MModes: Computing Modes of a Graphical Model

Chao Chen
chao.chen@qc.cuny.edu
http://eniac.cs.qc.cuny.edu/cchen/

CUNY Queens College & CUNY Graduate Center

October 2015

Outline

Graphical Models

MModes Problem

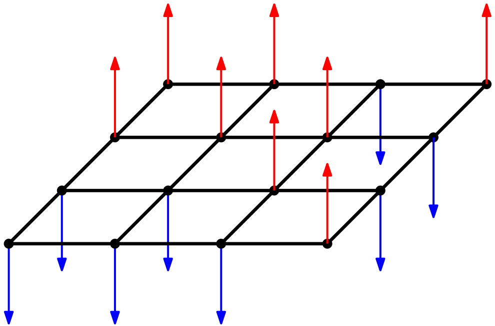
Algorithms

Applications

Verification and Local Search Algorithm

The First Example: Ising Model

- Statistical physics: ferromagnetism, Ernst Ising (1924, PhD Thesis)
- Atoms in a lattice: magnetic dipole moments (up/down)
- Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, Labels $\mathcal{L} = \{+1, -1\} = \{\text{up}, \text{down}\}$
- All configurations (labelings): $\mathcal{Y} = \{+1, -1\}^{\mathcal{V}}$
- Energy: $E(y) = \sum_{(i,j)\in\mathcal{E}} E_{i,j}(y_i, y_j) + \sum_{i\in\mathcal{V}} E_i(y_i)$
- Probability: $P(y) = \exp(-E(y))/Z$, where $Z = \sum_{y\in\mathcal{Y}} P(y)$



Ising Model (Cont'd)

- All configurations (labelings): $\mathcal{Y} = \{+1, -1\}^{|\mathcal{V}|}$
- Energy
  $E(y) = \sum_{(i,j)\in\mathcal{E}} \theta(y_i, y_j) + \sum_{i\in\mathcal{V}} \mu(y_i)$
- Probability
  $P(y) = \exp\left(-E(y)\right)/Z$, where $Z = \sum_{y\in\mathcal{Y}} P(y)$
- In this example: $E(y) = 12\alpha + 12\beta + 8\gamma + 8\eta$

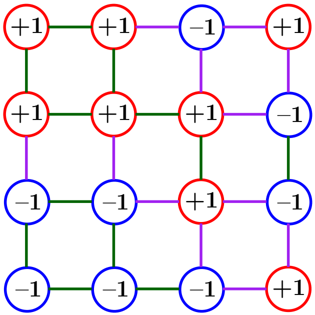
Parameters:

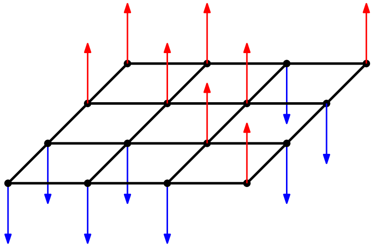
$\theta(y_i, y_j)$

y_i \ y_j	+1	-1
+1	α	β
-1	β	α

$\mu(y_i)$

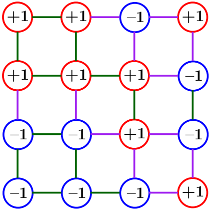
y_i	
+1	γ
-1	η





Graphical Model

- Ising model:
  - A grid graph
  - A discrete distribution
  - The domain has 2^{16} configurations/labelings
  - 4 parameters $(\alpha, \beta, \gamma, \eta)$

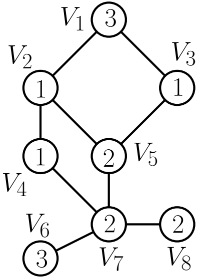


y_i \ y_j	+1	-1
+1	α	β
-1	β	α

y_i	
+1	γ
-1	η

Graphical Model (cont'd)

- Labelings: $\mathcal{Y} = \mathcal{L}^{|\mathcal{V}|} = \{1, \cdots, L\}^{|\mathcal{V}|}$
- Parameters: $[\vec{\theta}, \vec{\mu}]$, $|\mathcal{E}| \cdot L^2 + |\mathcal{V}| \cdot L$ long



Binaries

$y_i \backslash y_j$	1	2	$\cdots$	L
1	$\theta_{ij}(1,1)$	$\theta_{ij}(1,2)$	$\cdots$	$\theta_{ij}(1,L)$
2	$\theta_{ij}(2,1)$	$\theta_{ij}(2,2)$	$\cdots$	$\theta_{ij}(2,L)$
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
L	$\theta_{ij}(L,1)$	$\theta_{ij}(L,2)$	$\cdots$	$\theta_{ij}(L,L)$

