{E}_{\pi_{\theta}} \Phi(X) \) has a closed-form, differentiable expression

In experiments:
\( \Phi \): Random Fourier sampling [Bourrier 2013] (with new distribution of frequencies)
Algorithm: Compressive Learning OMPR (CL-OMPR)

Continuous (off-the-grid) adaptation of Orthogonal Matching Pursuit with Replacement [Jain 2011]

Can be applied if: \( f(\theta) = \mathbb{E}_{\pi_{\theta}} \Phi(X) \) has a closed-form, differentiable expression

In experiments:
\( \Phi \): Random Fourier sampling [Bourrier 2013] (with new distribution of frequencies)

Model such that: \( \pi_{\theta} \) has a closed-form characteristic function
1. Sketched Mixture Model Estimation
   1.1 A flexible greedy algorithm
   1.2 Experiments

2. Information-preservation guarantees
   2.1 Generic analysis
   2.2 Statistical Learning with sketches of limited size

3. Conclusion
GMM diagonal cov.

Sketched mixture model estimation
Available at sketchml.gforge.inria.fr

\[ \pi_\theta = \mathcal{N}(\mu, \text{diag}(\sigma)) \]
\[ \theta = (\mu, \sigma) \in \mathbb{R}^{2d} \]

Classic approach on full data

Algorithm: EM
[Dempster 1977]
(VLFeat's gmm)
Models

GMM diagonal cov.

Mixture of Diracs

Sketched mixture model estimation

\[ \pi_\theta = \mathcal{N}(\mu, diag(\sigma)) \]
\[ \theta = (\mu, \sigma) \in \mathbb{R}^{2d} \]

\[ \pi_\theta = \delta_\theta \quad \theta \in \mathbb{R}^d \]
(clustered distribution = noisy mixture of Diracs)

Classic approach on full data

Algorithm: **EM**
[Dempster 1977]
(VLFeat’s gmm)

Algorithm: **k-means**
[Lloyd 1982]
(Matlab’s kmeans)
### Models

**GMM diagonal cov.**

\[ \pi_\theta = \mathcal{N}(\mu, \text{diag}(\sigma)) \]
\[ \theta = (\mu, \sigma) \in \mathbb{R}^{2d} \]

**Mixture of Diracs**

\[ \pi_\theta = \delta_\theta \quad \theta \in \mathbb{R}^d \]
(clustered distribution = noisy mixture of Diracs)

**Mixture of stable dist.**

\[ \pi_\theta = \mathcal{S}_\alpha(\mu, \text{diag}(\sigma)) \]
\[ \theta = (\mu, \sigma, \alpha) \in \mathbb{R}^{2d+1} \]

**Sketched mixture model estimation**

Available at [sketchml.gforge.inria.fr](http://sketchml.gforge.inria.fr)

**Classic approach on full data**

- **Algorithm**: EM
  - [Dempster 1977]
  - *(VLFeat’s gmm)*

- **Algorithm**: k-means
  - [Lloyd 1982]
  - *(Matlab’s kmeans)*

**None!**

1-D method with MCMC: can be very long...
Large-scale evaluation on synthetic data

GMM: \( d = 10, k = 5, m = 500 \)

GMM: \( d = 10, k = 20, m = 2000 \)

Size of database
Large-scale evaluation on synthetic data

GMM: $d = 10$, $k = 5$, $m = 500$

GMM: $d = 10$, $k = 20$, $m = 2000$

- **Does not need** replicates (despite some randomness in CL-OMPR)
Large-scale evaluation on synthetic data

GMM: $d = 10, k = 5, m = 500$

- Does not need replicates (despite some randomness in CL-OMPR)
- Comparatively better on large databases (despite fixed sketch size)

GMM: $d = 10, k = 20, m = 2000$

Size of database

Size of database
Large-scale evaluation on synthetic data

k-means: d = 10, k = 10

Relative error

Size of sketch

12/10/2017
Large-scale evaluation on synthetic data

k-means: \( d = 10, k = 10 \)

- Size of sketch \( m \) independent of size of data \( n \)
  - Intuitively: dependent on complexity of the problem \( k, d \) ...
Alpha stable on synthetic data: toy example

Toy example

- CL-OMPR able to precisely estimate all parameters
  (10^{-2} precision in approx. 80 sec)

  (reported result for 1D approaches with MCMC: 10^{-1} precision in 1.5 hours)

Very heavy tailed...
Application on real data

• Efficient at large scales even on real data?
Application on real data

- Efficient at large scales even on real data?

