Rademacher Embedding
with application to Earth-Mover Distance

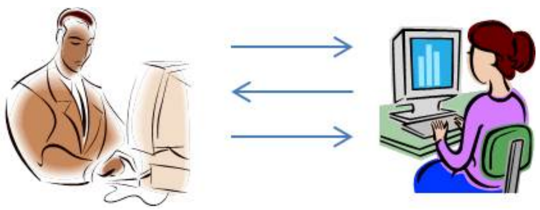
Elad Verbin and Qin Zhang

MADALGO and CTIC, Aarhus University

Streaming'12,
TU Dortmund University

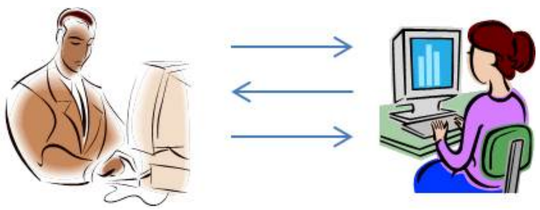
Communication, sketches and embeddings

- Communication problems
  - Alice has input x. Bob has input y. $x, y \in \{0,1\}^n$
  - Wish to compute $f(x,y)$, e.g., $\|x-y\|_{\mathbb{X}}$
  - Minimize $\#$ bits communicated

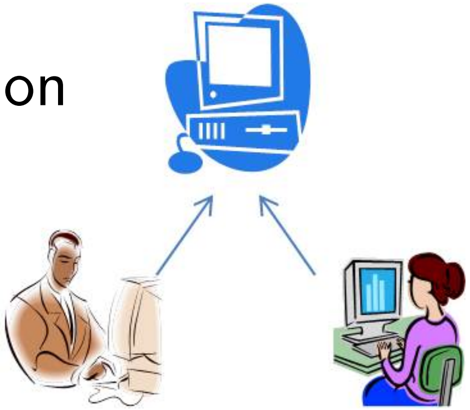


Communication, sketches and embeddings

- Communication problems
  - Alice has input x. Bob has input y. $x, y \in \{0,1\}^n$
  - Wish to compute $f(x,y)$, e.g., $\|x - y\|_{\mathbb{X}}$
  - Minimize # bits communicated

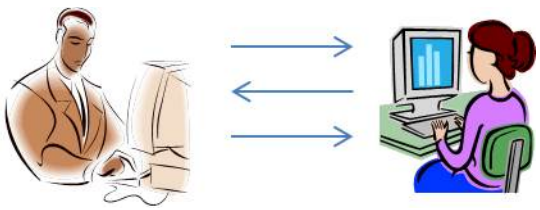


- Linear sketches
  - $sk(x,y) = \text{combine}(sk(x), sk(y))$; $\text{Decode}(sk(x,y)) \approx f(x,y)$
  - Allow multiplicative approx
  - Useful for data streams, distributed computation

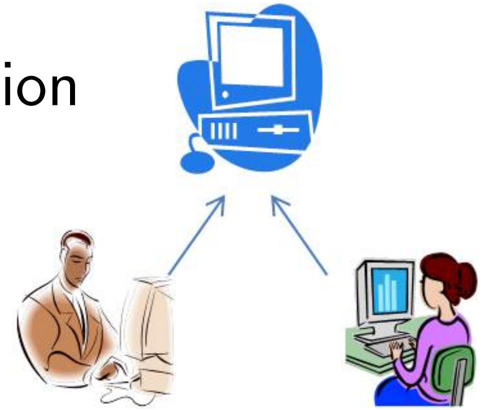


Communication, sketches and embeddings

- Communication problems
  - Alice has input x. Bob has input y. $x, y \in \{0,1\}^n$
  - Wish to compute $f(x,y)$, e.g., $\|x-y\|_{\mathbb{X}}$
  - Minimize # bits communicated



- Linear sketches
  - $sk(x,y) = \text{combine}(sk(x), sk(y))$; $\text{Decode}(sk(x,y)) \approx f(x,y)$
  - Allow multiplicative approx
  - Useful for data streams, distributed computation



- (Linear) embeddings ($\|\cdot\|_{\mathbb{X}} \to \|\cdot\|_{\mathbb{Y}}$)
  - A special sketch. $\|\text{embed}(x)\|_{\mathbb{Y}} \approx \|x\|_{\mathbb{X}}$
  - Simple and intuitive. Have additional advantages

Warm-up

A Classic Problem: Embedding ℓ_1^n to ℓ_1^d $(d \ll n)$

Goal: $O(1)$ distortion

Embedding ℓ_1^n to ℓ_1^d

- First try: sample d random coordinates.
  - Works for $(1, 1, \ldots, 1)$.
  - Fails for $(1, 0, \ldots, 0)$.

Embedding ℓ_1^n to ℓ_1^d

- First try: sample d random coordinates.
  - Works for $(1, 1, \ldots, 1)$.
  - Fails for $(1, 0, \ldots, 0)$.
- Second try: pick subsets of coordinates.
  - Some big subsets, some small subsets.

