CS 263: Counting and Sampling

Nima Anari

slides for

High-Dimensional Expanders
Review

### Monomer-dimer system

Prob of matching $\propto \prod_{e \in M} \lambda_e \cdot \prod_{v \not\in M} z_v$

Assume wlog $z_v = 1$

- [Jerrum-Sinclair]: Metropolis mixes in $\text{poly}(n, \lambda_{\text{max}})$
- [Jerrum-Sinclair-Vigoda]: on bipartite graphs, can sample in $\text{poly}(n, \log \lambda_{\text{max}})$

Open: non-bipartite graphs

Challenging example: chain of boxes

To resolve we reweigh: $\mu(M) \propto \lambda^M \cdot \Omega^{\text{monomers}(M)}$

Metropolis on perfect and near-perfect $\propto \mu$ mixes fast.

The problem: how to estimate $\lambda(\Omega_S)$?

Solution: slowly change official name: cooling schedule $\lambda$
Monomer-dimer system

Prob of matching $\propto \prod_{e \in M} \lambda_e \cdot \prod_{v \not\in M} z_v$

- dimer
- monomer

Assume wlog $z_v = 1$
Review

Monomer-dimer system

Prob of matching $\propto \prod_{e \in M} \lambda_e \cdot \prod_{v \not\in M} z_v$

- Assume wlog $z_v = 1$
- [Jerrum-Sinclair]: Metropolis mixes in $\text{poly}(n, \lambda_{\text{max}})$
Monomer-dimer system

Prob of matching \( \propto \prod_{e \in \mathcal{M}} \lambda_e \cdot \prod_{v \not\in \mathcal{M}} z_v \)

- Assume wlog \( z_v = 1 \)
- [Jerrum-Sinclair]: Metropolis mixes in \( \text{poly}(n, \lambda_{\text{max}}) \)
- [Jerrum-Sinclair-Vigoda]: on bipartite graphs, can sample in \( \text{poly}(n, \log \lambda_{\text{max}}) \)
Monomer-dimer system

Prob of matching \( \propto \prod_{e \in M} \lambda_e \cdot \prod_{v \not\in M} z_v \)

- Assume wlog \( z_v = 1 \)
- [Jerrum-Sinclair]: Metropolis mixes in \( \text{poly}(n, \lambda_{\text{max}}) \)
- [Jerrum-Sinclair-Vigoda]: on bipartite graphs, can sample in \( \text{poly}(n, \log \lambda_{\text{max}}) \)
- Open: non-bipartite graphs
Review

Monomer-dimer system

Prob of matching $\propto \prod_{e \in M} \lambda_e \cdot \prod_{v \not\in M} z_v$

Assume wlog $z_v = 1$

[Jerrum-Sinclair]: Metropolis mixes in $\text{poly}(n, \lambda_{\text{max}})$

[Jerrum-Sinclair-Vigoda]: on bipartite graphs, can sample in $\text{poly}(n, \log \lambda_{\text{max}})$

Open: non-bipartite graphs

Challenging example: chain of boxes
**Monomer-dimer system**

Prob of matching \( \propto \prod_{e \in M} \lambda_e \cdot \prod_{v \not\in M} z_v \)

- Assume wlog \( z_v = 1 \)
- [Jerrum-Sinclair]: Metropolis mixes in \( \text{poly}(n, \lambda_{\text{max}}) \)
- [Jerrum-Sinclair-Vigoda]: on bipartite graphs, can sample in \( \text{poly}(n, \log \lambda_{\text{max}}) \)
- Open: non-bipartite graphs

**Challenging example: chain of boxes**

- To resolve we reweigh:
  \[ \mu(M) \propto \frac{\lambda^M}{\lambda(\Omega_{\text{monomers}}(M))} \]

- Metropolis on perfect and near-perfect \( \propto \mu \) mixes fast.

- The problem: how to estimate \( \lambda_{\Omega(S)} \)?

- Solution: slowly change \( \lambda \), official name: cooling schedule
Review

Monomer-dimer system

Prob of matching $\propto \prod_{e \in M} \lambda_e \cdot \prod_{v \not\in M} z_v$

Assume wlog $z_v = 1$

[Jerrum-Sinclair]: Metropolis mixes in $\text{poly}(n, \lambda_{\text{max}})$

[Jerrum-Sinclair-Vigoda]: on bipartite graphs, can sample in $\text{poly}(n, \log \lambda_{\text{max}})$

Open: non-bipartite graphs

Challenging example: chain of boxes

To resolve we reweigh:

$$\mu(M) \propto \frac{\lambda^M}{\lambda(\Omega_{\text{monomers}(M)})}$$

Metropolis on perfect and near-perfect $\propto \mu$ mixes fast.
Monomer-dimer system

Prob of matching $\propto \prod_{e \in M} \lambda_e \cdot \prod_{v \not\sim M} z_v$

Assume wlog $z_v = 1$

[Jerrum-Sinclair]: Metropolis mixes in $\text{poly}(n, \lambda_{\text{max}})$

[Jerrum-Sinclair-Vigoda]: on bipartite graphs, can sample in $\text{poly}(n, \log \lambda_{\text{max}})$

Open: non-bipartite graphs

Challenging example: chain of boxes

To resolve we reweigh:

$$\mu(M) \propto \frac{\lambda^M}{\lambda(\Omega_{\text{monomers}(M)})}$$

Metropolis on perfect and near-perfect $\propto \mu$ mixes fast.