Unaries

y_i	
1	$\mu_i(1)$
2	$\mu_i(2)$
$\cdots$	$\cdots$
L	$\mu_i(L)$

- Energy: $E(y) = \sum_{(i,j)\in\mathcal{E}} \theta_{ij}(y_i, y_j) + \sum_{i\in\mathcal{V}} \mu_i(y_i)$
- Probability: $P(y) = \exp\left(-E(y)\right)/Z$
- Partition function: $Z = \sum_{y\in\mathcal{Y}} P(y)$
- Conditional independent: $\forall (i,j) \notin \mathcal{E}$, $V_i \perp V_j \mid \mathcal{V}\backslash\{V_i, V_j\}$

Applications

A Graphical Model

- A very flexible model for high dimensional discrete/continuous data
- Explicitly capture and leverage structures between data
- A playground for researchers interested in both statisitical modeling and algorithm design

Example applications

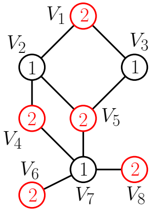
- Neuronscience, fMRI, biomedicine etc.
- General Vision

Outline

Problems

Inference

- Computing the maximum a posteriori (MAP): $\operatorname{argmax}_{y \in \mathcal{Y}} P(y) = \operatorname{argmin}_{y \in \mathcal{Y}} E(y)$ NP-hard (reduction from maximal independent set problem)

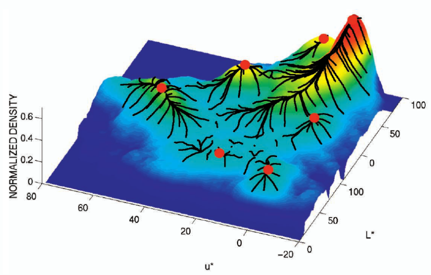


$y_i \backslash y_j$	1	2
1	0	0
2	0	∞

y_i	
1	1
2	0

New Problem: Computing Modes

- Modes: local maxima
  - Very global characterization of the distribution
  - Generating new hypotheses



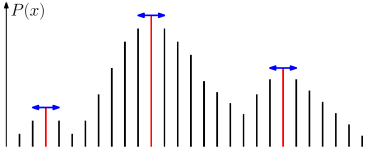
Comaniciu and Meer, PAMI 2002

New Problem: Computing Modes

- D the dimension; $\mathcal{L} = \{1, \ldots, L\}$ the label set; $\mathcal{X} = \mathcal{L}^D$ the domain
- Given a distance function $d(\cdot, \cdot)$ and a scalar δ
  - Neighborhood: $N_\delta(x) = \{x' \mid d(x, x') \leq \delta\}$
  - x is a mode if it has a bigger prob. than all its neighbors
  - $\mathcal{M}^\delta$: the set of all modes for a given scale δ

New Problem: Computing Modes

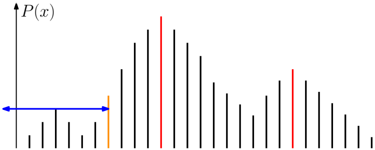
- D the dimension; $\mathcal{L} = \{1, \ldots, L\}$ the label set; $\mathcal{X} = \mathcal{L}^D$ the domain
- Given a distance function $d(\cdot, \cdot)$ and a scalar δ
  - Neighborhood: $N_\delta(x) = \{x' \mid d(x, x') \leq \delta\}$
  - x is a mode if it has a bigger prob. than all its neighbors
  - $\mathcal{M}^\delta$: the set of all modes for a given scale δ



$\delta = 1$

New Problem: Computing Modes

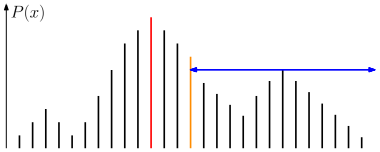
- D the dimension; $\mathcal{L} = \{1, \ldots, L\}$ the label set; $\mathcal{X} = \mathcal{L}^D$ the domain
- Given a distance function $d(\cdot, \cdot)$ and a scalar δ
  - Neighborhood: $N_\delta(x) = \{x' \mid d(x, x') \leq \delta\}$
  - x is a mode if it has a bigger prob. than all its neighbors
  - $\mathcal{M}^\delta$: the set of all modes for a given scale δ



$\delta = 4$

New Problem: Computing Modes

- D the dimension; $\mathcal{L} = \{1, \ldots, L\}$ the label set; $\mathcal{X} = \mathcal{L}^D$ the domain
- Given a distance function $d(\cdot, \cdot)$ and a scalar δ
  - Neighborhood: $N_\delta(x) = \{x' \mid d(x, x') \leq \delta\}$
  - x is a mode if it has a bigger prob. than all its neighbors
  - $\mathcal{M}^\delta$: the set of all modes for a given scale δ