Sub-sampling

- Input: $(1, 1, \ldots, 1)$
  - Pick each coord w.pr. $p = 1/n$
  - On avg one non-zero coord picked
  - Sum up sampled coords and scale the sum back by $1/p$

Sub-sampling

- Input: $(1, 1, \ldots, 1)$
  - Pick each coord w.pr. $p = 1/n$
  - On avg one non-zero coord picked
  - Sum up sampled coords and scale the sum back by $1/p$
- Input: $(1, 0, \ldots, 0)$
  - Pick each coord w.pr. $p = 1$
  - Exactly one non-zero coord picked

Sub-sampling

- Input: $(1, 1, \ldots, 1)$
  - Pick each coord w.pr. $p = 1/n$
  - On avg one non-zero coord picked
  - Sum up sampled coords and scale the sum back by $1/p$
- Input: $(1, 0, \ldots, 0)$
  - Pick each coord w.pr. $p = 1$
  - Exactly one non-zero coord picked
- Input: $(1, 1/2, 1/2, 1/4, 1/4, 1/4, 1/4, \ldots)$
  - Sample on $\log n$ levels, on level i we sample w. pr. $p_i = 1/2^i$.
  - Each class will get roughly one coord sampled at a certain level
  - On each level i, sum up all sampled items and scale by $1/p_i$ to create a coord in host space

Sub-sampling

- Input: $(1, 1, \ldots, 1)$
  - Pick each coord w.pr. $p = 1/n$
  - On avg one non-zero coord picked
  - Sum up sampled coords and scale the sum back by $1/p$
- Input: $(1, 0, \ldots, 0)$
  - Pick each coord w.pr. $p = 1$
  - Exactly one non-zero coord picked
- class 1 class 2 • • •

  Input: $(1, 1/2, 1/2, 1/4, 1/4, 1/4, 1/4, \ldots)$
  - Sample on $\log n$ levels, on level i we sample w. pr. $p_i = 1/2^i$.
  - Each class will get roughly one coord sampled at a certain level
  - On each level i, sum up all sampled items and scale by $1/p_i$ to create a coord in host space

Sub-sampling

- Input: $(1, 1, \ldots, 1)$
  - Pick each coord w.pr. $p = 1/n$
  - On avg one non-zero coord picked
  - Sum up sampled coords and scale the sum back by $1/p$
- Input: $(1, 0, \ldots, 0)$
  - Pick each coord w.pr. $p = 1$
  - Exactly one non-zero coord picked
- class 1 class 2 • • •

  Input: $(1, 1/2, 1/2, 1/4, 1/4, 1/4, 1/4, \ldots)$
  - Sample on $\log n$ levels, on level i we sample w. pr. $p_i = 1/2^i$.
  - Each class will get roughly one coord sampled at a certain level
  - On each level i, sum up all sampled items and scale by $1/p_i$ to create a coord in host space

For **general** vectors. Class i are all coords $|x| \in [1/2^{i-1}, 1/2^i)$.

A problem

- Something is wrong . . .
  - $(1, 1, \ldots, 1) \rightarrow (n, n, \ldots, n)$
  - $(1, \ldots, 1, -1, \ldots, -1) \rightarrow (0, 0, \ldots, 0, n)$

    (Note: Everything I show here is just for illustration purpose)

A problem

- Something is wrong ...
  - $(1, 1, \ldots, 1) \rightarrow (n, n, \ldots, n)$
  - $(1, \ldots, 1, -1, \ldots, -1) \rightarrow (0, 0, \ldots, 0, n)$

    (Note: Everything I show here is just for illustration purpose)
- $d = \log n$, approximation factor: $\log n$

A problem

- Something is wrong . . .
  - $(1, 1, \ldots, 1) \to (n, n, \ldots, n)$
  - $(1, \ldots, 1, -1, \ldots, -1) \to (0, 0, \ldots, 0,\ n)$

    (Note: Everything I show here is just for illustration purpose)
- $d = \log n$, approximation factor: $\log n$
- Another angle to see the problem:

  $(1, 1/2, 1/2, 1/4, 1/4, 1/4, 1/4, \ldots) \to (\log n, \ldots, \log n)$
  Many $1/2^i$'s could be sampled on levels $j \ll i$.

  But in fact we only want to count items from class j on level j to avoid double counts. In other words we want to isolate class j at level j.

A problem

- Something is wrong . . .
  - $(1, 1, \ldots, 1) \rightarrow (n, n, \ldots, n)$
  - $(1, \ldots, 1, -1, \ldots, -1) \rightarrow (0, 0, \ldots, 0, \ n)$

    (Note: Everything I show here is just for illustration purpose)

- $d = \log n$, approximation factor: $\log n$

- Another angle to see the problem:

  $(1, 1/2, 1/2, 1/4, 1/4, 1/4, 1/4, \ldots) \rightarrow (\log n, \ldots, \log n)$
  Many $1/2^i$'s could be sampled on levels $j \ll i$.