The problem: how to estimate $\lambda(\Omega_S)$?
Review

**Monomer-dimer system**

Prob of matching $\propto \prod_{e \in M} \lambda_e \cdot \prod_{v \not\in M} z_v$

- Assume wlog $z_v = 1$
- [Jerrum-Sinclair]: Metropolis mixes in $\text{poly}(n, \lambda_{\text{max}})$
- [Jerrum-Sinclair-Vigoda]: on bipartite graphs, can sample in $\text{poly}(n, \log \lambda_{\text{max}})$
- Open: non-bipartite graphs

**Challenging example: chain of boxes**

- To resolve we reweigh: $\mu(M) \propto \frac{\lambda^M}{\lambda(\Omega_{\text{monomers}(M)})}$
- Metropolis on perfect and near-perfect $\propto \mu$ mixes fast.
- The problem: how to estimate $\lambda(\Omega_S)$?
- Solution: slowly change $\lambda$

official name: cooling schedule
Note that $\lambda$ as $\lambda_t = \lambda_e \lambda_f \lambda_x$. Let $e$'s endpoints be $a, a'$ and $f$'s endpoints be $b, b'$.

Prove: via injective map $\lambda(\Omega_{\emptyset}) \geq 1$ poly$(n) \cdot \lambda(e \lambda(f \lambda(x))) \lambda(\Omega_{\{a, b\}} \lambda(\Omega_{\{u, v, a', b'\}} enc)).$

Thus $\mu(s) \leq poly(n) \cdot \mu(x) \mu(\text{enc})$. Similar ineqs yield $\mu(s) \mu(t) \leq poly(n) \cdot \mu(y) \mu(\text{enc})$. So cong $\leq poly(n)$. 

\[ s \oplus t \oplus x \]
Note that \( \lambda_s \lambda_t = \lambda_e \lambda_f \lambda_x \). Let \( e \)'s endpoints be \( a, a' \) and \( f \)'s endpoints be \( b, b' \).

Prove: via injective map 
\[
\lambda(\Omega \emptyset t) \geq 1 \text{poly}(n) \cdot \lambda(\Omega \{u,v\} s) \\
\leq \lambda(\Omega \{a,b\} x) \\
\leq \lambda(\Omega \{u,v,a',b'\} enc)
\]
Thus \( \mu(s) \mu(t) \leq \text{poly}(n) \cdot \mu(x) \mu(enc) \).

Similar ineqs yield \( \mu(s) \mu(t) \leq \text{poly}(n) \cdot \mu(y) \mu(enc) \).

So cong \( \leq \text{poly}(n) \).
\[
\text{enc} = s \oplus t \oplus x - e - f
\]

Note that \( \lambda^s \lambda^t = \lambda_e \lambda_f \lambda_x \lambda_{\text{enc}} \). Let \( e \)'s endpoints be \( a, a' \) and \( f \)'s endpoints be \( b, b' \). Prove: via injective map

\[
\lambda(\Omega_{\emptyset})\lambda(\Omega_{\{u,v\}}) \geq \frac{1}{\text{poly}(n)} \cdot \lambda_e \lambda_f \lambda(\Omega_{\{a,b\}})\lambda(\Omega_{\{u,v,a',b'\}})
\]
\[
\text{enc} = s \oplus t \oplus \chi - e - f
\]

Note that \(\lambda^s \lambda^t = \lambda_e \lambda_f \lambda^x \lambda^\text{enc}\). Let \(e\)'s endpoints be \(a, a'\) and \(f\)'s endpoints be \(b, b'\). Prove: \(\lambda(\Omega_\emptyset) \lambda(\Omega_{\{u,v\}}) \geq \frac{1}{\text{poly}(n)} \cdot \lambda_e \lambda_f \lambda(\Omega_{\{a,b\}}) \lambda(\Omega_{\{u,v,a',b'\}})\)

Thus \(\mu(s) \mu(t) \leq \text{poly}(n) \cdot \mu(x) \mu(\text{enc})\). Similar ineqs yield \(\mu(s) \mu(t) \leq \text{poly}(n) \cdot \mu(y) \mu(\text{enc})\). So cong \(\leq \text{poly}(n)\).
Down-Up Walks
- Matroids
- Down and up kernels
- Simplicial complex

High-Dimensional Expansion
- Local-to-global
Down-Up Walks
- Matroids
- Down and up kernels
- Simplicial complex

High-Dimensional Expansion
- Local-to-global
Sample $S$ with $\mathbb{P}[S] \propto \mu(S)$:

$$\mu : \binom{U}{k} \rightarrow \mathbb{R}_{\geq 0}$$

weighted hypergraph
Example: matroids

\[ U = \{ \} \]

\[ \mu (\{ \}) = 1 \]

\[ \mu (\{ \}) = 0 \]

\[ \mu (\{ \}) = 0 \]
Example: matroids

\[ u = \left\{ \begin{array}{c} \rightarrow, \ \rightarrow, \ \rightarrow, \ \rightarrow, \ \rightarrow, \ \rightarrow \end{array} \right\} \]
Example: matroids