Classic method for **speaker verification**
[Reynolds 2000] (for proof of concept) NIST
2005 database, MFCCs.

GMM (d=12, k=64, m=10000)

Results (EER, lower is better)
- EM on **300 000** MFCCs: **29.53**
- Sketch on **200 millions** MFCCs: **28.96**
  (120 000-fold compression)
Application on real data

- Efficient at large scales even on real data?

**Classic method for speaker verification**
[Reynolds 2000] (for proof of concept) NIST 2005 database, MFCCs.

**Spectral clustering for classification** [Uw 2001], augmented MNIST database [Loosli 2007].

- GMM (d=12, k=64, m=10000)

**Results (EER, lower is better)**
- EM on **300 000** MFCCs: **29.53**
- Sketch on **200 millions** MFCCs: **28.96**
  (120 000-fold compression)
How big a sketch?

**k-means**

Relative sketch size $m/(kd)$
How big a sketch?

Relative sketch size \( m/(kd) \)

k-means

GMM

Relative sketch size
How big a sketch?

Relative sketch size $m/(kd)$

Stable distributions

Relative sketch size
How big a sketch?

**Relative sketch size** $m/(kd)$

**Stable distributions**

**Sufficient sketch size**

$m \approx O(kd)$
How big a sketch?

**k-means**

Relative sketch size $m/(kd)$

**GMM**

Relative sketch size

Stable distributions

Relative sketch size

Sufficient sketch size

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

Can we characterize that?
Outline

1. Sketched Mixture Model Estimation
   1.1 A flexible greedy algorithm
   1.2 Experiments

2. Information-preservation guarantees
   2.1 Generic analysis
   2.2 Statistical Learning with sketches of limited size

3. Conclusion
Linear inverse problem

\[ \hat{z} = \hat{E} \Phi(X) \]
Linear inverse problem

Assumption on the data
- True distribution: $x_1, \ldots, x_n \overset{i.i.d.}{\sim} \pi^*$
Linear inverse problem

Assumption on the data
- True distribution: \( x_1, \ldots, x_n \sim i.i.d. \pi^* \)

Reformulation of the sketching
\[
\hat{z} = \hat{E}\Phi(X)
\]
Linear inverse problem

Assumption on the data
- True distribution: \( x_1, \ldots, x_n \overset{i.i.d.}{\sim} \pi^* \)

Reformulation of the sketching
- Linear operator:
  \[
  A_\pi = \mathbb{E}_{X \sim \pi} \Phi(X)
  \]
Linear inverse problem

Assumption on the data
- True distribution: \( x_1, \ldots, x_n \sim i.i.d. \pi^* \)

Reformulation of the sketching
- Linear operator:
  \[ A\pi = \mathbb{E}_{X \sim \pi} \Phi(X) \]
- « Noisy » linear measurement:
  \[ \hat{z} = A\pi^* + \hat{e} \]

Noise
\[ \hat{e} = \hat{\mathbb{E}}\Phi(X) - \mathbb{E}_{\pi^*} \Phi(X) \text{ small by Law of Large Numbers} \]
Linear inverse problem

Assumption on the data
- True distribution: \( x_1, \ldots, x_n \overset{i.i.d.}{\sim} \pi^* \)

Reformulation of the sketching
- Linear operator:
  \[ A\pi = \mathbb{E}_{X \sim \pi} \Phi(X) \]
- « Noisy » linear measurement:
  \[ \hat{z} = A\pi^* + \hat{e} \]

Noise
\[ \hat{e} = \hat{E}\Phi(X) - \mathbb{E}_{\pi^*} \Phi(X) \]
small by Law of Large Numbers

• Data = distribution

• Sketch = noisy linear measurement of the distribution (non-linear in data)
Linear inverse problem

- **Data** = distribution
- **Sketch** = noisy linear measurement of the distribution (non-linear in data)
- **Estimation problem** = linear inverse problem

**Assumption on the data**

- True distribution: \( x_1, \ldots, x_n \sim \pi^* \)

**Reformulation of the sketching**

- Linear operator:
  \[
  A\pi = \mathbb{E}_{X \sim \pi} \Phi(X)
  \]
- « Noisy » linear measurement:
  \[
  \hat{z} = A\pi^* + \hat{e}
  \]