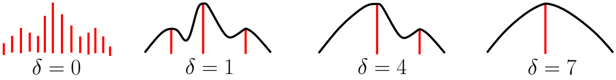


$\delta = 7$

New Problem: Computing Modes

- D the dimension; $\mathcal{L} = \{1, \ldots, L\}$ the label set; $\mathcal{X} = \mathcal{L}^D$ the domain
- Given a distance function $d(\cdot, \cdot)$ and a scalar δ
  - Neighborhood: $N_\delta(x) = \{x' \mid d(x, x') \leq \delta\}$
  - x is a mode if it has a bigger prob. than all its neighbors
  - $\mathcal{M}^\delta$: the set of all modes for a given scale δ

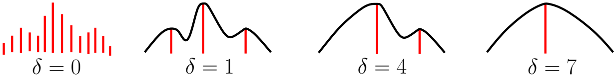
$$\mathcal{X} = \mathcal{M}^0 \supseteq \mathcal{M}^1 \supseteq \cdots \supseteq \mathcal{M}^\infty = \{\text{global maximum}\}$$



New Problem: Computing Modes

- D the dimension; $\mathcal{L} = \{1, \ldots, L\}$ the label set; $\mathcal{X} = \mathcal{L}^D$ the domain
- Given a distance function $d(\cdot, \cdot)$ and a scalar δ
  - Neighborhood: $N_\delta(x) = \{x' \mid d(x, x') \leq \delta\}$
  - x is a mode if it has a bigger prob. than all its neighbors
  - $\mathcal{M}^\delta$: the set of all modes for a given scale δ

$$\mathcal{X} = \mathcal{M}^0 \supseteq \mathcal{M}^1 \supseteq \cdots \supseteq \mathcal{M}^\infty = \{\text{global maximum}\}$$



Problem

MModes Given the scale δ, computer the top M elements in $\mathcal{M}^\delta$.

[**CC** *et al.* AISTATS 2013, NIPS 2014, submitted to JMLR]

Setting

Trees or Chains

$G = (\mathcal{V}, \mathcal{E})$

$\mathcal{V} = \{1, \ldots, D\}, \quad \mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$

Setting

Trees or Chains

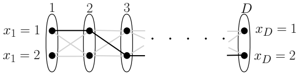
$G = (\mathcal{V}, \mathcal{E})$

$\mathcal{V} = \{1, \dots, D\}, \quad \mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$

the set of labels: $\mathcal{L} = \{1, \dots, L\}$

labeling $x = (x_1, \dots, x_D), x_i \in \mathcal{L}$

The space of labelings $\mathcal{X} = \mathcal{L}^{|\mathcal{V}|}$



Setting

Trees or Chains

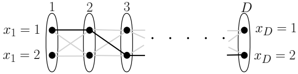
$G = (\mathcal{V}, \mathcal{E})$

$\mathcal{V} = \{1, \ldots, D\}, \quad \mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$

the set of labels: $\mathcal{L} = \{1, \ldots, L\}$

labeling $x = (x_1, \ldots, x_D), x_i \in \mathcal{L}$

The space of labelings $\mathcal{X} = \mathcal{L}^{|\mathcal{V}|}$



$$E(x) = \sum_{(i,j) \in \mathcal{E}} \theta_{i,j}(x_i, x_j) \qquad P(x) = \frac{1}{Z} \exp\left(-E(x)\right)$$

Setting

Trees or Chains

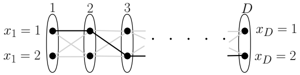
$G = (\mathcal{V}, \mathcal{E})$

$\mathcal{V} = \{1, \ldots, D\}, \quad \mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$

the set of labels: $\mathcal{L} = \{1, \ldots, L\}$

labeling $x = (x_1, \ldots, x_D), x_i \in \mathcal{L}$

The space of labelings $\mathcal{X} = \mathcal{L}^{|\mathcal{V}|}$



$$E(x) = \sum_{(i,j) \in \mathcal{E}} \theta_{i,j}(x_i, x_j) \qquad P(x) = \frac{1}{Z} \exp\left(-E(x)\right)$$

Setting

Trees or Chains

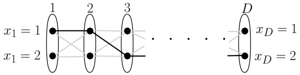
$G = (\mathcal{V}, \mathcal{E})$

$\mathcal{V} = \{1, \dots, D\}, \quad \mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$

the set of labels: $\mathcal{L} = \{1, \dots, L\}$

labeling $x = (x_1, \dots, x_D), x_i \in \mathcal{L}$

The space of labelings $\mathcal{X} = \mathcal{L}^{|\mathcal{V}|}$

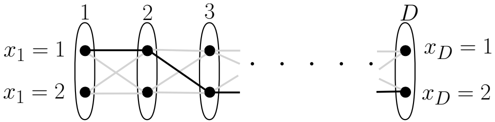


$$E(x) \quad = \quad \sum_{(i,j) \in \mathcal{E}} \theta_{i,j}(x_i, x_j) \qquad P(x) = \frac{1}{Z} \exp\left(-E(x)\right)$$

