Simultaneous recovery of the consensus and structure of permutations

Marina Meilă
University of Washington
with Chris Meek, Microsoft Research

UW Theory Seminar 2/2/16

The "Sushi preference" data

$N = 5000$ people ranked $n = 12$ types of sushi

sake |ebi |ika |uni |tamago |kappa-maki |tekka-maki |anago |toro |maguro
ebi |kappa-maki |tamago |ika |toro |maguro |tekka-maki |anago |sake |uni
toro |ebi |maguro |ika |tekka-maki |uni |sake |anago |kappa-maki |tamago
tekka-maki |tamago |sake |ebi |ika |kappa-maki |maguro |toro |uni |anago
tamago |maguro |kappa-maki |ebi |sake |anago |uni |tekka-maki |toro |ika
uni |toro |ebi |anago |maguro |tekka-maki |ika |sake |kappa-maki |tamago
maguro |ika |toro |tekka-maki |ebi |uni |sake |tamago |anago |kappa-maki

Consensus Ranking Problem

Given a set of rankings $\{\pi_1, \pi_2, \ldots \pi_N\} \subset \mathbb{S}_n$ find the consensus ranking π_0 such that

$$\pi_0 \; = \; \operatorname*{argmin}_{\mathbb{S}_n} \sum_{i=1}^{N} d(\pi_i, \pi_0)$$

for d = inversion distance / Kendall τ-distance / "bubble sort" distance

The "Sushi preference" data

$N = 5000$ people ranked $n = 12$ types of sushi

sake |ebi |ika |uni |tamago |kappa-maki |tekka-maki |anago |toro |maguro
ebi |kappa-maki |tamago |ika |toro |maguro |tekka-maki |anago |sake |uni
toro |ebi |maguro |ika |tekka-maki |uni |sake |anago |kappa-maki |tamago
tekka-maki |tamago |sake |ebi |ika |kappa-maki |maguro |toro |uni |anago
tamago |maguro |kappa-maki |ebi |sake |anago |uni |tekka-maki |toro |ika
uni |toro |ebi |anago |maguro |tekka-maki |ika |sake |kappa-maki |tamago
maguro |ika |toro |tekka-maki |ebi |uni |sake |tamago |anago |kappa-maki

Consensus Ranking Problem

Given a set of rankings $\{\pi_1, \pi_2, \ldots \pi_N\} \subset \mathbb{S}_n$ find the consensus ranking π_0 such that

$$\pi_0 \; = \; \operatorname*{argmin}_{\mathbb{S}_n} \sum_{i=1}^{N} d(\pi_i, \pi_0)$$

for d = inversion distance / Kendall τ-distance / "bubble sort" distance

This problem is NP-hard []

Related work

Consensus Ranking/Single parameter/Mallows model

[Cohen,S,Singer 99] CSS ALGORITHM = greedy search on Q
improved by extracting strongly connected components

[Ailon,Newman,Charikar 05] Randomized algorithm guaranteed 11/7 factor approximation (ANC)

[Mohri, Ailon 08] linear program

[Mathieu, Schudy 07] $(1+\epsilon)$ approximation, time $\mathcal{O}(n^6/\epsilon + 2^{2^{O(1/\epsilon)}})$

[Davenport,Kalagnanan 03] Heuristics based on edge-disjoint cycles used by our B&B implementation

[Conitzer,D,K 05] Exact algorithm based on integer programming, better bounds for edge disjoint cycles (DK)

[Betzler,Brandt, 10] Exact problem reductions

[Awasthi,Blum,Sheffet,Vijayaraghavan 14]

- Most of this work based on the MinFAS view

$$Q_{ij} > .5 \quad \Leftrightarrow \quad i\bullet \xrightarrow{Q_{ij}-.5} \bullet j$$

Prune graph to a DAG removing minimum weight

Extensions and applications to social choice

Social choice

- Inferring rakings under partial and aggregated information [ShahJabatula08], [JabatulaFariasShah10]
- Vote elicitation under probabilistic models of choice [LuBoutillier11]
- Voting rules viewed as Maximum Likelihood [ConitzerSandholm08]
- Algorithms guaranteed to retrive certain "winners" [LinAgarwal14]

"Noisy sorting"

- Using Hodge decompositions and L1, L2 distances [JiangLimYaoYe11]
- Noisy comparison [BravermanMossel08]