$\mu \left( \left\{ \begin{array}{c} \rightarrow, \quad \rightarrow, \quad \downarrow, \quad \rightarrow, \quad \rightarrow, \quad \rightarrow \end{array} \right\} \right) = 1$

uniform on bases
Example: matroids

\[ u = \left\{ \begin{array}{c}
\rightarrow, \\
\rightarrow, \\
\downarrow, \\
\uparrow, \\
\rightarrow,
\end{array} \right\} \]

\[ \nabla \quad \mu \left( \left\{ \begin{array}{c}
\rightarrow, \\
\rightarrow, \\
\downarrow,
\end{array} \right\} \right) = 1 \]

\[ \nabla \quad \mu \left( \left\{ \begin{array}{c}
\rightarrow, \\
\uparrow, \\
\rightarrow
\end{array} \right\} \right) = 1 \]
Example: matroids

\[ u = \left\{ \begin{array}{c}
\begin{array}{c}
\uparrow \\
\rightarrow \\
\rightarrow \\
\downarrow \\
\rightarrow
\end{array}
\end{array} \right\} \]

\[ \mu \left( \left\{ \begin{array}{c}
\begin{array}{c}
\uparrow \\
\rightarrow \\
\rightarrow \\
\downarrow \\
\rightarrow
\end{array}
\end{array} \right\} \right) = 1 \]

\[ \mu \left( \left\{ \begin{array}{c}
\begin{array}{c}
\uparrow \\
\rightarrow \\
\rightarrow \\
\downarrow \\
\rightarrow
\end{array}
\end{array} \right\} \right) = 1 \]

\[ \mu \left( \left\{ \begin{array}{c}
\begin{array}{c}
\uparrow \\
\rightarrow \\
\rightarrow \\
\downarrow \\
\rightarrow
\end{array}
\end{array} \right\} \right) = 0 \]
Example: matroids

$\mu\biggl(\\{\quad\quad\quad\quad\biggr\}\biggr) = 1$

$\mu\biggl(\\{\quad\quad\quad\biggr\}\biggr) = 1$

$\mu\biggl(\\{\quad\quad\biggr\}\biggr) = 0$

$\mu\biggl(\\{\quad\biggr\}\biggr) = 0$
Note: matroids larger than linear matroids. Results apply to all.
Note: matroids larger than linear matroids. Results apply to all.

Matroids are self-reducible:

\[
\begin{pmatrix}
 a & b & c & d & e & f & g \\
 u & +1 & 0 & 0 & 0 & 0 & +1 \\
 v & 0 & -1 & +1 & 0 & -1 & 0 \\
 w & 0 & 0 & -1 & +1 & 0 & 0 \\
 x & -1 & +1 & 0 & 0 & 0 & 0 \\
 y & 0 & 0 & 0 & -1 & 0 & +1 & +1 \\
\end{pmatrix}
\]
Note: matroids larger than linear matroids. Results apply to all.

Matroids are self-reducible:

Example: hypercube
Matroid on \(\{\pm 1, \ldots, \pm 1_n\}\)
Note: matroids larger than linear matroids. Results apply to all.

Matroids are self-reducible:

Example: hypercube
Matroid on \(\{\pm 1, \ldots, \pm 1_n\}\)

Example: spanning trees
Note: matroids larger than linear matroids. Results apply to all.

Matroids are self-reducible:

**Example: hypercube**
Matroid on \{±1_1, \ldots, ±1_n\}

**Example: spanning trees**

![Graph with vertices u, v, w, x, y, e, c, f, d, g and adjacency matrix](vertices-edges-adj-matrix.png)

Vertex-edge adj matrix

\[
\begin{bmatrix}
a & b & c & d & e & f & g \\
u & +1 & 0 & 0 & 0 & +1 & 0 & 0 \\
v & 0 & -1 & +1 & 0 & -1 & -1 & 0 \\
w & 0 & 0 & -1 & +1 & 0 & 0 & 0 \\
x & -1 & +1 & 0 & 0 & 0 & 0 & -1 \\
y & 0 & 0 & 0 & -1 & 0 & +1 & +1 \\
\end{bmatrix}
\]
Down-up walks

Random walk:
1. Drop element u.a.r.
2. Add element with prob. \( \propto \mu(\text{resulting set}) \).

\[ \mu(S) = 1 [S \text{ is basis}] . \]
Down-up walks

Random walk:

1. Drop element u.a.r.
2. Add element with prob. $\propto \mu$(resulting set).

$\mu(S) = 1[S \text{ is basis}].$
Down-up walks

Random walk:
1. Drop element u.a.r.
2. Add element with prob. \( \propto \mu(\text{resulting set}) \).

\[ \mu(S) = 1[S \text{ is basis}] \]
Down-up walks

Random walk:

1. Drop element u.a.r.
2. Add element with prob. $\propto \mu(\text{resulting set})$.

$$\mu(S) = 1 [S \text{ is basis}].$$
Down-up walks

Random walk:

1. Drop element u.a.r.
2. Add element with prob. \( \propto \mu(\text{resulting set}). \)

\[ \mu(S) = 1 [S \text{ is basis}] . \]
Down-up walks

Random walk:

1. Drop element u.a.r.
2. Add element with prob. \( \propto \mu(\text{resulting set}) \).

\[ \mu(S) = 1[S \text{ is basis}] . \]
Down-up walks

Random walk:

1. Drop element u.a.r.
2. Add element with prob. \( \propto \mu(\text{resulting set}) \).

\[ \mu(S) = 1 \text{[S is basis]} \]
Down-up walks

Random walk:
1. Drop element u.a.r.
2. Add element with prob. $\propto \mu(\text{resulting set})$.