- global optimal (max P, min E)
  - Trees: polynomial
  - Graphs: mostly NP-hard

Algorithm: Chains

- MAP = global minimum



Optimal path problem (dynamic programming), $O(DL^2)$

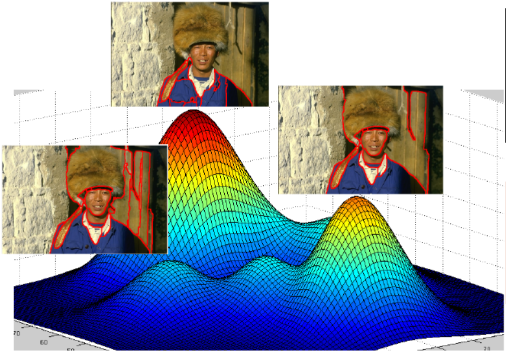
- MBest: best M labelings

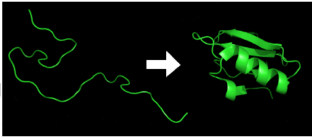
$$x^1 = \operatorname*{argmin}_{x \in \mathcal{X}} E(x)$$
$$x^m = \operatorname*{argmin}_{x \in \mathcal{X} \setminus \{x^1, \cdots, x^{m-1}\}} E(x)$$

Nilsson'98 $O(DL^2 + MDL + MD\log(MD))$

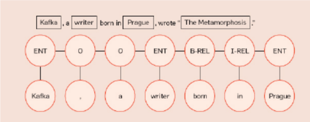
Motivation

- Reason:
  - model is not perfect, ambiguity
  - multiple hypotheses, diverse, highly possible





Protein Folding



NLP

Outline

Algorithm: Modes on Chains

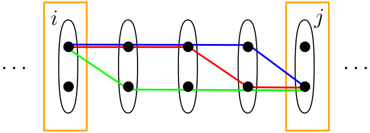
Idea: divide and conquer

- The whole chain $[1, D] \to$ subchains $[i, j]$ of a fixed length

Algorithm: Modes on Chains

Idea: divide and conquer

- The whole chain $[1, D] \rightarrow$ subchains $[i, j]$ of a fixed length
- A partial labeling $x_{i:j}$

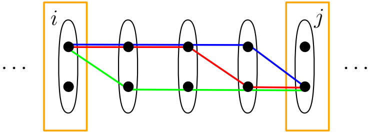


- $x_{i:j}$ is a local mode iff for any $y_{i:j}$ s.t. $y_i = x_i$, $y_j = x_j$
  $E(x_{i:j}) < E(y_{i:j})$

Algorithm: Modes on Chains

Idea: divide and conquer

- The whole chain $[1, D] \rightarrow$ subchains $[i, j]$ of a fixed length
- A partial labeling $x_{i:j}$



- $x_{i:j}$ is a local mode iff for any $y_{i:j}$ s.t. $y_i = x_i$, $y_j = x_j$
  $E(x_{i:j}) < E(y_{i:j})$

Lemma

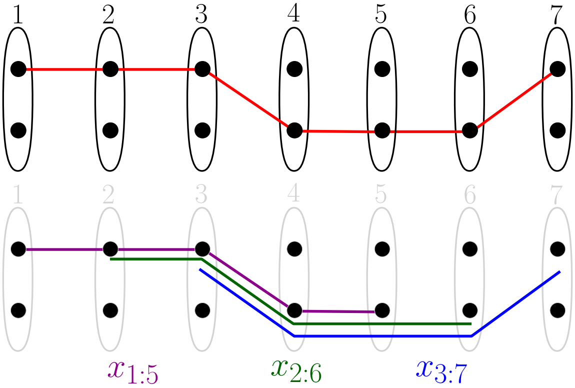
any $[i, j]$ has L^2 local modes, computable in polynomial time

Algorithm: Modes on Chains

Theorem (local-global)

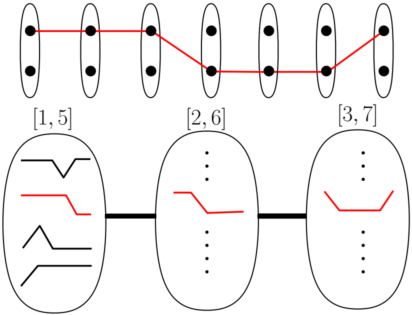
x is a mode iff any length $\delta + 2$ partial labeling $x_{i:j}$ is a local mode