  But in fact we only want to count items from class j on level j to avoid double counts. In other words we want to isolate class j at level j.

- An easy fix: create cancellations.

New idea: Cancellations

- Cancellations Idea: multiply random ± 1 before summing.

  At level i ($i = 0, 1, \ldots, \log n$), each coord is picked w.p. $p_i = 1/2^i$, then multiplied random ± 1, then sum all.

New idea: Cancellations

- Cancellations Idea: multiply random ± 1 before summing.

  At level i ($i = 0, 1, \ldots, \log n$), each coord is picked w.p. $p_i = 1/2^i$, then multiplied random ± 1, then sum all.

- Now:
  - $(1, 1, \ldots, 1) \rightarrow (0, 0, \ldots, 0, \; n)$

  Isolated at $p = 1/n$, rest vanishes. And

New idea: Cancellations

- Cancellations Idea: multiply random ± 1 before summing.

  At level i ($i = 0, 1, \ldots, \log n$), each coord is picked w.p. $p_i = 1/2^i$, then multiplied random ± 1, then sum all.

- Now:

  - $(1, 1, \ldots, 1) \rightarrow (0, 0, \ldots, 0,\ n)$

  Isolated at $p = 1/n$, rest vanishes. And

  - $(1, 1/2, 1/2, 1/4, 1/4, 1/4, 1/4, \ldots) \rightarrow (1, \ldots, 1, 1)$

  Many $1/2^i$'s that sampled on levels $j \ll i$ cancel with each other.

New idea: Cancellations

- Cancellations Idea: multiply random ± 1 before summing.

  At level i ($i = 0, 1, \ldots, \log n$), each coord is picked w.p. $p_i = 1/2^i$, then multiplied random ± 1, then sum all.

- Now:
  - $(1, 1, \ldots, 1) \rightarrow (0, 0, \ldots, 0, \ n)$

  Isolated at $p = 1/n$, rest vanishes. And

  - $(1, 1/2, 1/2, 1/4, 1/4, 1/4, 1/4, \ldots) \rightarrow (1, \ldots, 1, 1)$

  Many $1/2^i$'s that sampled on levels $j \ll i$ cancel with each other.

- We call it Rademacher embedding

Rademacher embedding ℓ_1^n to ℓ_1^d

- Properties of Rademacher embedding for ℓ_1^n to ℓ_1^d
  - $\Pr[\|\text{embed}(x)\| \leq 0.1\,\|x\|] \leq o(1)$
  - $\Pr[\|\text{embed}(x)\| \geq 10\,\|x\|] \leq 0.1$
  - $d = \mathit{poly}\log(n)$
  - Linear and efficient
  - Randomized (distribution over embeddings)
  - Weak (failure prob. on each vector separately)

Generalization

Generalization

- In ℓ_1^n, each element has the form $x = (x_1, \ldots, x_n)$ where $x_i \in \mathbb{R}$. What if $x_i \in \mathbb{X}$ instead?

Generalization

- In ℓ_1^n, each element has the form $x = (x_1, \ldots, x_n)$ where $x_i \in \mathbb{R}$. What if $x_i \in \mathbb{X}$ instead?
- $\|(x_1, \ldots, x_n)\|_{\ell_1^n \otimes \mathbb{X}} = \sum_i \|x_i\|_{\mathbb{X}} \quad (\|x\|_{1,\mathbb{X}}$ for short $)$.

  Goal: Embed $\ell_1^n \otimes \mathbb{X}$ into $\ell_1^d \otimes \mathbb{X}$ for some $d \ll n$

Generalization

- In ℓ_1^n, each element has the form $x = (x_1, \ldots, x_n)$ where $x_i \in \mathbb{R}$. What if $x_i \in \mathbb{X}$ instead?
- $\|(x_1, \ldots, x_n)\|_{\ell_1^n \otimes \mathbb{X}} = \sum_i \|x_i\|_{\mathbb{X}} \quad (\|x\|_{1,\mathbb{X}}$ for short).

  Goal: Embed $\ell_1^n \otimes \mathbb{X}$ into $\ell_1^d \otimes \mathbb{X}$ for some $d \ll n$
- Rademacher embedding works.

Generalization

- In ℓ_1^n, each element has the form $x = (x_1, \ldots, x_n)$ where $x_i \in \mathbb{R}$. What if $x_i \in \mathbb{X}$ instead?
- $\|(x_1, \ldots, x_n)\|_{\ell_1^n \otimes \mathbb{X}} = \sum_i \|x_i\|_{\mathbb{X}}$ $\quad$ ($\|x\|_{1,\mathbb{X}}$ for short).

  Goal: Embed $\ell_1^n \otimes \mathbb{X}$ into $\ell_1^d \otimes \mathbb{X}$ for some $d \ll n$
- Rademacher embedding works.
- Where did we use the nature of $\mathbb{R}$?