Noise \( \hat{e} = \hat{\mathbb{E}}\Phi(X) - \mathbb{E}_{\pi^*} \Phi(X) \) small by Law of Large Numbers
Information preservation guarantees

\( P \)

\( \pi^* \)
Information preservation guarantees

$\mathcal{G}$ : Model set of « simple » distributions (eg. GMMs)
Information preservation guarantees

\( \mathcal{S} : \text{Model set of « simple » distributions (eg. GMMs)} \)
Information preservation guarantees

$\mathcal{G}$: Model set of « simple » distributions (eg. GMMs)
Information preservation guarantees

\[ \mathcal{G} : \text{Model set of « simple » distributions (eg. GMMs)} \]

**Goal**
Prove the existence of a **decoder** \( \Delta \) robust to **noise** and stable to **modeling error**.

« **Instance-optimal** » decoder
**Information preservation guarantees**

\[ \mathcal{G} : \text{Model set of « simple » distributions (eg. GMMs)} \]

**Goal**

Prove the existence of a **decoder** \( \Delta \) robust to **noise** and stable to **modeling error**.

« **Instance-optimal » decoder

**Lower Restricted Isometry Property**

\[ \| \sigma - \sigma' \| \lesssim \| A\sigma - A\sigma' \|_2 \]
**Information preservation guarantees**

\[ \delta : \text{Model set of « simple » distributions (eg. GMMs)} \]

\[ \Delta(\hat{z}) \in \arg \min_{\sigma \in \mathcal{G}} \|\hat{z} - A\sigma\|_2 \]

*Cost function used in practice (Part 1)*

**Goal**

Prove the existence of a *decoder* \( \Delta \) robust to *noise* and stable to *modeling error*.

« *Instance-optimal* » decoder

**Lower Restricted Isometry Property**

\[ \|\sigma - \sigma'\| \lesssim \|A\sigma - A\sigma'\|_2 \]
Information preservation guarantees

\[ \Delta(\hat{z}) \in \arg \min_{\sigma \in \mathcal{G}} \| \hat{z} - A\sigma \|_2 \]

Cost function used in practice (Part 1)!

**Goal**
Prove the existence of a decoder \( \Delta \) robust to noise and stable to modeling error.

« Instance-optimal » decoder

**Lower Restricted Isometry Property**
\[ \| \sigma - \sigma' \| \lesssim \| A\sigma - A\sigma' \|_2 \]

New goal: find/construct models \( \mathcal{G} \) and operators \( \mathcal{A} \) that satisfy the LRIP (w.h.p.)
Goal: LRIP \quad \text{w.h.p. on } A, \forall \sigma, \sigma' \in \mathcal{G}, \| \sigma - \sigma' \| \lesssim \| A\sigma - A\sigma' \|_2.
Goal: LRIP \ w.h.p. on $A$, $\forall \sigma, \sigma' \in \mathcal{G}$, $\|\sigma - \sigma\| \lesssim \|A\sigma - A\sigma'\|_2$.

1. **Pointwise LRIP**

   **Construction of $A$**:
   - Kernel mean [Gretton 2006, Borgwardt 2006]
   - Random features [Rahimi 2007]
Goal: LRIP \ w.h.p. on $A$, $\forall \sigma, \sigma' \in \mathcal{S}$, $||\sigma - \sigma'|| \leq ||A\sigma - A\sigma'||_2$. 

1. Pointwise LRIP

Construction of $A$:
- Kernel mean \cite{Gretton2006, Borgwardt2006}
- Random features \cite{Rahimi2007}

$\forall \sigma, \sigma'$, w.h.p. on $A$, LRIP.
Goal: LRIP \quad w.h.p. \text{ on } A, \forall \sigma, \sigma' \in \mathcal{G}, \|\sigma - \sigma'\| \lesssim \|A\sigma - A\sigma'\|_2.

1 \quad Pointwise LRIP

Construction of $A$:
\begin{itemize}
  \item Kernel mean [Gretton 2006, Borgwardt 2006]
  \item Random features [Rahimi 2007]
\end{itemize}

\forall \sigma, \sigma', \text{ w.h.p. on } A, \text{ LRIP}.

2 \quad Extension to LRIP

Covering numbers (compacity) of the normalized secant set $\mathcal{S}(\mathcal{G})$
Proving the LRIP

Goal: LRIP \quad \text{w.h.p. on } \mathcal{A}, \forall \sigma, \sigma' \in \mathcal{S}, \|\sigma - \sigma\| \lesssim \|A\sigma - A\sigma'\|_2.