An example: $D = 7$, $\delta = 3$



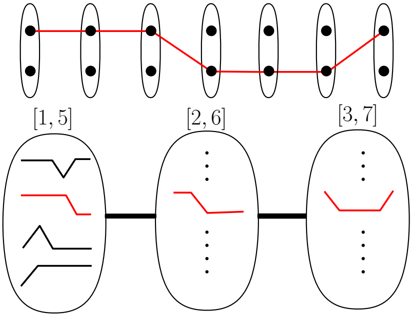
Algorithm: Modes on Chains

- Intuition
  - Combinations of local modes → global modes
  - Consistent: agree at common vertices



Algorithm: Modes on Chains

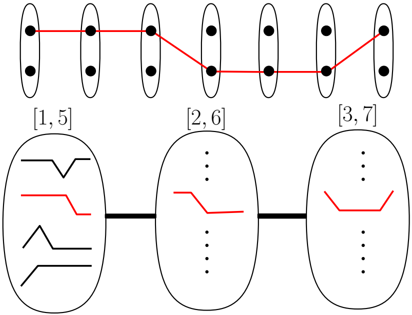
- Intuition
  - Combinations of local modes $\rightarrow$ global modes
  - Consistent: agree at common vertices



- Step 1: construct a new chain,
  - supernodes $[i, j]$
  - labels $\{$local modes of $[i, j]\}$
  - feasible only if consistent
  - preserve the energy of the original graph

Algorithm: Modes on Chains

- Intuition
  - Combinations of local modes $\rightarrow$ global modes
  - Consistent: agree at common vertices



- Step 1: construct a new chain,
  - supernodes $[i, j]$
  - labels $\{$local modes of $[i, j]\}$
  - feasible only if consistent
  - preserve the energy of the original graph

Fact

New chain labeling space: $\widehat{\mathcal{X}} = \mathcal{M}^{\delta}$

Algorithm: Modes on Chains

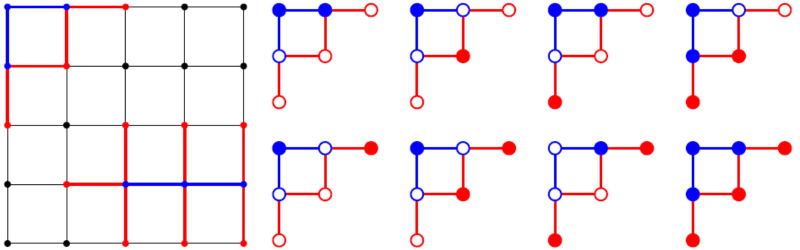
- Step 1: construct a new chain,
  - Labeling space: $\widehat{\mathcal{X}} = \mathcal{M}^{\delta}$
  - Energy: $\widehat{E}(\widehat{x}) = E(x)$
  - L DP on $O(D)$ length δ intervals
  - Complexity : $O(DL^3\delta)$
- Step 2: M-Modes is reduced to M-Best in the new chain
  - M-Best: compute the top M labelings
  - Use Nilsson'98
  - $O(DL^4 + MDL^2 + MD\log(MD))$
  - Can reduce to $O(DL^3 + MDL^2 + MD\log(MD))$
- Total Complexity $O(DL^3\delta + MDL^2 + MD\log(MD))$

Local Global

algorithms. For general graph, we can prove the following theorem. Denote by $\mathcal{S} = \{S_1, S_2, \cdots\}$ the set of size-δ subgraphs of a given graph G. Let $\bar{S}_i$ be the closure of S_i, namely, $S_i \cup \{v \notin S_i | \exists u \in S_i, (u, v) \in \mathcal{E}\}$. Let $\partial S_i = \bar{S}_i \backslash S_i$.

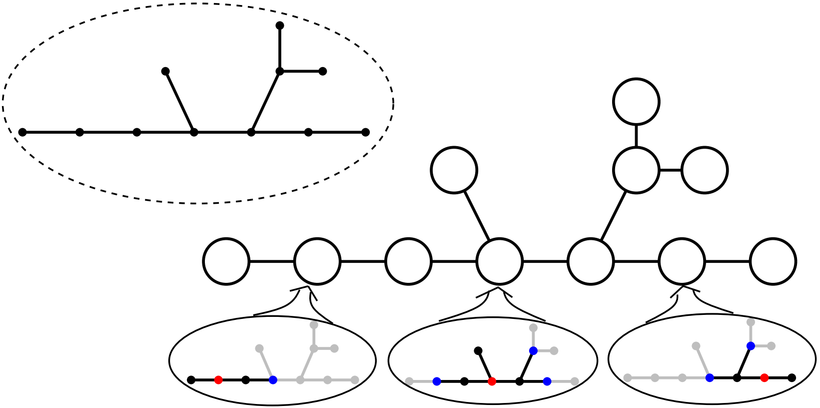
Definition 4.1 (Local Modes). *For each subgraph $\bar{S}_i$, for each fixed labeling of the boundary ∂S_i, the optimal labeling of $\bar{S}_i$ is a local mode.*