  – When we analyzed cancellations.

Generalization

- In ℓ_1^n, each element has the form $x = (x_1, \ldots, x_n)$ where $x_i \in \mathbb{R}$. What if $x_i \in \mathbb{X}$ instead?

- $\|(x_1, \ldots, x_n)\|_{\ell_1^n \otimes \mathbb{X}} = \sum_i \|x_i\|_{\mathbb{X}} \quad (\|x\|_{1,\mathbb{X}}$ for short $)$.

  Goal: Embed $\ell_1^n \otimes \mathbb{X}$ into $\ell_1^d \otimes \mathbb{X}$ for some $d \ll n$

- Rademacher embedding works.

- Where did we use the nature of $\mathbb{R}$?
  – When we analyzed cancellations.

- **Take-home message**:
  The better $\mathbb{X}$ supports cancellations,
  The better Rademacher embedding will work.

Rademacher dimension

A normed space $\mathbb{X}$ has *Rademacher dimension* α if for any natural number s, and for any $x_1, x_2, \ldots, x_s \in \mathbb{X}$ with $\|x_i\|_{\mathbb{X}} \leq T$, we have with pr. at least $1 - 1/\alpha^{\Omega(1)}$ that

$$\left\|\sum_{i\in[s]} \varepsilon_i x_i\right\|_{\mathbb{X}} \leq \alpha \cdot \sqrt{s} \cdot T.$$

Here, $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_s$ are (± 1)-valued random variables such that $\Pr[\varepsilon_i = +1] = \Pr[\varepsilon_i = -1] = 1/2$ for all $i \in [s]$ (Rademacher distribution).

Rademacher dimension

A normed space $\mathbb{X}$ has *Rademacher dimension* α if for any natural number s, and for any $x_1, x_2, \ldots, x_s \in \mathbb{X}$ with $\|x_i\|_{\mathbb{X}} \leq T$, we have with pr. at least $1 - 1/\alpha^{\Omega(1)}$ that

$$\left\|\sum_{i \in [s]} \varepsilon_i x_i\right\|_{\mathbb{X}} \leq \alpha \cdot \sqrt{s} \cdot T.$$

Here, $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_s$ are (± 1)-valued random variables such that $\Pr[\varepsilon_i = +1] = \Pr[\varepsilon_i = -1] = 1/2$ for all $i \in [s]$ (Rademacher distribution).

$\ell_1^1 = \mathbb{R}$ has Rademacher dimension $\alpha = O(1)$

Rademacher embedding: $\ell_1^n \otimes \mathbb{X} \to \ell_1^d \otimes \mathbb{X}$

Main Theorem

Let $\mathbb{X}$ be a normed space with Rademacher dimension α. Let $\lambda = \max\{\alpha, \log^3 n\}$. Then there exists a distribution over linear embeddings $\mu : \ell_1^n \otimes \mathbb{X} \to \ell_1^{\lambda^{O(1)}} \otimes \mathbb{X}$, such that

- $\|\mu(x)\|_{1,\mathbb{X}} \geq \Omega(\|x\|_{1,\mathbb{X}})$ with pr. $1 - 1/\lambda^{\Omega(1)}$.
- $\|\mu(x)\|_{1,\mathbb{X}} \leq O(\|x\|_{1,\mathbb{X}})$ with pr. 0.99.

The actual embedding algorithm

We sample $x_1, \ldots, x_n$ at $\ell = \lceil \log_\lambda(4\lambda n) \rceil$ levels ($\lambda = \max\{\alpha, \log^3 n\}$).
At level $k \in [\ell]$, let $p_k = \lambda^{-k}$.
For each level k we maintain a hash table H_k of $\lambda^{O(1)}$ cells,
with hash function $h_k : [n] \to [\lambda^{O(1)}]$.

The embedding algorithm (for each sample level $k \in [\ell]$)

1. Subsample a set $I_k \subseteq [n]$

   where each $x_i \in [n]$ is picked with pr. p_k.

2. For each cell $v \in [t]$ in the hash table H_k, compute

$$Z_k^v = \sum_{i \in [n]} \chi[i \in I_k] \cdot \chi[h_k(i) = v] \cdot \epsilon_i \cdot x_i \cdot 1/p_k,$$

   where $\Pr[\epsilon_i = +1] = \Pr[\epsilon_i = -1] = 1/2$ for all $i \in [n]$.

   The summations/multiplications are coordinate-wise.

At the end, $\mu(x)$ consists of $\lambda^{O(1)} \cdot l = \lambda^{O(1)}$ cells/coordinates.

The actual embedding algorithm

We sample $x_1, \ldots, x_n$ at $\ell = \lceil \log_\lambda (4\lambda n) \rceil$ levels ($\lambda = \max\{\alpha, \log^3 n\}$).
At level $k \in [\ell]$, let $p_k = \lambda^{-k}$.
For each level k we maintain a hash table H_k of $\lambda^{O(1)}$ cells,
with hash function $h_k : [n] \to [\lambda^{O(1)}]$.