1 \quad \textbf{Pointwise LRIP}

\textbf{Construction of } \mathcal{A} : \\
Kernel mean \ [Gretton 2006, Borgwardt 2006] \\
Random features \ [Rahimi 2007]

\forall \sigma, \sigma', \text{ w.h.p. on } \mathcal{A}, \text{ LRIP.}

2 \quad \textbf{Extension to LRIP}

\textbf{Covering numbers} (compacity) of the normalized secant set \ \mathcal{S}(\mathcal{S})

\textit{Subset of a unit ball (infinite dimension) that only depends on } \mathcal{S}
Proving the LRIP

Goal: LRIP  \text{w.h.p. on } \mathcal{A}, \forall \sigma, \sigma' \in \mathcal{G}, \|\sigma - \sigma'\| \lesssim \|A\sigma - A\sigma'\|_2.

1. Pointwise LRIP

Construction of \( \mathcal{A} \):
- Kernel mean \([\text{Gretton 2006, Borgwardt 2006}]\)
- Random features \([\text{Rahimi 2007}]\)

\(\forall \sigma, \sigma', \text{ w.h.p. on } \mathcal{A}, \text{ LRIP.}\)

2. Extension to LRIP

Covering numbers (compacity) of the normalized secant set \(\mathcal{S}(\mathcal{G})\)

Subset of a unit ball (infinite dimension) that only depends on \(\mathcal{G}\)

\(\text{w.h.p. on } \mathcal{A}, \forall \sigma, \sigma', \text{ LRIP.}\)
Main result

Main hypothesis

The normalized secant set $S(\mathcal{G})$ has finite covering numbers.

Result

For $m \geq C \times \log(\text{cov. num.})$,

- Quality of pointwise LRIP
- Dimensionality of the model

W.h.p.

$$\|\pi^* - \Delta(\hat{z})\| \leq d(\pi^*, \mathcal{G}) + \mathcal{O}(1/\sqrt{n})$$
Main result

Main hypothesis

The normalized secant set \( S(\mathcal{G}) \) has finite covering numbers.

Result

For \( m \geq C \times \log(\text{cov. num.}) \),

- Quality of pointwise LRIP
- Dimensionality of the model

W.h.p.

\[
\| \pi^* - \Delta(\hat{z}) \| \leq d(\pi^*, \mathcal{G}) + O\left(\frac{1}{\sqrt{n}}\right)
\]

- Classic CS: finite dimension: Known
- Here: infinite dimension: Technical
Main result

Main hypothesis

The normalized secant set $\mathcal{S}(\mathcal{G})$ has finite covering numbers.

Result

For $m \geq C \times \log(\text{cov. num.})$,

- Quality of pointwise LRIP
- Dimensionality of the model

W.h.p.

$$\|\pi^* - \Delta(\hat{z})\| \leq d(\pi^*, \mathcal{G}) + \mathcal{O}(1/\sqrt{n})$$

- Classic CS: finite dimension: Known
- Here: infinite dimension: Technical

Under simplified hypothesis:

$$m \approx n$$

(applied to mixture of stable dist.)
Outline

1. Sketched Mixture Model Estimation
   1.1 A flexible greedy algorithm
   1.2 Experiments

2. Information-preservation guarantees
   2.1 Main analysis and first results
   2.2 Statistical Learning with sketches of limited size

3. Conclusion
Key assumption for mixture models: separation of components
Compressive statistical learning

Key assumption for mixture models: separation of components

k-means with mixtures of Diracs
Key assumption for mixture models: separation of components

k-means with mixtures of Diracs

Hypotheses
- $\varepsilon$- separated centroids
- $\mathcal{M}$- bounded domain for centroids
Compressive statistical learning

Key assumption for **mixture models: separation of components**

**k-means with mixtures of Diracs**

**Hypotheses**

- $\mathcal{E}$ - separated centroids
- $\mathcal{M}$ - bounded domain for centroids

(no assumption on the data)
Key assumption for **mixture models: separation of components**