Theorem 4.2 (Local-Global). *A labeling is a global mode if and only if it is a local mode within every subgraph $\bar{S}_i$.*



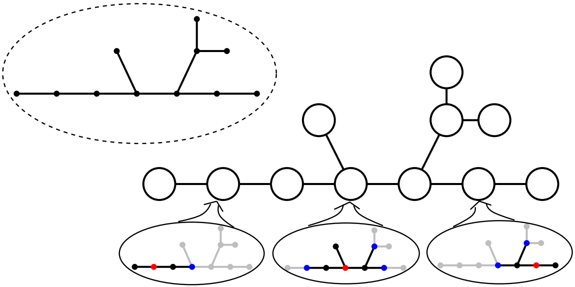
Trees

- Chains $\rightarrow$ trees
- subchains of length $\delta + 2 \rightarrow$ geodesic balls of radius $r = \lfloor \delta/2 \rfloor + 1$
  - Geod dist: $\#$ of edges



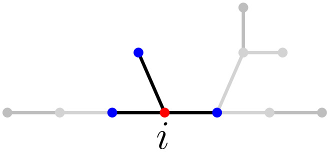
Trees

- ∂B boundary of the ball
- x_B is a local mode iff for any y_B s.t. $y_{\partial B} = x_{\partial B}$, $E(x_B) < E(y_B)$ and $d(x_B, y_B) \leq \delta$.
- Partial-global still holds.
  - Build a new tree (supernodes: geod. balls)
  - M-Best



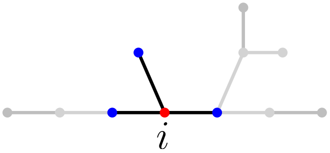
Computing Local Modes

- $\delta = 1$: enumeration;



Computing Local Modes

- $\delta = 1$: enumeration;



- $\delta > 1$:
  - A δ-local mode has to be $(\delta - 1)$-local modes everywhere
  - Reuse local modes of $\delta - 1$
  - Generate a candidate set
  - Check each candidate (DP)

Complexity

Compute top M-modes for all δ's

$$O\left(D^2 dL\delta^2(L+\delta)(L^d+\lambda^d)+D\lambda^2+MD\lambda+MD\log(MD)\right)$$

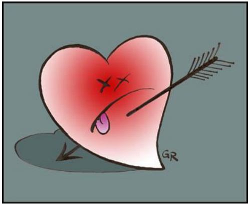
- d tree degree
- λ max $\#$ of local modes for any ball

Complexity

Compute top M-modes for all δ's

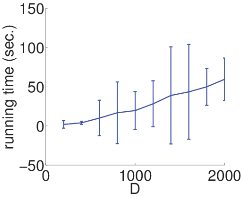
$$O\left(D^2 d L \delta^2 (L+\delta)(L^d+\lambda^d) + D\lambda^2 + MD\lambda + MD\log(MD)\right)$$

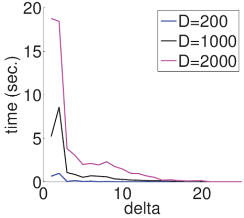
- d tree degree
- λ max # of local modes for any ball
- Exponential



Scalability

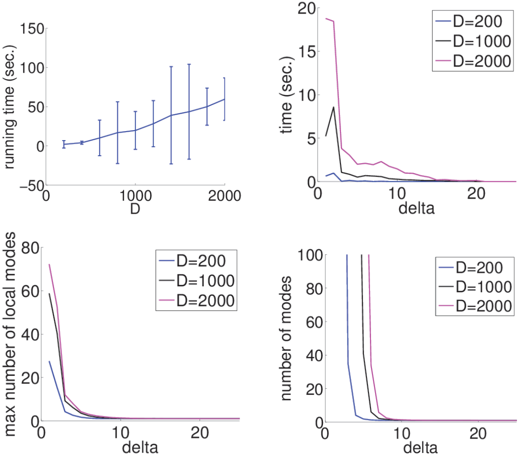
- exponential to d: bounded degree tree
- λ can be large, but drops quickly as δ increases





Scalability

- exponential to d: bounded degree tree
- λ can be large, but drops quickly as δ increases



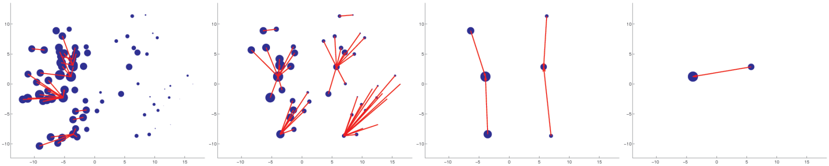
Model Unknown

- Input: samples
- Algorithm:
  - Step 1: estimate a tree distribution [Liu et al. JMLR'11]
  - Step 2: compute modes
- Theoretical guarantee $P(\widehat{\mathcal{M}^\delta} = \mathcal{M}^\delta) \to 1$ as $S \to \infty$