$[\text{A-Liu-OveisGharan-Vinzant’19}]$

$t_{\text{mix}} = O(k^2 \log n)$

$[\text{Cryan-Guo-Mousa’20}]$

$[\text{A-Liu-OveisGharan-Vinzant-Vuong’21}]$

$\mu(S) = 1[S \text{ is basis}]$. 
Down-up walks

Random walk:
1. Drop element u.a.r.
2. Add element with prob. $\propto \mu(\text{resulting set})$.

[A-Liu-OveisGharan-Vinzant’19]
$t_{\text{mix}} = O(k^2 \log n)$ $\leftarrow$ Poincaré

[Cryan-Guo-Mousa’20]
$t_{\text{mix}} = O(k(\log k + \log \log n))$ $\leftarrow$ MLSI

$\mu(S) = 1$ [S is basis].
Down-up walks

Random walk:
1. Drop element u.a.r.
2. Add element with prob. $\propto \mu(\text{resulting set}).$

[A-Liu-OveisGharan-Vinzant'19]
$t_{\text{mix}} = O(k^2 \log n)$ → Poincaré

[Cryan-Guo-Mousa'20]
$t_{\text{mix}} = O(k(\log k + \log \log n))$ → MLSI

[A-Liu-OveisGharan-Vinzant-Vuong'21]
$t_{\text{mix}} = O(k \log k)$ → exchange

$\mu(S) = 1 [S \text{ is basis}].$
Down-up walk is of the form

\[ \mathbb{N} \to \mathbb{N}^\circ \]

down  up

Distribution on level \( \ell \):
\[ \mu_\mathbb{D}_k \to \ell \]

Time-reversals are up kernels:
\[ U_\ell \to k = D_k \to \ell \]

Down kernels \( D_k \to \ell \) is Markov kernel from \( \Omega = (U_k) \) to \( \Omega' = (U_\ell) \): drop \( k - \ell \) u.r. elements.

Algorithmically useful:
\[ \ell = k - 1 \] or more generally
\[ \ell = k - O(1) \]

\[ \cdots \]
Down-up walk is of the form

\[ \text{NN} \circ \text{NN}^o \]

down up

**Down kernels**

\( D_{k \rightarrow \ell} \) is Markov kernel from \( \Omega = \left( \binom{u}{k} \right) \)
to \( \Omega' = \left( \binom{u}{\ell} \right) \): drop \( k - \ell \) u.r. elements.
Down-up walk is of the form

Down kernels

$D_{k \to \ell}$ is Markov kernel from $\Omega = \binom{u}{k}$ to $\Omega' = \binom{u}{\ell}$: drop $k - \ell$ u.r. elements.

Distribution on level $\ell$: $\mu_{D_{k \to \ell}}$
Down-up walk is of the form

\[ \mathcal{N} \mathcal{N}^\circ \]

**Down kernels**

\( D_{k \to \ell} \) is Markov kernel from \( \Omega = \binom{\mathcal{U}}{k} \) to \( \Omega' = \binom{\mathcal{U}}{\ell} \): drop \( k - \ell \) u.r. elements.

**Distribution on level \( \ell \):** \( \mu D_{k \to \ell} \)

**Time-reversals are up kernels:**

\[ \mathcal{U}_{\ell \to k} = D_{k \to \ell}^\circ \]
Down-up walk is of the form

\[ \text{Down kernels} \]

\[ D_{k \rightarrow \ell} \] is Markov kernel from \( \Omega = \binom{\ell}{k} \) to \( \Omega' = \binom{\ell}{\ell} \): drop \( k - \ell \) u.r. elements.

\[ \text{Distribution on level } \ell: \mu D_{k \rightarrow \ell} \]

\[ \text{Time-reversals are up kernels:} \]

\[ U_{\ell \rightarrow k} = D_{k \rightarrow \ell}^\circ \]

\[ k \leftrightarrow \ell \text{ down-up walk: } D_{k \rightarrow \ell} U_{\ell \rightarrow k} \]
Down-up walk is of the form

\[ \text{Down kernels} \]
\[ D_{k \rightarrow \ell} \text{ is Markov kernel from } \Omega = \binom{u}{k} \text{ to } \Omega' = \binom{u}{\ell} : \text{ drop } k - \ell \text{ u.r. elements.} \]

- Distribution on level \( \ell \): \( \mu D_{k \rightarrow \ell} \)
- Time-reversals are up kernels:
  \[ U_{\ell \rightarrow k} = D_{k \rightarrow \ell}^\circ \]
- \( k \leftrightarrow \ell \) down-up walk: \( D_{k \rightarrow \ell} U_{\ell \rightarrow k} \)
- Algorithmically useful: \( \ell = k - 1 \) or more generally \( \ell = k - O(1) \)
HDX recipe