**k-means with mixtures of Diracs**

### Hypotheses
- $\mathcal{E}$ - separated centroids
- $\mathcal{M}$ - bounded domain for centroids

### Sketch
- Adjusted Fourier features *(for technical reasons)*

(no assumption on the data)
Key assumption for mixture models: separation of components

k-means with mixtures of Diracs

**Hypotheses**
- $\mathcal{E}$ - separated centroids
- $\mathcal{M}$ - bounded domain for centroids

**(no assumption on the data)**

**Sketch**
- Adjusted Fourier features (*for technical reasons*)

**Result**
- W.r.t. k-means usual cost (SSE)
Key assumption for **mixture models**: *separation of components*

**k-means with mixtures of Diracs**

**Hypotheses** *(no assumption on the data)*
- $\varepsilon$-separated centroids
- $\mathcal{M}$-bounded domain for centroids

**Sketch**
- *Adjusted* Fourier features *(for technical reasons)*

**Result**
- W.r.t. k-means usual cost (SSE)

**Sketch size**

$$m \geq \mathcal{O} \left( k^2 d^2 \text{polylog}(k, d) \log(\mathcal{M}/\varepsilon) \right)$$
Compressive statistical learning

Key assumption for **mixture models**: 
*separation of components*

### k-means with mixtures of Diracs

**Hypotheses**
- \( \mathcal{E} \)- separated centroids
- \( M \)- bounded domain for centroids

**Sketch**
- Adjusted Fourier features (*for technical reasons*)

**Result**
- W.r.t. k-means usual cost (SSE)

**Sketch size**

\[
m \geq \mathcal{O} \left( k^2 d^2 \text{polylog}(k, d) \log(M/\varepsilon) \right)
\]

### GMM with known covariance

(no assumption on the data)
Key assumption for **mixture models**: *separation of components*

**k-means with mixtures of Diracs**

**Hypotheses**
- $\varepsilon$- separated centroids
- $M$- bounded domain for centroids

**Sketch**
- *Adjusted* Fourier features *(for technical reasons)*

**Result**
- W.r.t. k-means usual cost (SSE)

**Sketch size**
\[
m \geq O\left(k^2d^2\text{polylog}(k,d)\log(M/\varepsilon)\right)
\]

**GMM with known covariance**

**Hypotheses**
- Sufficiently separated means
- Bounded domain for means
Compressive statistical learning

Key assumption for **mixture models**: *separation of components*

<table>
<thead>
<tr>
<th>k-means with mixtures of Diracs</th>
<th>GMM with known covariance</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Hypotheses</strong></td>
<td><strong>Hypotheses</strong></td>
</tr>
<tr>
<td>- $\varepsilon$- separated centroids</td>
<td>- Sufficiently separated means</td>
</tr>
<tr>
<td>- $M$- bounded domain for centroids</td>
<td>- Bounded domain for means</td>
</tr>
<tr>
<td><strong>Sketch</strong></td>
<td><strong>Sketch</strong></td>
</tr>
<tr>
<td>- Adjusted Fourier features <em>for technical reasons</em></td>
<td>- Fourier features</td>
</tr>
<tr>
<td><strong>Result</strong></td>
<td></td>
</tr>
<tr>
<td>- W.r.t. k-means usual cost (SSE)</td>
<td></td>
</tr>
<tr>
<td><strong>Sketch size</strong></td>
<td></td>
</tr>
<tr>
<td>$m \geq \mathcal{O} \left( k^2 d^2 \text{polylog}(k, d) \log(M/\varepsilon) \right)$</td>
<td></td>
</tr>
</tbody>
</table>
### Key assumption for mixture models: separation of components

#### k-means with mixtures of Diracs

**Hypotheses**
- $\mathcal{E}$- separated centroids
- $M$- bounded domain for centroids

**Sketch**
- Adjusted Fourier features *(for technical reasons)*

**Result**
- W.r.t. k-means usual cost (SSE)

**Sketch size**
$$m \geq O \left( k^2 d^2 \text{polylog}(k, d) \log(M/\varepsilon) \right)$$

#### GMM with known covariance

**Hypotheses**
- Sufficiently separated means
- Bounded domain for means

**Sketch**
- Fourier features

**Result**
- With respect to log-likelihood
Key assumption for **mixture models: separation of components**

**k-means with mixtures of Diracs**

**Hypotheses**
- $\varepsilon$-separated centroids
- $\mathcal{M}$-bounded domain for centroids

**Sketch**
- Adjusted Fourier features *(for technical reasons)*

**Result**
- W.r.t. k-means usual cost (SSE)

**Sketch size**
$$m \geq \mathcal{O}\left(k^2d^2\text{polylog}(k,d)\log(\mathcal{M}/\varepsilon)\right)$$

**GMM with known covariance**

**Hypotheses**
- Sufficiently separated means
- Bounded domain for means

**Sketch**
- Fourier features

**Result**
- With respect to log-likelihood

**Sketch size**
$$m \geq \mathcal{O}(k^2d^2\text{polylog}(k,d)\varphi(\text{sep.}))$$
GMM trade-off

Separation of means | Size of sketch
---|---
More High Freq.

