Compressive learning (e.g. clustering) from a (quantized) sketch of the dataset

Vincent Schellekens
Context: machine learning

A machine learning classic: hand-written digit recognition

Veeeeeeery difficult to program explicitly!

“This is a two”

MNIST dataset
Context: machine learning

Solution: let the computer figure it out by itself!
Context: machine learning

Solution: let the computer figure it out by itself!

Requires a lot of training data (examples)
Solution: let the computer figure it out by itself!

Requires *a lot* of training data (examples)
Machine learning limitations :-(

Large datasets means:
• Large memory required
• Slow learning algorithm
Machine learning limitations :-(

Large datasets means:
• Large memory required
• Slow learning algorithm

BUT extracted “knowledge” is “simple”
-> do we really need all this data?
Machine learning limitations :-(

Large datasets means:
  • Large memory required
  • Slow learning algorithm

BUT extracted “knowledge” is “simple”
-> do we really need all this data?

NO!
(otherwise this talk would be finished)
Compressive learning
(from a sketch)

Dataset → Learning algorithm → Model

- Compressed representation
- Preserves relevant information

Sketch

[Gribonval17]
Compressing a dataset?

\[ X = \{x_1, x_2, \ldots, x_N\} \]

\( N \) examples

Dataset

\( x_i \in \mathbb{R}^n \)

\( n \)-dimensional
Compressing a dataset?

$X = \{x_1, x_2, \ldots, x_N\} \subset \mathbb{R}^n$ to $\{y_1, y_2, \ldots, y_N\} \subset \mathbb{R}^p$

- Compressed representation
- Preserves relevant information
Compressing a dataset?

Dataset  $X = \{x_1, x_2, \ldots, x_N\}$  

Compressed representation  

Dimensionality reduction  $\mathbb{R}^n \rightarrow \mathbb{R}^p$  

$y_i \in \mathbb{R}^p$  

- Compressed representation ✓
- Preserves relevant information ✓
- Constant number of examples ✗

$N$ can be VERY large ("big data")!
Compressing a dataset!

$X = \begin{bmatrix} x_1, \ldots, x_N \end{bmatrix}$

$z_X = z \in \mathbb{C}^m$

- Compressed representation
- Preserves relevant information
- Dataset summary = single vector

$z_X = \mathcal{S}(X)$

$\mathcal{S}$ is a sketching operator, as defined in Gribonval (2017).
\[ X = \{ x_1, x_2, \ldots, x_n \} \]

Sketching

\[ z_X = \left[ \frac{1}{N} \sum_{x_i \in X} e^{-i \omega_j^T x_i} \right]_{j=1}^m \in \mathbb{C}^m \]
Sketch of a dataset

\[ X = \ldots \]

Sketching

\[ z_X = \left[ \frac{1}{N} \sum_{x_i \in X} e^{-i \omega_j^T x_i} \right]_{j=1}^m \in \mathbb{C}^m \]

1. Project on (random) vectors

\[ \omega_j \sim \Lambda \] (cfr. later)
Sketch of a dataset

\[ X = \begin{bmatrix} \vdots & \vdots & \cdots & \vdots \end{bmatrix} \]

1. Project on (random) vectors
2. Nonlinear periodic signature function

\[ z_X = \left[ \frac{1}{N} \sum_{\mathbf{x}_i \in X} e^{-i\omega_j^T \mathbf{x}_i} \right]_{j=1}^m \in \mathbb{C}^m \]
Sketch of a dataset

\[ X = \begin{bmatrix} \vdots & \vdots & \cdots & \vdots \end{bmatrix} \]

**Sketching**

\[ z_X = \left[ \frac{1}{N} \sum_{x_i \in X} e^{-i\omega_j^T x_i} \right]_{j=1}^{m} \in \mathbb{C}^m \]

1. Project on (random) vectors
2. Nonlinear periodic signature function
3. Pooling (average)
Sketch of a distribution

Sketching: an operator on probability distributions!

\[ A(\mathcal{P}) := \mathbb{E}_{x \sim \mathcal{P}} \left[ e^{-i \omega_j^T x} \right]_{j=1}^m \]

Input: probability distribution

Output: \( m \) moments of it
Sketch of a distribution

Sketching: an operator on probability distributions!

\[ A(\mathcal{P}) := \mathbb{E}_{x \sim \mathcal{P}} \left[ e^{-i \omega_j^T x} \right]_{j=1}^m \]