To prevent coords from the same class in their own "isolated" levels from cancelling with each other.

The embedding algorithm (for each sample level)

1. Subsample a set $I_k \subseteq [n]$
   where each $x_i \in [n]$ is picked with pr. p_k.
2. For each cell $v \in [t]$ in the hash table H_k, compute

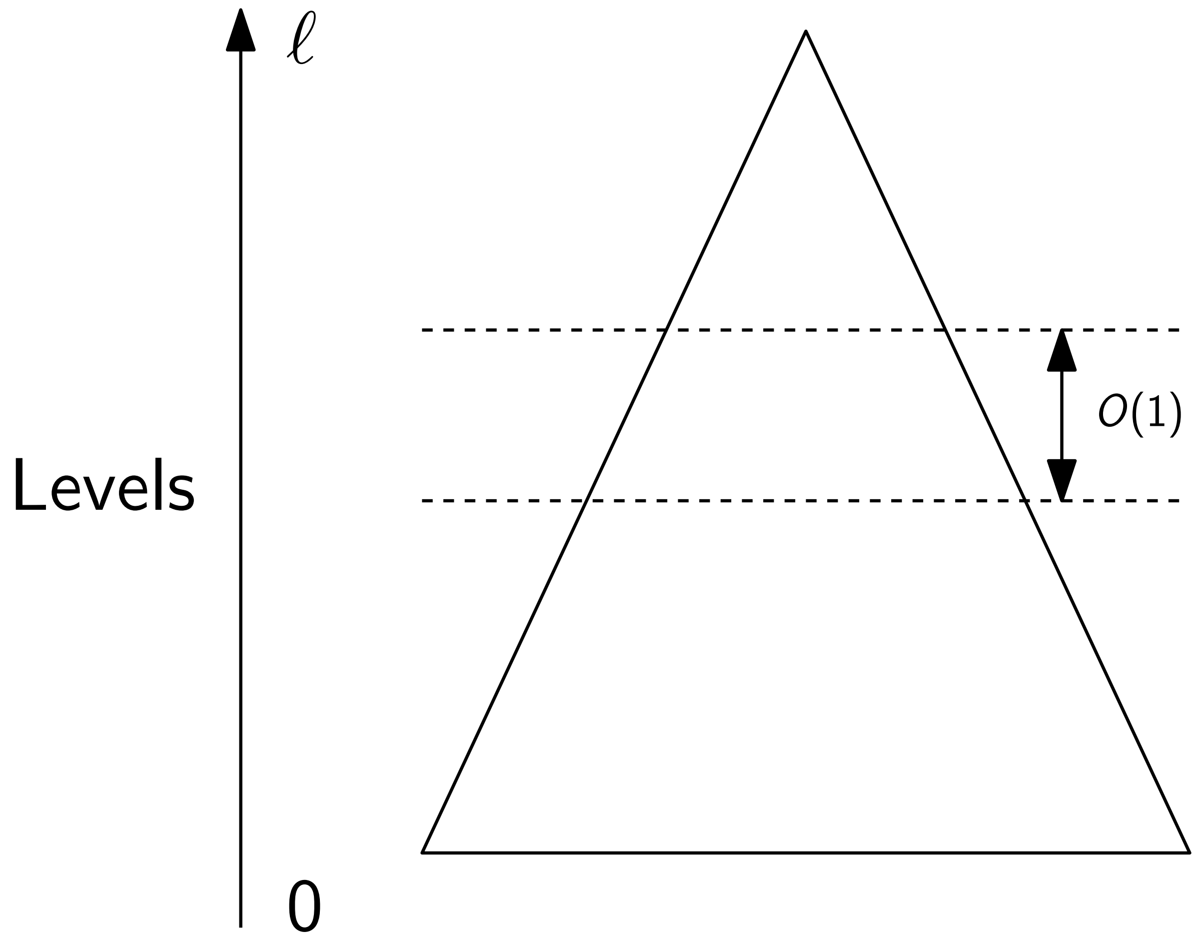
$$Z_k^v = \sum_{i \in [n]} \chi[i \in I_k] \cdot \chi[h_k(i) = v] \cdot \epsilon_i \cdot x_i \cdot 1/p_k,$$

   where $\Pr[\epsilon_i = +1] = \Pr[\epsilon_i = -1] = 1/2$ for all $i \in [n]$.
   The summations/multiplications are coordinate-wise.

At the end, $\mu(x)$ consists of $\lambda^{O(1)} \cdot l = \lambda^{O(1)}$ cells/coordinates.

The idea of analysis

For each class



0 w.h.p.:
no item sampled

Isolated in $O(1)$ levels

small w.h.p. :
they cancel each other heavily since we multiply $\{+1, -1\}$s.