1. Dist on $\left( \binom{u}{k} \right)$ ← simplicial complex
2. Local: show $\mathcal{D}_\phi$ contraction for $D_{k\rightarrow 1}$
3. Global: transfer contraction for $D_{k\rightarrow 1}$ to $D_{k\rightarrow k-1}$

Example: coloring $U = \{1, 2, \ldots\}$

$$\mu(\emptyset) = 0$$
$$\mu(\{1, 2, \ldots\}) = 1$$
HDX recipe

1. Dist on \( \binom{U}{k} \) \( \leftarrow \) simplicial complex
2. Local: show \( \mathcal{D}_\phi \) contraction for \( \mathcal{D}_{k \rightarrow 1} \)
3. Global: transfer contraction for \( \mathcal{D}_{k \rightarrow 1} \) to \( \mathcal{D}_{k \rightarrow k-1} \)

How to convert to simplicial complex?

For product spaces \( \Omega_1 \times \cdots \times \Omega_n \):

\( \Omega_1 \sqcup \cdots \sqcup \Omega_n \)

Put zero mass on invalid sets.

Example: coloring \( U = \{ , \ldots , \} \)

\( \mu(\{ , , \}) = 0 \)

\( \mu(\{ , , \}) = 0 \)

\( \mu(\{ , , \}) = 1 \)
HDX recipe

1. Dist on \( \binom{u}{k} \) ← simplicial complex
2. **Local:** show \( D_\phi \) contraction for \( D_{k \rightarrow 1} \)
3. **Global:** transfer contraction for \( D_{k \rightarrow 1} \) to \( D_{k \rightarrow k-1} \)

How to convert to simplicial complex?

For product spaces \( \Omega_1 \times \cdots \times \Omega_n \):

\( \Omega_1 \sqcup \cdots \sqcup \Omega_n \)

Put zero mass on invalid sets.

Example: coloring \( U = \{ \} \):

\[
\mu(\{ \}, \{ \}, \{ \}, \ldots) = 0
\]

\[
\mu(\{ \}, \{ \}, \{ \}, \ldots) = 0
\]

\[
\mu(\{ \}, \{ \}, \{ \}, \ldots) = 1
\]
HDX recipe

1. Dist on \( \binom{u}{k} \) \leftarrow simplicial complex
2. Local: show \( D_\phi \) contraction for \( D_{k\rightarrow 1} \)
3. Global: transfer contraction for \( D_{k\rightarrow 1} \) to \( D_{k\rightarrow k-1} \)

How to convert to simplicial complex?

For product spaces \( \Omega_1 \times \cdots \times \Omega_n \):

\[
(\Omega_1 \sqcup \cdots \sqcup \Omega_n)
\]

Put zero mass on invalid sets.

Example: coloring \( U = \{\ldots\} \)

\[
\mu(\{\ldots\}) = 0
\]
HDX recipe

1. Dist on \( \binom{U}{k} \) \( \leftarrow \) simplicial complex
2. Local: show \( \mathcal{D}_\phi \) contraction for \( \mathcal{D}_{k \rightarrow 1} \)
3. Global: transfer contraction for \( \mathcal{D}_{k \rightarrow 1} \) to \( \mathcal{D}_{k \rightarrow k-1} \)

How to convert to simplicial complex?

For product spaces \( \Omega_1 \times \cdots \times \Omega_n \):
\[
(\Omega_1 \sqcup \cdots \sqcup \Omega_n)
\]

Put zero mass on invalid sets.

Example: coloring \( U = \{1, 2, 3, \ldots\} \)

\[
\mu(\{1, 2, 3\}) = 0
\]

\[
\mu(\{1, 2, 3\}) = 0
\]

\[
\mu(\{1, 2, 3\}) = 1
\]
HDX recipe

1. Dist on \( \binom{u}{k} \) \( \leftrightarrow \) simplicial complex
2. Local: show \( D_\phi \) contraction for \( D_{k \rightarrow 1} \)
3. Global: transfer contraction for \( D_{k \rightarrow 1} \) to \( D_{k \rightarrow k-1} \)

- How to convert to simplicial complex?
- For product spaces \( \Omega_1 \times \cdots \times \Omega_n : \)
  \( (\Omega_1 \sqcup \cdots \sqcup \Omega_n)_n \)
- Put zero mass on invalid sets.
HDX recipe

1. Dist on $\binom{u}{k} \leftrightarrow$ simplicial complex
2. Local: show $\mathcal{D}_\phi$ contraction for $\mathcal{D}_{k \rightarrow 1}$
3. Global: transfer contraction for $\mathcal{D}_{k \rightarrow 1}$ to $\mathcal{D}_{k \rightarrow k-1}$

> How to convert to simplicial complex?

> For product spaces $\Omega_1 \times \cdots \times \Omega_n$:

\[
(\Omega_1 \sqcup \cdots \sqcup \Omega_n)
\]

> Put zero mass on invalid sets.