<table>
<thead>
<tr>
<th>Separation of means</th>
<th>Number of measurements</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathcal{O}(\sqrt{d \log k})$</td>
<td>$m \geq \mathcal{O}(k^2d^2 \cdot \text{polylog}(k, d))$</td>
</tr>
<tr>
<td>$\mathcal{O}(\sqrt{d} + \log k)$</td>
<td>$m \geq \mathcal{O}(k^3d^2 \cdot \text{polylog}(k, d))$</td>
</tr>
<tr>
<td>$\mathcal{O}(\sqrt{\log k})$</td>
<td>$m \geq \mathcal{O}(k^2d^2e^d \cdot \text{polylog}(k, d))$</td>
</tr>
</tbody>
</table>
Outline

1. Sketched Mixture Model Estimation
   1.1 A flexible greedy algorithm
   1.2 Experiments

2. Information-preservation guarantees
   2.1 Main analysis and first results
   2.2 Statistical Learning with sketches of limited size

3. Conclusion
• Sketching method for **large-scale density estimation**
  • Well-adapted to **distributed** or **streaming** context
  • Focus on **mixture models**
Summary of contributions

- Practical illustration: **flexible greedy algorithm for any sketched mixture model estimation**
Summary of contributions

- Practical illustration: **flexible greedy algorithm for any sketched mixture model estimation**
  - GMM with diagonal covariance
  - k-means (mixture of Diracs)
  - *Mixture of multivariate elliptic stable distributions*
Summary of contributions

- Practical illustration: **flexible greedy algorithm for any sketched mixture model estimation**
  - GMM with diagonal covariance
  - k-means (mixture of Diracs)
  - *Mixture of multivariate elliptic stable distributions*
- Validation on real and synthetic data
Summary of contributions

- **Practical illustration:** flexible greedy algorithm for any sketched mixture model estimation
  - GMM with diagonal covariance
  - k-means (mixture of Diracs)
  - *Mixture of multivariate elliptic stable distributions*
- Validation on real and synthetic data

- **Information-preservation guarantees for sketched density estimation**
Summary of contributions

- Practical illustration: **flexible greedy algorithm for any sketched mixture model estimation**
  - GMM with diagonal covariance
  - k-means (mixture of Diracs)
  - *Mixture of multivariate elliptic stable distributions*
- Validation on real and synthetic data

- Information-preservation guarantees for **sketched density estimation**
  - Infinite dimensional *Compressive Sensing* (Restricted isometry property)
  - *Kernel methods* on distributions (Kernel mean, Random features)
Summary of contributions

- Practical illustration: **flexible greedy algorithm for any sketched mixture model estimation**
  - GMM with diagonal covariance
  - k-means (mixture of Diracs)
  - *Mixture of multivariate elliptic stable distributions*
- Validation on real and synthetic data

- Information-preservation guarantees for **sketched density estimation**
  - Infinite dimensional **Compressive Sensing** (Restricted isometry property)
  - **Kernel methods** on distributions (Kernel mean, Random features)
- Generic assumptions of **low-dimensionality** of the model set
Summary of contributions

- Practical illustration: **flexible greedy algorithm for any sketched mixture model estimation**
  - GMM with diagonal covariance
  - k-means (mixture of Diracs)
  - *Mixture of multivariate elliptic stable distributions*
- Validation on real and synthetic data

- Information-preservation guarantees for **sketched density estimation**
  - Infinite dimensional **Compressive Sensing** (Restricted isometry property)
  - **Kernel methods** on distributions (Kernel mean, Random features)
- Generic assumptions of **low-dimensionality** of the model set
- Focus on mixture models
  - Estimator of mixture of multivariate elliptic stable distributions
  - Statistical learning with controlled sketch size for k-means, sketched GMM with known covariance
• Obtain algorithmic guarantees?
• Obtain algorithmic guarantees?
  • Similar algorithms can be found in e.g. super-resolution with other interpretations (Frank-Wolfe, conditional gradient...) [eg Bredies 2012...]
  • Convergence guarantees as $k \to \infty$, no guarantees for exactly $k$-sparse measures...
Outlooks: sketch

• Obtain algorithmic guarantees?
  • Similar algorithms can be found in e.g. super-resolution with other interpretations (Frank-Wolfe, conditional gradient...) [eg Bredies 2012...]
  • Convergence guarantees as \( k \to \infty \), no guarantees for exactly \( k \)-sparse measures...