Earth-Mover Distance

Earth-Mover Distance

- Given two multisets A, B in the grid $[\Delta]^2$ with $|A| = |B| = N$, the *Earth-Mover Distance* (EMD) is defined as the minimum cost of a perfect matching between points in A and B, that is,

$$\mathrm{EMD}(A, B) = \min_{\pi : A \to B} \sum_{a \in A} \|a - \pi(a)\|_1 .$$

Earth-Mover Distance

- Given two multisets A, B in the grid $[\Delta]^2$ with $|A| = |B| = N$, the *Earth-Mover Distance* (EMD) is defined as the minimum cost of a perfect matching between points in A and B, that is,

$$\text{EMD}(A, B) = \min_{\pi: A \to B} \sum_{a \in A} \|a - \pi(a)\|_1 .$$

- **Our goal**: construct a small sketch ν such that
  1. $\text{EMD}(A, B) \leq \textit{dist}(\nu(A), \nu(B)) \leq C \cdot \text{EMD}(A, B)$.
  2. $\nu(A)$ and $\nu(B)$ can be stored in a small space S.

Earth-Mover Distance

- Given two multisets A, B in the grid $[\Delta]^2$ with $|A| = |B| = N$, the *Earth-Mover Distance* (EMD) is defined as the minimum cost of a perfect matching between points in A and B, that is,

$$\mathrm{EMD}(A, B) = \min_{\pi : A \to B} \sum_{a \in A} \|a - \pi(a)\|_1 .$$

- **Our goal**: construct a small sketch ν such that
  1. $\mathrm{EMD}(A, B) \leq \mathit{dist}(\nu(A), \nu(B)) \leq C \cdot \mathrm{EMD}(A, B)$.
  2. $\nu(A)$ and $\nu(B)$ can be stored in a small space S.

- Better if ν is a linear sketch (support streaming computation). Even better if it is an embedding.

Earth-Mover Distance

- A special case of Kantorovich metric, which is proposed by Nobel prize winner L. V. Kantorovich in an article in 1942. This metric has numerous applications in
  - Image retrieval.
  - Data mining
  - Probablistic concurrency.
  - Bioinformatics.
  - etc ...

Earth-Mover Distance

- A special case of Kantorovich metric, which is proposed by Nobel prize winner L. V. Kantorovich in an article in 1942. This metric has numerous applications in
  - Image retrieval.
  - Data mining
  - Probablistic concurrency.
  - Bioinformatics.
  - etc ...
- A major open problem
  Find a tradeoff between C and S.

Previous work

- Offline algorithm
  1. General exact: Hungarian method. Time complexity $O(N^3)$.
  2. Planar exact: Vaidya '89, Agarwal, Efrat & Sharir '95. $O(N^{2+\delta})$.
  3. Planar approximate: Agarwal & Varadarajan '99, Charikar '02, Indyk & Thaper 2003, Agarwal & Varadarajan '04, Indyk '07.

     Current best $O(1)$-approx: Indyk '07 with time $O(N \log^{O(1)} N)$.

Previous work

- Offline algorithm

  1. General exact: Hungarian method. Time complexity $O(N^3)$.
  2. Planar exact: Vaidya '89, Agarwal, Efrat & Sharir '95. $O(N^{2+\delta})$.
  3. Planar approximate: Agarwal & Varadarajan '99, Charikar '02, Indyk & Thaper 2003, Agarwal & Varadarajan '04, Indyk '07.
     Current best $O(1)$-approx: Indyk '07 with time $O(N \log^{O(1)} N)$.

- Linear sketch:

  1. Charikar '02, Indyk & Thaper '03 can be modified to get approximation ratio $O(\log \Delta)$ using $O(\log \Delta)$ space.
  2. Andoni-Do Ba-Indyk-Woodruff.'09 proposed a sketching algorithm with Δ^ϵ space and $O(1/\epsilon)$ approximation,

Previous work

- Offline algorithm

  1. General exact: Hungarian method. Time complexity $O(N^3)$.
  2. Planar exact: Vaidya '89, Agarwal, Efrat & Sharir '95. $O(N^{2+\delta})$.
  3. Planar approximate: Agarwal & Varadarajan '99, Charikar '02, Indyk & Thaper 2003, Agarwal & Varadarajan '04, Indyk '07.

     Current best $O(1)$-approx: Indyk '07 with time $O(N \log^{O(1)} N)$.

- Linear sketch:

  Embed into ℓ_1, but $\Omega(\sqrt{\log \Delta})\ldots$

  1. Charikar '02, Indyk & Thaper '03 can be modified to get approximation ratio $O(\log \Delta)$ using $O(\log \Delta)$ space.
  2. Andoni-Do Ba-Indyk-Woodruff.'09 proposed a sketching algorithm with Δ^ϵ space and $O(1/\epsilon)$ approximation,

Previous work

- Offline algorithm

  1. General exact: Hungarian method. Time complexity $O(N^3)$.
  2. Planar exact: Vaidya '89, Agarwal, Efrat & Sharir '95. $O(N^{2+\delta})$.
  3. Planar approximate: Agarwal & Varadarajan '99, Charikar '02, Indyk & Thaper 2003, Agarwal & Varadarajan '04, Indyk '07.