Example: coloring

\[
u = \left\{ \begin{array}{c}
\begin{array}{c}
\cdots
\end{array}
\end{array} \right\}
\]
HDX recipe

1. Dist on \( \binom{u}{k} \) \( \leftarrow \) simplicial complex
2. Local: show \( \mathcal{D}_\phi \) contraction for \( \mathcal{D}_{k \to 1} \)
3. Global: transfer contraction for \( \mathcal{D}_{k \to 1} \) to \( \mathcal{D}_{k \to k - 1} \)

How to convert to simplicial complex?

For product spaces \( \Omega_1 \times \cdots \times \Omega_n \):

\[
\left( \Omega_1 \sqcup \cdots \sqcup \Omega_n \right)
\]

Put zero mass on invalid sets.

Example: coloring

\( u = \{ \text{\includegraphics{example-coloring}} \} \)

\[
\mu\left( \left\{ \text{\includegraphics{example-coloring}} \right\} \right) = 0
\]
HDX recipe

1. Dist on $\binom{U}{k} \leftrightarrow$ simplicial complex
2. Local: show $D_\phi$ contraction for $D_{k\to1}$
3. Global: transfer contraction for $D_{k\to1}$ to $D_{k\to k-1}$

How to convert to simplicial complex?

For product spaces $\Omega_1 \times \cdots \times \Omega_n$:

$$(\Omega_1 \sqcup \cdots \sqcup \Omega_n)$$

Put zero mass on invalid sets.

Example: coloring

$u = \{ \text{valid sets} \}$

$\mu(\{ \text{valid sets} \}) = 0$

$\mu(\{ \text{invalid sets} \}) = 0$
**HDX recipe**

1. Dist on $\binom{\mathcal{U}}{k} \leftarrow$ simplicial complex
2. Local: show $\mathcal{D}_\phi$ contraction for $D_{k \to 1}$
3. Global: transfer contraction for $D_{k \to 1}$ to $D_{k \to k-1}$

How to convert to simplicial complex?

For product spaces $\Omega_1 \times \cdots \times \Omega_n$:

$$(\Omega_1 \sqcup \cdots \sqcup \Omega_n)$$

Put zero mass on invalid sets.

**Example: coloring**

$\mathcal{U} = \{ \{ \ldots \} \}$

- $\mu(\{ \ldots \}) = 0$
- $\mu(\{ \ldots \}) = 0$
- $\mu(\{ \ldots \}) = 1$
Down-Up Walks

- Matroids
- Down and up kernels
- Simplicial complex

High-Dimensional Expansion

- Local-to-global
Down-Up Walks

- Matroids
- Down and up kernels
- Simplicial complex

High-Dimensional Expansion

- Local-to-global
Goal: analyze $\mathcal{D}_\phi$ contraction for the Markov kernels $\mathcal{D}_{k \rightarrow \ell}$.
Goal: analyze $\mathcal{D}_\phi$ contraction for the Markov kernels $\mathcal{D}_{k\rightarrow\ell}$

From data processing, we know

$$\mathcal{D}_\phi(\nu \parallel \mu) - \mathcal{D}_\phi(\nu \mathcal{D}_{k\rightarrow\ell} \parallel \mu \mathcal{D}_{k\rightarrow\ell})$$

equals

$$\mathbb{E}_{T \sim \mu \mathcal{D}_{k\rightarrow\ell}} \left[ \text{Ent}^\phi_{\mathcal{U}_{\ell\rightarrow k}(T, \cdot)} \left[ \frac{\nu}{\mu} \right] \right]$$
Goal: analyze $\mathcal{D}_\phi$ contraction for the Markov kernels $\mathcal{D}_{k\rightarrow\ell}$

From data processing, we know

$$\mathcal{D}_\phi(\nu \parallel \mu) - \mathcal{D}_\phi(\nu \mathcal{D}_{k\rightarrow\ell} \parallel \mu \mathcal{D}_{k\rightarrow\ell})$$

equals

$$\mathbb{E}_{T \sim \mu \mathcal{D}_{k\rightarrow\ell}} \left[ \text{Ent}^\phi_{U_{\ell\rightarrow k}(T, \cdot)} \left[ \frac{\nu}{\mu} \right] \right]$$

### Conditionals

For a set $|T| = \ell$ and dist $\mu$ on $\binom{U}{k}$, we call $U_{\ell\rightarrow k}(T, \cdot)$ the **conditional dist**:

$$\text{dist}_{S \sim \mu}(S \mid T \subseteq S)$$
Goal: analyze $\mathcal{D}_\phi$ contraction for the Markov kernels $D_{k\rightarrow \ell}$

From data processing, we know
\[ \mathcal{D}_\phi(\nu \| \mu) - \mathcal{D}_\phi(\nu D_{k\rightarrow \ell} \| \mu D_{k\rightarrow \ell}) \]
equals
\[ \mathbb{E}_{T \sim \mu D_{k\rightarrow \ell}} \left[ \operatorname{Ent}_U^\phi(U_{\ell\rightarrow k}(T, \cdot)) \left[ \frac{\nu}{\mu} \right] \right] \]

Conditionals

For a set $|T| = \ell$ and dist $\mu$ on $\binom{U}{k}$, we call $U_{\ell\rightarrow k}(T, \cdot)$ the conditional dist:
\[ \operatorname{dist}_{S \sim \mu}(S \mid T \subseteq S) \]