• Bridge observed gap between theory and practice?
Outlooks: sketch

• Obtain algorithmic guarantees?
  • Similar algorithms can be found in e.g. super-resolution with other interpretations (Frank-Wolfe, conditional gradient...) [eg Bredies 2012...]
  • Convergence guarantees as \( k \rightarrow \infty \), no guarantees for exactly \( k \)-sparse measures...

• Bridge observed gap between theory and practice?
  • Does not come from coverings numbers
  • Improve pointwise concentration?
Outlooks : sketch

• Obtain algorithmic guarantees?
  • Similar algorithms can be found in e.g. super-resolution with other interpretations (Frank-Wolfe, conditional gradient...) [eg Bredies 2012...]
  • Convergence guarantees as $k \to \infty$, no guarantees for exactly $k$-sparse measures...

• Bridge observed gap between theory and practice ?
  • Does not come from coverings numbers
  • Improve pointwise concentration?
  • Recent result: $k^2 d^2 \to k^3 d$
Outlooks : beyond sketches

- Combine with **dimension reduction** for HD data?
  - First map in low-d, then sketch
Outlooks : beyond sketches

- Combine with **dimension reduction** for HD data?
  - First map in low-d, then sketch

\[
\begin{array}{cccc}
  x_1 & x_2 & \cdots & x_n \\
\end{array}
\]
Outlooks: beyond sketches

- Combine with **dimension reduction** for HD data?
  - First map in low-d, then sketch

\[
\begin{array}{ccc}
  x_1 & x_2 & \ldots & x_n \\
\end{array}
\rightarrow
\begin{array}{ccc}
  x'_1 & x'_2 & \ldots & x'_n \\
\end{array}
\]

Eg. [Boutsidis 2010]
Outlooks: beyond sketches

- Combine with **dimension reduction** for HD data?
  - First map in low-d, then sketch

\[
\begin{array}{cccc}
  x_1 & x_2 & \cdots & x_n \\
\end{array}
\quad \xrightarrow{\text{Eg. [Boutsidis 2010]}} \quad
\begin{array}{ccc}
  x'_1 & x'_2 & \cdots & x'_n \\
\end{array}
\quad \xrightarrow{\text{Our guarantees}} \quad Z
\]
Outlooks: beyond sketches

• Combine with **dimension reduction** for HD data?
  • First map in low-d, then sketch

\[ \begin{bmatrix} x_1 & x_2 & \ldots & x_n \end{bmatrix} \xrightarrow{\text{Eg. [Boutsidis 2010]}} \begin{bmatrix} x'_1 & x'_2 & \ldots & x'_n \end{bmatrix} \]

• Extend framework to other tasks?
  • « Sketchify » other kernel methods?

\[ K(x, y) \approx \mathbf{z}(\mathbf{x})^T \mathbf{z}(\mathbf{y}) \]

*Oliva 2016*
Outlooks: beyond sketches

- Extension to multi-layer sketches? (Neural networks...)

\[ X \rightarrow \text{Multiplication by frequencies (aka weights)} \rightarrow W^T X \rightarrow \rho(W^T X) \rightarrow \text{Average (aka pooling)} \]

- Complex exponential (aka pointwise non-linearity)
Outlooks: beyond sketches

- Extension to multi-layer sketches? (Neural networks...)
  - Equivalence between LRIP and instance optimality still valid for **non-linear operators**!

\[ \mathbf{X} \]

- Multiplication by frequencies (**aka weights**)

\[ \mathbf{W}^\top \mathbf{X} \]

- Complex exponential (**aka pointwise non-linearity**)

\[ \rho (\mathbf{W}^\top \mathbf{X}) \]

- Average (**aka pooling**)

\[ \hat{\mathbf{z}} \]
Thank you!