Particular case: dataset <-> empirical distribution

\[ \hat{z}_X = A(\hat{\mathcal{P}}_X) \]

\[ \hat{\mathcal{P}}_X = \frac{1}{N} \sum_{x_i \in X} \delta_{x_i} \]
Sketch interpretation (1)

Sketch of \( P = \text{Random Fourier sampling of } P \)

Characteristic function: characterises \( P \)

\[ \phi_P(\omega) := \mathbb{E}_{x \sim P} e^{-i \omega^T x} \]

\[ A(P) := \mathbb{E}_{x \sim P} \left[ e^{-i \omega_j^T x} \right]_{j=1}^m \]
Sketch of $\mathbf{P} = \text{Random Fourier sampling of } \mathbf{P}$

$\mathbf{A}(\mathbf{P})_j = \phi_{\mathbf{P}}(\omega_j)$

**Characteristic function: characterises \( \mathbf{P} \)**

$\phi_{\mathbf{P}}(\omega) := \mathbb{E}_{\mathbf{x} \sim \mathbf{P}} e^{-i\omega^T \mathbf{x}}$

$\mathbf{A}(\mathbf{P}) := \mathbb{E}_{\mathbf{x} \sim \mathbf{P}} \left[ e^{-i\omega_j^T \mathbf{x}} \right]_{j=1}^m$
Sketch interpretation (1)

Sketch of $P =$ Random Fourier sampling of $P$

$$A(P)_j = \phi_P(\omega_j)$$

Typically:

$$A(P) := \mathbb{E}_{x \sim P} \left[ e^{-i \omega^T x} \right]_{j=1}^m$$
Sketch interpretation (1)

Sketch of $P = \text{Random Fourier sampling of } P$

$$A(P)_j = \phi_P(\omega_j)$$

Typically:

In practice:
- Application-dependent
- Requires some data

See also: Distilled sensing
Sketch interpretation (2)

Sketch of $P = \text{view } P$ through kernel $K$: “Similarity measure”

$A(P) := \mathbb{E}_{x \sim P} \left[ e^{-i \omega \mathcal{F} x} \right]_{j=1}^m$

[Rahimi08]
Sketch interpretation (2)

Sketch of $P = \text{view } P$ through kernel $K$ : “Similarity measure”

$$\Lambda(\omega)$$

Fourier domain $\|\omega\|$

Bochner's Thm.

$$\mathcal{F}$$

$$K(x)$$

Direct domain $\|x\|$
Sketch interpretation (3)

Sketch of $P = \text{low-dimensional embedding of } P$

$\text{pdf space}$

$\dim = \infty$

$P$

\[ A(P) := \mathbb{E}_{x \sim P} \left[ e^{-i \omega^T x} \right]_{j=1}^m \]
Sketch interpretation (3)

Sketch of $P = \text{low-dimensional embedding of } P$

PDF space
$\text{dim} = \infty$

$A(P) := \mathbb{E}_{x \sim P} \left[ e^{-i \omega_j^T x} \right]_{j=1}^m$
Sketch interpretation (3)

Sketch of $P = \text{low-dimensional embedding of } P$

pdf space
\[ \dim = \infty \]

$P \quad Q$

Preserved distance?

$A(P) := \mathbb{E}_{x \sim \mathcal{P}} \left[ e^{-i \omega_j x} \right]_{j=1}^{m}$
Sketch of $P = \text{low-dimensional embedding of } P$

pdf space
$\text{dim} = \infty$

Hilbert space defined by $K$

Preserved distance?

$\mathcal{A}(P) := \mathbb{E}_{x \sim P} \left[ e^{-i \omega^T x} \right]_{j=1}^m$
Sketch interpretation (3)

Sketch of $P$ = low-dimensional embedding of $P$

Infinite-dimensional Compressed Sensing!

$\mathcal{A}(P) := \mathbb{E}_{x \sim P} \left[ e^{-i\omega_j^T x} \right]_{j=1}^m$

[Hilbert space defined by $K$]

$\mathcal{C}^m$

Preserved distance?

$A{P}$

$A{Q}$
Infinite-dimensional Compressed Sensing!

Sketch interpretation (3)

Sketch of $P = \text{low-dimensional embedding of } P$

Underlying assumption:

$P \in (\text{insert sparse set here})$

Infinite-dimensional Compressed Sensing!