Links

Removing the known part $T$, we get links $\mu_T$:
\[ \mu_T = \operatorname{dist}_{S \sim \mu}(S - T \mid T \subseteq S) \]
Goal: analyze $\mathcal{D}_\phi$ contraction for the Markov kernels $D_{k \to \ell}$

From data processing, we know

$$\mathcal{D}_\phi(\nu \parallel \mu) - \mathcal{D}_\phi(\nu D_{k \to \ell} \parallel \mu D_{k \to \ell})$$

equals

$$\mathbb{E}_{T \sim \mu D_{k \to \ell}} \left[ \text{Ent}^\phi_{U_{\ell \to k}(T, \cdot)} \left[ \frac{\nu}{\mu} \right] \right]$$

Conditionals

For a set $|T| = \ell$ and dist $\mu$ on $\binom{U}{k}$, we call $U_{\ell \to k}(T, \cdot)$ the conditional dist:

$$\text{dist}_{S \sim \mu}(S \mid T \subseteq S)$$

Links

Removing the known part $T$, we get links $\mu_T$:

$$\mu_T = \text{dist}_{S \sim \mu}(S - T \mid T \subseteq S)$$

Example: matroids

Conditionals: project all in $U - T$ on $\text{span}(T)^\perp$.

Links: remove $T$ and set vector space to $\text{span}(T)^\perp$.

Links of matroids are matroids. 😊
Theorem [Alev-Lau, ...]

Suppose for each $|T| = t$, we know $D_{k-t\rightarrow 1}$ contracts $\mathcal{D}_\phi(\cdot \parallel \mu_T)$ by $1 - \rho_t$. Then

$$\frac{\mathcal{D}_\phi(\nu D_{k\rightarrow \ell} \parallel \mu D_{k\rightarrow \ell})}{\mathcal{D}_\phi(\nu \parallel \mu)} \leq 1 - \rho,$$

where

$$\rho = \rho_0 \rho_1 \cdots \rho_{l-1}.$$
Local-to-global

**Theorem [Alev-Lau, ...]**

Suppose for each $|T| = t$, we know $D_{k-t \rightarrow 1}$ contracts $\mathcal{D}_\phi(\cdot \mid \mu_T)$ by $1-\rho_t$. Then

\[
\frac{\mathcal{D}_\phi(\nu D_{k \rightarrow \ell} \mid \mu D_{k \rightarrow \ell})}{\mathcal{D}_\phi(\nu \mid \mu)} \leq 1 - \rho,
\]

where

\[
\rho = \rho_0 \rho_1 \cdots \rho_{l-1}.
\]

▷ For matroids and $\mathcal{D}_\phi \in \{\chi^2, \mathcal{D}_{KL}\}$:

\[
\rho_0 \geq 1 - \frac{1}{k}.
\]
Local-to-global

**Theorem [Alev-Lau, ...]**

Suppose for each $|T| = t$, we know $D_{k-t\to 1}$ contracts $\mathcal{D}_\phi(\cdot \parallel \mu_T)$ by $1-\rho_t$. Then

$$\frac{\mathcal{D}_\phi(\nu D_{k\to \ell} \parallel \mu D_{k\to \ell})}{\mathcal{D}_\phi(\nu \parallel \mu)} \leq 1 - \rho,$$

where

$$\rho = \rho_0 \rho_1 \cdots \rho_{l-1}.$$

This automatically means $\rho_t \geq 1 - \frac{1}{k-t}.$

For matroids and $\mathcal{D}_\phi \in \{\chi^2, \mathcal{D}_{\text{KL}}\}$:

$$\rho_0 \geq 1 - \frac{1}{k}.$$
Local-to-global

Theorem [Alev-Lau, …]

Suppose for each $|T| = t$, we know $D_{k \to t} \rightarrow 1$ contracts $D_{\phi}(\cdot \| \mu_T)$ by $1 - \rho_t$. Then

$$D_{\phi}(\nu D_{k \to \ell} \| \mu D_{k \to \ell}) \leq 1 - \rho,$$

where

$$\rho = \rho_0 \rho_1 \cdots \rho_{l-1}.$$

- This automatically means $\rho_t \geq 1 - \frac{1}{k-t}$.
- By local-to-global, for $D_{k \to \ell}$, $\rho \geq \left(1 - \frac{1}{k}\right) \cdots \left(1 - \frac{1}{k-\ell+1}\right) = \frac{k-\ell}{k}$

For matroids and $D_{\phi} \in \{\chi^2, D_{KL}\}$:

$$\rho_0 \geq 1 - \frac{1}{k}.$$
Theorem [Alev-Lau, …]

Suppose for each $|T| = t$, we know $D_{k-t \rightarrow 1}$ contracts $D_\phi(\cdot \parallel \mu_T)$ by $1 - \rho_t$. Then

$$\frac{D_\phi(\nu D_{k \rightarrow \ell} \parallel \mu D_{k \rightarrow \ell})}{D_\phi(\nu \parallel \mu)} \leq 1 - \rho,$$

where

$$\rho = \rho_0 \rho_1 \cdots \rho_{l-1}.$$

This automatically means

$$\rho_t \geq 1 - \frac{1}{k-t}.$$  

By local-to-global, for $D_{k \rightarrow \ell}$, $\rho \geq \left(1 - \frac{1}{k}\right) \cdots \left(1 - \frac{1}{k - \ell + 1}\right) = \frac{k - \ell}{k}$