Outline

Visualizing a Distribution

- Microarray gene data of Arabidopsis thaliana plant (39 dim)



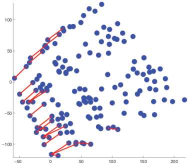
$\delta = 1$ $\delta = 2$ $\delta = 3$ $\delta = 4$

multidimensional scaling

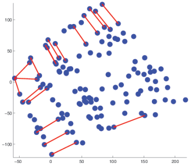
[**CC** *et al.* AISTATS 2013] [**CC** *et al.* NIPS 2014]

ADHD-200 (Attention Deficit Hyperactivity Disorder)

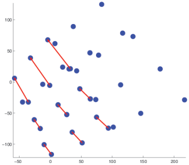
- 776 subjects: 491 healthy, 285 ADHD
- resting state FMRI
- 264 voxels on cortex, discretize (0/1)



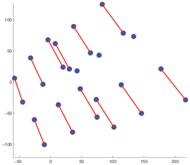
$\delta = 11$



$\delta = 14$



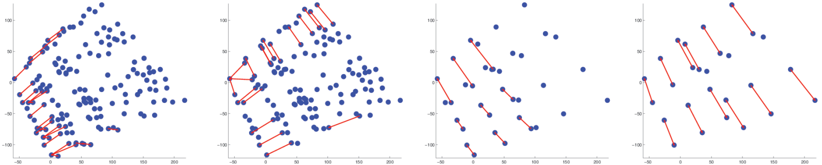
$\delta = 17$



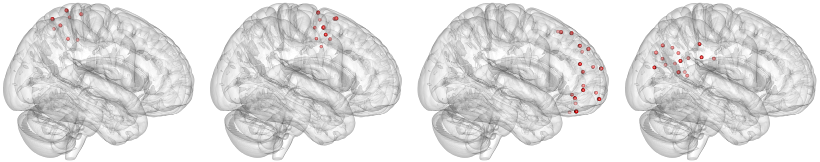
$\delta = 18$

ADHD-200 (Attention Deficit Hyperactivity Disorder)

- 776 subjects: 491 healthy, 285 ADHD
- resting state FMRI
- 264 voxels on cortex, discretize (0/1)



$\delta = 11$ $\delta = 14$ $\delta = 17$ $\delta = 18$

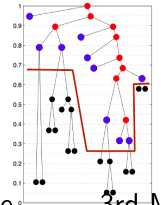


Chains and Trees: Applications

- Video labeling
- Simple Chain:
  - nodes $\rightarrow$ frames
  - labels $\rightarrow$ different gestures

Chen *et al.* AISTATS 13

Application: Multiple Predictions

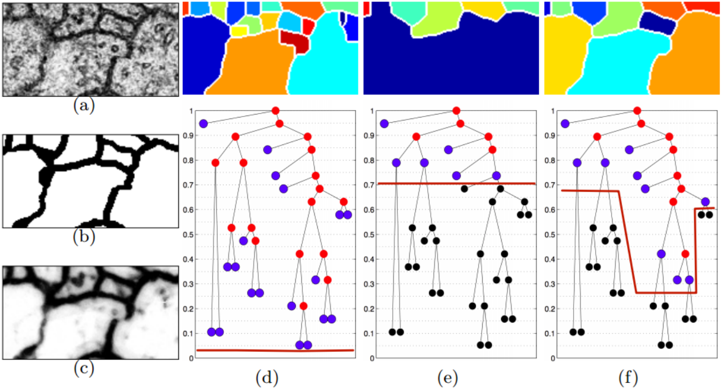


- High Probability; Diversity
- Image Partitioning Task (Berkeley Dataset)

Ground Truth 1st Mode 2nd Mode 3rd Mode

Chains and Trees: Applications

- Neuron image segmentation: hierarchical merging tree
  - nodes → regions of the image
  - labels → whether the region is
    - a subregion of a neuron
    - a neuron region
    - union of several