$A(P) := \mathbb{E}_{x \sim P} \left[ e^{-i \omega_j^T x} \right]_{j=1}^{m}$
Sketch interpretation (3)

Sketch of $P = \text{low-dimensional embedding of } P$

pdf space
$\dim = \infty$

Underlying assumption:
$P \in (\text{insert sparse set here})$

Defined by the application

Infinite-dimensional Compressed Sensing!

$A(P) := \mathbb{E}_{x \sim P} \left[ e^{-i\omega_j x} \right]_{j=1}^m$
Compressive clustering

Usually (K-means,…)

Iterative algorithm

Goal: centroids

\[ X = \{ \mathbf{x}_i \in \mathbb{R}^n \}_{i=1}^N \]

\[ C = \{ \mathbf{c}_k \}_{k=1}^K \]
Compressive clustering

Usually (K-means,...)

Iterative algorithm

Goal: centroids

$C = \{c_k\}_{k=1}^K$

$X = \{x_i \in \mathbb{R}^n\}_{i=1}^N$

$Z_X = \begin{bmatrix} \end{bmatrix} \in \mathbb{C}^m$

[Keriven16-CKM]
Compressive clustering

\[ z_X = \in \mathbb{C}^m \]

\[ C = \{c_k\}_{k=1}^{K} \]
Compressive clustering

\[ z_X = \mathbf{c} \in \mathbb{C}^m \]

Sketch matching! (cfr. CS)

\[
\min_{C, \alpha} \left\| z_X - A\left(\sum_{k=1}^{K} \alpha_k \delta_{c_k}\right)\right\|_2^2
\]

Sparse model: mixture of K diracs
Compressive clustering

Sketch matching! (cfr. CS)

$$\min_{C, \alpha} \| z_X - A(\sum_{k=1}^{K} \alpha_k \delta_{c_k}) \|_2^2$$
Compressive clustering

Sketch matching! (cfr. CS)

$$\min_{C, \alpha} \| z_X - A(\sum_{k=1}^{K} \alpha_k \delta_{c_k}) \|_2^2$$
Compressive clustering

Sketch matching! (cfr. CS)

\[
\min_{C, \alpha} \left\| z_X - A \left( \sum_{k=1}^{K} \alpha_k \delta_{c_k} \right) \right\|_2^2
\]

Highly non-convex problem!
=> CS-based heuristics

Compared by the sketch distance
The power of the sketch

Number of “measurements” \( m \) needed?

\[
m = \mathcal{O}(nK)
\]

\( \sim \) information rate
The power of the sketch

Number of “measurements” $m$ needed?

$m = \mathcal{O}(nK)$

No dependence on $N$!

Same learning time!
More data = better estimation of pdf

~ information rate
The power of the sketch

Number of “measurements” \( m \) needed?

\[ m = \mathcal{O}(nK) \]

\( m \) is independent of \( N \)!

Same learning time!

More data = better estimation of pdf

+ easy update/parallel computing of \( z_X \)
BUT…

Signal acquisition

Dataset

Sketch computation

Sketch
BUT…

Why not do:

Direct sketch acquisition
Quantized sketch (my work)

\[
z_X = \left[ \frac{1}{N} \sum_{x_i \in X} e^{-10 \Omega_j^T x_i} \right]_{j=1}^m
\]

No easy hardware implementation
Quantized sketch (my work)

\[ z_{Q,X} = \frac{1}{N} \sum_{x_i \in X} Q_\Delta (\omega^T_j x_i + \xi_j) \]

LSB of quantizer => hardware friendly

Validated on clustering
\[ m = O(nK) \text{ increase but ok!} \]
Quantized sketch (my work)

\[ z_{Q,X} = \frac{1}{N} \sum_{x_i \in X} Q_\Delta (\omega_j^T x_i + \xi_j) \]

LSB of quantizer => hardware friendly

Validated on clustering

\( m = O(nK) \) increase but ok!

New sketch operator!

Discontinuous objective => gradient KO?
What does it mean?

Sketch interpretation is (only a little bit) modified
What will I do next?

Some things I look forward to do:
• Other tasks than clustering
• Other sketch functions
• Theoretical guarantees
• Algorithmic guarantees (local convexity?)
• New applications (e.g. in HS imaging?)
• …
Thank you for your attention! Questions?
References