For $\ell = k - 1$ (algorithmically relevant), we get $\rho \geq 1/k$. 😊

For matroids and $D_\phi \in \{\chi^2, D_{KL}\}$:

$$\rho_0 \geq 1 - \frac{1}{k}.$$
Local-to-global

**Theorem [Alev-Lau, ...]**

Suppose for each $|T| = t$, we know $D_{k\to t} \rightarrow 1$ contracts $D_{\phi}(\cdot \| \mu_T)$ by $1 - \rho_t$. Then

$$\frac{D_{\phi}(\nu D_{k\to \ell} \| \mu D_{k\to \ell})}{D_{\phi}(\nu \| \mu)} \leq 1 - \rho,$$

where

$$\rho = \rho_0 \rho_1 \cdots \rho_{l-1}.$$

- This automatically means $\rho_t \geq 1 - \frac{1}{k-t}$.
- By local-to-global, for $D_{k\to \ell}$, $\rho \geq \frac{(1 - \frac{1}{k}) \cdots (1 - \frac{1}{k-\ell+1})}{k} = \frac{k - \ell}{k}$.
- For $\ell = k - 1$ (algorithmically relevant), we get $\rho \geq 1/k$.
- This transfers to down-up walk. By data processing, $U_{k\to k-1}$ cannot increase $D_{\phi}$.

For matroids and $D_{\phi} \in \{\chi^2, D_{KL}\}$:

$$\rho_0 \geq 1 - \frac{1}{k}$$
$\mathcal{D}_\phi$ contracts by $1 - \rho$
$1 - \rho$ contraction

down step

no increase

up step
Proof:

- Consider $f = \nu/\mu$. 

...
Proof:

- Consider \( f = \nu / \mu \).
- We will track quantity:

\[
z_T = \text{Ent}^\Phi_{U_{|T| \to k}(T, \cdot)}[f]
\]
Proof:

- Consider $f = \nu/\mu$.
- We will track quantity:
  \[ z_T = \text{Ent}^\phi_{u_{|T|\to_k}(T, \cdot)}[f] \]

- Consider process $T_0 \subseteq T_1 \subseteq \ldots$:
  sample $S \sim \mu$, and u.a.r. permute its elements to be $S = \{e_1, \ldots, e_k\}$,
  and let $T_i = \{e_1, \ldots, e_i\}$.
Proof:

- Consider $f = \nu/\mu$.
- We will track quantity:
  
  $$z_T = \text{Ent}^{\phi}_{U|T|\rightarrow k(T, \cdot)}[f]$$

- Consider process $T_0 \subseteq T_1 \subseteq \ldots$:
  sample $S \sim \mu$, and u.a.r. permute its elements to be $S = \{e_1, \ldots, e_k\}$,
  and let $T_i = \{e_1, \ldots, e_i\}$.
- Claim: from assumption
  
  $$\mathbb{E}[z_{T_{i+1}} \mid T_i] \geq \rho_i z_{T_i}.$$
Proof:

▷ Consider $f = \nu/\mu$.

▷ We will track quantity:

$$z_T = \text{Ent}^\phi_{U|T| \to k(T, \cdot)}[f]$$

▷ Consider process $T_0 \subseteq T_1 \subseteq \ldots$:

- sample $S \sim \mu$, and u.a.r. permute its elements to be $S = \{e_1, \ldots, e_k\}$,
- and let $T_i = \{e_1, \ldots, e_i\}$.

▷ Claim: from assumption

$$\mathbb{E}[z_{T_{i+1}} \mid T_i] \geq \rho_i z_{T_i}.$$ 

▷ Chaining the inequalities we get

$$\mathbb{E}[z_{T_\ell}] \geq \rho_0 \cdots \rho_{\ell-1} z_0.$$
Proof:

- Consider $f = \nu/\mu$.
- We will track quantity:

$$z_T = \text{Ent}^\phi_{U|T|\to k(T, \cdot)}[f]$$

- Consider process $T_0 \subseteq T_1 \subseteq \ldots$:
  sample $S \sim \mu$, and u.a.r. permute its elements to be $S = \{e_1, \ldots, e_k\}$, and let $T_i = \{e_1, \ldots, e_i\}$.

- Claim: from assumption

$$\mathbb{E}[z_{T_{i+1}} | T_i] \geq \rho_i z_{T_i}.$$

- Chaining the inequalities we get

$$\mathbb{E}[z_{T_\ell}] \geq \rho_0 \cdots \rho_{\ell-1} z_0.$$

- This is the same as

$$\frac{\mathcal{D}_\phi(\nu \mathcal{D}_{k\to \ell} \parallel \mu \mathcal{D}_{k\to \ell})}{\mathcal{D}_\phi(\nu \parallel \mu)} \leq 1 - \rho,$$

where

$$\rho = \rho_0 \cdots \rho_{\ell-1}.$$