     Current best $O(1)$-approx: Indyk '07 with time $O(N \log^{O(1)} N)$.

- Linear sketch:

  Embed into ℓ_1, but $\Omega(\sqrt{\log \Delta})\ldots$

  1. Charikar '02, Indyk & Thaper '03 can be modified to get approximation ratio $O(\log \Delta)$ using $O(\log \Delta)$ space.
  2. Andoni-Do Ba-Indyk-Woodruff.'09 proposed a sketching algorithm with Δ^ϵ space and $O(1/\epsilon)$ approximation,

     Not an embedding

A few more notations

- EEMD is an extension of EMD to any multisets $A, B \subseteq [\Delta]^2$ (not necessary the same size) defined as:

$$\begin{aligned}\mathrm{EEMD}_{\Delta}(A, B) &= \min_{S \subseteq A, S' \subseteq B, |S|=|S'|} [\mathrm{EMD}(S, S') \\ &\quad + \Delta(|A - S'| + |B - S'|)].\end{aligned}$$

  Note that if $|A| = |B|$, then $\mathrm{EMD}(A, B) = \mathrm{EEMD}(A, B)$.

- EEMD can be induced by a norm $\|\cdot\|_{\mathrm{EEMD}}$.

A few more notations

- EEMD is an extension of EMD to any multisets $A, B \subseteq [\Delta]^2$ (not necessary the same size) defined as:

$$\begin{aligned} \text{EEMD}_\Delta(A, B) &= \min_{S \subseteq A, S' \subseteq B, |S| = |S'|} [\text{EMD}(S, S') \\ &\quad + \Delta(|A - S'| + |B - S'|)]. \end{aligned}$$

  Note that if $|A| = |B|$, then $\text{EMD}(A, B) = \text{EEMD}(A, B)$.

- EEMD can be induced by a norm $\|\cdot\|_{\text{EEMD}}$.

- For a multiset $A \subseteq [\Delta]^2$, let $x(A) \in \mathbb{R}^{\Delta^2}$ be the characteristic vector of A.

- For $x \in \mathbb{X}^n$, let $\|x\|_{1,\mathbb{X}} = \sum_{i \in [n]} \|x_i\|_{\mathbb{X}}$.

Theorem for EMD

For any $\epsilon \in (0, 1)$, there exists a distribution over linear mappings $\nu : \mathrm{EEMD}_{\Delta} \rightarrow \ell_1^{\Delta^{O(\epsilon)}} \otimes \mathrm{EEMD}_{\Delta^{\epsilon}}$, such that for any two $A, B \subseteq [\Delta]^2$ of equal size, we have

- $\|\nu(x(A) - x(B))\|_{1,\mathrm{EEMD}} \geq \Omega(\mathrm{EMD}(A, B))$ with pr. $1 - o(1)$.
- $\|\nu(x(A) - x(B))\|_{1,\mathrm{EEMD}} \leq O(1/\epsilon \cdot \mathrm{EMD}(A, B))$ with pr. $2/3$.

Theorem for EMD

For any $\epsilon \in (0,1)$, there exists a distribution over linear mappings $\nu : \mathrm{EEMD}_\Delta \to \ell_1^{\Delta^{O(\epsilon)}} \otimes \mathrm{EEMD}_{\Delta^\epsilon}$, such that for any two $A, B \subseteq [\Delta]^2$ of equal size, we have

- $\|\nu(x(A) - x(B))\|_{1,\mathrm{EEMD}} \geq \Omega(\mathrm{EMD}(A,B))$ with pr. $1 - o(1)$.
- $\|\nu(x(A) - x(B))\|_{1,\mathrm{EEMD}} \leq O(1/\epsilon \cdot \mathrm{EMD}(A,B))$ with pr. $2/3$.

Total space used $\Delta^{O(\epsilon)}$

Compared with Andoni et. al. (FOCS 2009)

We got the same bounds as Andoni-Do Ba-Indyk-Woodruff But,

- Our sktech is a linear embedding to a product normed space.

  While theirs needs *binary decisions* which do not admit efficient embeddings.

- Our sketch algorithm is much simpler.

- Our effective use of the low-“Rademacher dimensionality” property of a norm space may be of independent interest.

Two steps towards the goal

Main Theorem. Let $\mathbb{X}$ be a normed space with Rademacher dimension α. Let $\lambda = \max\{\alpha, \log^3 n\}$. Then there exists a distribution over linear mappings $\mu : \ell_1^n \otimes \mathbb{X} \to \ell_1^{\lambda^{O(1)}} \otimes \mathbb{X}$, s.t.