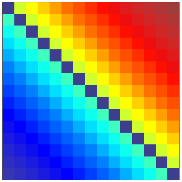
ML Estimation/Multiple parameters/GM model

l[VlignerVerducci 86] $\vec{\theta}$ estimation; heuristic for π_0

FV ALGORITHM/BORDA RULE

1. Compute $\bar{q}_j, j = 1 : n$ column sums of Q
2. Sort $(\bar{q}_j)_{j=1}^n$ in increasing order; π_0 is sorting permutation

- $\bar{q}_j$ are Borda counts
- FV is consistent for infinite N



Generalizing consensus ranking

- Not all inversions are equally important

Sushi preferences for uni have no consensus

sake |ebi |ika |uni |tamago |kappa-maki |tekka-maki |anago |toro |maguro
ebi |kappa-maki |tamago |ika |toro |maguro |tekka-maki |anago |sake |uni
toro |ebi |maguro |ika |tekka-maki |uni |sake |anago |kappa-maki |tamago
tekka-maki |tamago |sake |ebi |ika |kappa-maki |maguro |toro |uni |anago
tamago |maguro |kappa-maki |ebi |sake |anago |uni |tekka-maki |toro |ika
uni |toro |ebi |anago |maguro |tekka-maki |ika |sake |kappa-maki |tamago
maguro |ika |toro |tekka-maki |ebi |uni |sake |tamago |anago |kappa-maki

Generalizing consensus ranking

- Not all inversions are equally important

. . . but there is consensus for maguro (tuna) and tekka-maki (tuna roll)

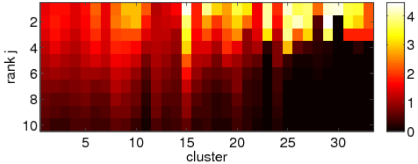
sake |ebi |ika |uni |tamago |kappa-maki |tekka-maki |anago |toro |maguro
ebi |kappa-maki |tamago |ika |toro |maguro |tekka-maki |anago |sake |uni
toro |ebi |maguro |ika |tekka-maki |uni |sake |anago |kappa-maki |tamago
tekka-maki |tamago |sake |ebi |ika |kappa-maki |maguro |toro |uni |anago
tamago |maguro |kappa-maki |ebi |sake |anago |uni |tekka-maki |toro |ika
uni |toro |ebi |anago |maguro |tekka-maki |ika |sake |kappa-maki |tamago
maguro |ika |toro |tekka-maki |ebi |uni |sake |tamago |anago |kappa-maki

Generalizing consensus ranking

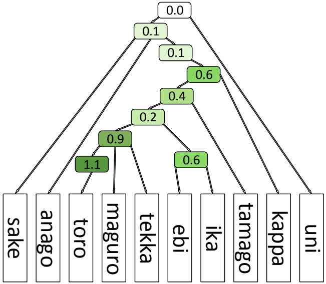
- Not all inversions are equally important
  introduce importance/weight parameters $\vec{\theta}$

Irish College Admissions data

Parameters of top 10 ranks in the 33 largest clusters found



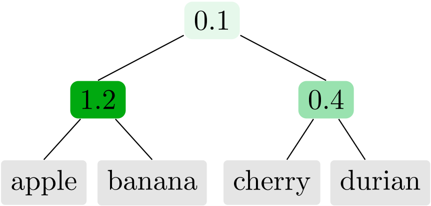
Combinatorial structure present



- described by a tree

Recursive Inversion Models (RIM)

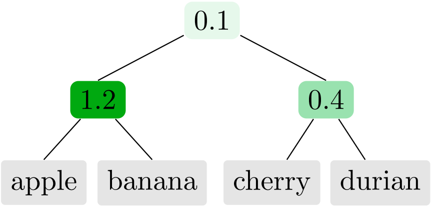
[Meek, M 14]



τ = tree structure
$\pi_0(\tau)$ = induced central ranking
$\theta_{1:n-1}$ = parameters at nodes

Recursive Inversion Models (RIM)

[Meek, M 14]

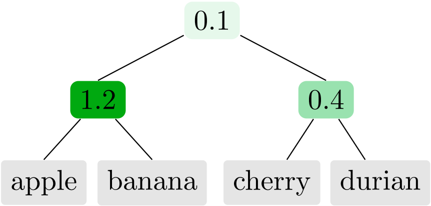


τ = tree structure
$\pi_0(\tau)$ = induced central ranking
$\theta_{1:n-1}$ = parameters at nodes
Inversions are penalized by θ_i parameters
Example: $\vec{\theta} = (0.1, 1.2, 0.4)$

$$\text{Cost}(a|b|c|d) = 0$$

Recursive Inversion Models (RIM)