Mustafa, Chen and Metaxas MICCAI'14, MedIA'15

Outline

Graphical Models

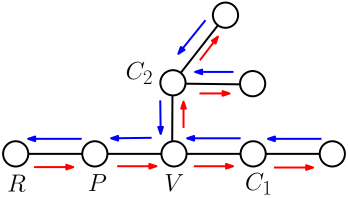
MModes Problem

Algorithms

Applications

Verification and Local Search Algorithm

A Tree: MAP

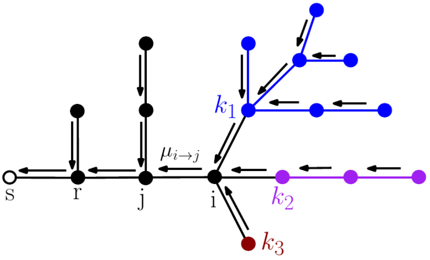


- $MAP = \text{argmin}\, E(y)$
- $M_{v \to p}(y_p = \ell) =$
  $\min_{y_v} \left(\theta_{v,p}(y_v, y_p) + M_{c_1 \to v}(y_v) + M_{c_2 \to v}(y_v)\right)$
- Message Scheduling:
  - Fix a root R
  - Compute messages from leaves to the root (blue)
  - Compute messages from the root to leaves (red)
- $E(MAP) = \min_{y_R} \sum_{u \in Children(R)} M_{u \to R}(y_R)$
- Recover MAP via backtracking

A Tree: Verification

- Given y, is it a δ-mode?
  Compute
  $$\underset{z \in \mathcal{N}_\delta(y) \setminus \{y\}}{\operatorname{argmin}} \; E(z)$$
  - $z = y$, YES, $z \neq y$, NO
- Complexity $O(DdL\delta^2(L+\delta))$

$$\mu_{i \to j}(\ell_i, \tau) = \min_{z_{T_i} : z_i = \ell_i, \rho(z_{T_i}, y) \leq \tau} f(z_{T_i}),$$



A Tree: Verification

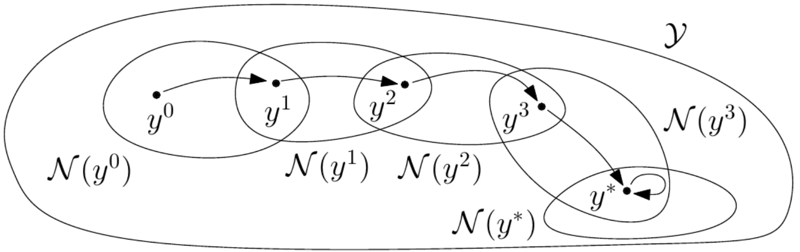
Algorithm 1 `Is-a-Mode`

Require: A tree G, a potential function f, a scale δ and a given labeling y

Ensure: Whether y is a mode

Build a rooted tree as in Figure 2
V = the post-order traversal of the the rooted tree
for $v \in V$, $v \neq s$ **do**
    u = the parent of v
    for all $\ell_u \in \mathcal{L}$, $\tau \in [0, \delta]$ **do**
        Compute $\mu_{u \to v}(\ell_u, \tau)$
    end for
end for
$f(y^*) = \min_{\ell_r} \mu_{r \to s}(\ell_r, \delta)$
Recover y^* through backtracking
if $y^* = y$ **then**
    return TRUE
else
    return FALSE
end if

Local Search: Graphcut for Each Step



- $\mathcal{N}_t : \mathcal{Y} \to 2^{\mathcal{Y}}$, neighborhood system
- Optimization with respect to $\mathcal{N}_t(y)$ must be tractable:

$$y^{t+1} = \operatorname*{argmax}_{y \in \mathcal{N}_t(y^t)} g(x, y)$$

Pic from Nowozin and Lampert

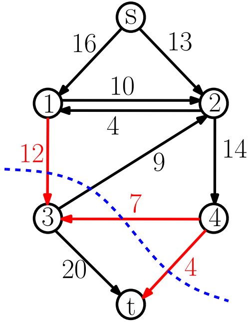
Graphcut: Minimum Cut

The minimal cut problem

- A directed graph,
  - positive weights on all edges, $w_{(u,v)} > 0$
  - one source (s) and a sink (t)
- A cut: a partion of vertices into two parts, $S \dot{\cup} T = \mathcal{V}$, such that $s \in S$ and $t \in T$.
- The minimum cut: a cut with the minimal total weight,

$$\sum_{(u,v)\in\mathcal{E}, u\in S, v\in T} w_{(s,t)}$$

- Max-flow min-cut theorem
- $O(|\mathcal{V}||\mathcal{E}|)$ (Orlin STOC'13)



Graphcut: min-cut for exact MAP computation

- Special case, binary labels (1,2)
  - $\forall u \in \mathcal{V}$, $\mu_u(1) > 0$, $\mu_u(2) > 0$
  - $\forall (u, v) \in \mathcal{E}$, $\theta_{uv}(1, 1) = \theta_{uv}(2, 2) = 0$, $\theta_{uv}(1, 2) = \theta_{uv}(2, 1) = w_{uv} > 0$
- Construct directed graph, $\mathcal{V} \cup \{s, t\}$
  - An edge from s to each original node (weight $\mu_u(2)$)
  - An edge from each orginal node to t (weight $\mu_u(1)$)
  - A pair of edges for each original edge (both with weight w_{uv})
- Min-cut = MAP: $u \in S$ iff $y_u = 1$, cut weight = $E(y)$