- $\|\mu(x)\|_{1,\mathbb{X}} \geq \Omega(\|x\|_{1,\mathbb{X}})$ with pr. $1 - 1/\lambda^{\Omega(1)}$.
- $\|\mu(x)\|_{1,\mathbb{X}} \leq O(\|x\|_{1,\mathbb{X}})$ with pr. 0.99.

Two steps towards the goal

Main Theorem. Let $\mathbb{X}$ be a normed space with Rademacher dimension α. Let $\lambda = \max\{\alpha, \log^3 n\}$. Then there exists a distribution over linear mappings $\mu : \ell_1^n \otimes \mathbb{X} \to \ell_1^{\lambda^{O(1)}} \otimes \mathbb{X}$, s.t.

- $\|\mu(x)\|_{1,\mathbb{X}} \geq \Omega(\|x\|_{1,\mathbb{X}})$ with pr. $1 - 1/\lambda^{\Omega(1)}$.
- $\|\mu(x)\|_{1,\mathbb{X}} \leq O(\|x\|_{1,\mathbb{X}})$ with pr. 0.99.

$+$ EEMD_Δ has Rademacher dimension $\alpha = \Delta^4$

Two steps towards the goal

Main Theorem. Let $\mathbb{X}$ be a normed space with Rademacher dimension α. Let $\lambda = \max\{\alpha, \log^3 n\}$. Then there exists a distribution over linear mappings $\mu : \ell_1^n \otimes \mathbb{X} \to \ell_1^{\lambda^{O(1)}} \otimes \mathbb{X}$, s.t.

- $\|\mu(x)\|_{1,\mathbb{X}} \geq \Omega(\|x\|_{1,\mathbb{X}})$ with pr. $1 - 1/\lambda^{\Omega(1)}$.
- $\|\mu(x)\|_{1,\mathbb{X}} \leq O(\|x\|_{1,\mathbb{X}})$ with pr. 0.99.

$+$ EEMD_Δ has Rademacher dimension $\alpha = \Delta^4$

$+$ **Fact.** (Indyk 07) There exists a distribution overlinear mappings $F : \text{EEMD}_\Delta \to \ell_1^n \otimes \text{EEMD}_{\Delta^\epsilon}$ $(n = \Delta^{O(1)})$, s.t.

- $\|x\|_{\text{EEMD}} \leq \|F(x)\|_{1,\text{EEMD}}$ with pr. 1.
- $\|F(x)\|_{1,\text{EEMD}} \leq O(1/\epsilon) \cdot \|x\|_{\text{EEMD}}$ with pr. 4/5.

$\Longrightarrow$ **Thm for** EMD

Open problems

- Can we do better for EMD?
  Or (approx C, space S) $= (1/\epsilon, \Delta^{\epsilon})$ is the lower bound?

Open problems

- Can we do better for EMD?
  Or (approx C, space S) $= (1/\epsilon, \Delta^{\epsilon})$ is the lower bound?
- Can we apply our technique for arbitrary (or a class of) metrics $\mathbb{X}$?

Open problems

- Can we do better for EMD?
  Or (approx C, space S) $= (1/\epsilon, \Delta^\epsilon)$ is the lower bound?
- Can we apply our technique for arbitrary (or a class of) metrics $\mathbb{X}$?
- How about embedding $\ell_2^n \otimes \mathbb{X} \to \ell_2^{n'} \otimes \mathbb{X}$?
  Does strong/better embeddings exist?

Open problems

- Can we do better for EMD?
  Or (approx C, space S) $= (1/\epsilon, \Delta^\epsilon)$ is the lower bound?
- Can we apply our technique for arbitrary (or a class of) metrics $\mathbb{X}$?
- How about embedding $\ell_2^n \otimes \mathbb{X} \to \ell_2^{n'} \otimes \mathbb{X}$?
  Does strong/better embeddings exist?
- Can we obtain strong embeddings
  $\ell_1^n \otimes \mathbb{X} \to \ell_{1/2}^{\log^{O(1)}} \otimes \ell_1^{n'} \otimes \mathbb{X}$? (Elad's conjecture)

Open problems

- Can we do better for EMD?

  Or (approx C, space S) $= (1/\epsilon, \Delta^{\epsilon})$ is the lower bound?

- Can we apply our technique for arbitrary (or a class of) metrics $\mathbb{X}$?

- How about embedding $\ell_2^n \otimes \mathbb{X} \to \ell_2^{n'} \otimes \mathbb{X}$?
  Does strong/better embeddings exist?

- Can we obtain strong embeddings
  $\ell_1^n \otimes \mathbb{X} \to \ell_{1/2}^{\log^{O(1)}} \otimes \ell_1^{n'} \otimes \mathbb{X}$? (Elad's conjecture)

  Andoni-Indyk-Krauthgamer showed Algorithms for product metrics of the form $\ell^r_{(\ell_2)^2} \otimes \ell^s_{\infty} \otimes \ell^t_1$ (SODA'09)

Thank you!