[Meek, M 14]

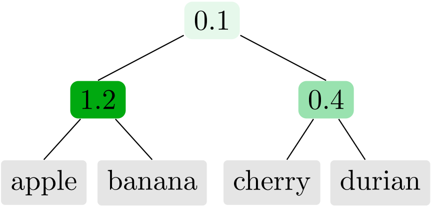


τ = tree structure
$\pi_0(\tau)$ = induced central ranking
$\theta_{1:n-1}$ = parameters at nodes
Inversions are penalized by θ_i parameters
Example: $\vec{\theta} = (0.1, 1.2, 0.4)$

$$\text{Cost}(a|b|c|d) = 0$$
$$\text{Cost}(b|a|c|d) = 1.2$$

Recursive Inversion Models (RIM)

[Meek, M 14]

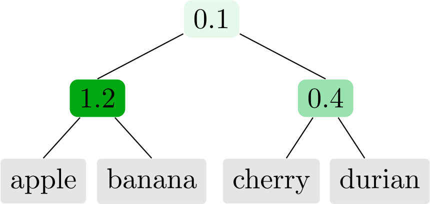


τ = tree structure
$\pi_0(\tau)$ = induced central ranking
$\theta_{1:n-1}$ = parameters at nodes
Inversions are penalized by θ_i parameters
Example: $\vec{\theta} = (0.1, 1.2, 0.4)$

$$\begin{aligned} \text{Cost}(a|b|c|d) &= 0 \\ \text{Cost}(b|a|c|d) &= 1.2 \\ \text{Cost}(c|b|a|d) &= 1.2 + 2 \times 0.1 \end{aligned}$$

Recursive Inversion Models (RIM)

[Meek, M 14]

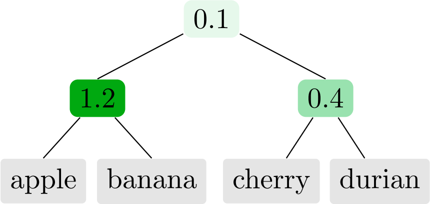


τ = tree structure
$\pi_0(\tau)$ = induced central ranking
$\theta_{1:n-1}$ = parameters at nodes
Inversions are penalized by θ_i parameters
Example: $\vec{\theta} = (0.1, 1.2, 0.4)$

$$\begin{aligned} \text{Cost}(a|b|c|d) &= 0 \\ \text{Cost}(b|a|c|d) &= 1.2 \\ \text{Cost}(c|b|a|d) &= 1.2 + 2 \times 0.1 \end{aligned}$$

Recursive Inversion Models (RIM)

[Meek, M 14]



τ = tree structure
$\pi_0(\tau)$ = induced central ranking
$\theta_{1:n-1}$ = parameters at nodes
Inversions are penalized by θ_i parameters
Example: $\vec{\theta} = (0.1, 1.2, 0.4)$

$$\begin{aligned} \text{Cost}(a|b|c|d) &= 0 \\ \text{Cost}(b|a|c|d) &= 1.2 \\ \text{Cost}(c|b|a|d) &= 1.2 + 2 \times 0.1 \end{aligned}$$

$$\begin{aligned} P(a|b|c|d) &\propto e^{0} \\ P(b|a|c|d) &\propto e^{-1.2} \\ P(c|b|a|d) &\propto e^{-1.2-2\times 0.1} \end{aligned}$$

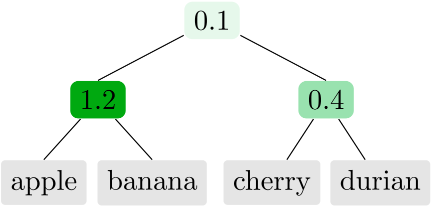
RIM distribution $P_{\tau,\vec{\theta}}$

Let v_i = number of inversions of π at node i

$$P_{\tau,\vec{\theta}}(\pi) \propto \prod_{i \in nodes} \exp\left(-\theta_i v_i\right)$$

Recursive Inversion Models (RIM)

[Meek, M 14]



τ = tree structure
$\pi_0(\tau)$ = induced central ranking
$\theta_{1:n-1}$ = parameters at nodes
Inversions are penalized by θ_i parameters
Example: $\vec{\theta} = (0.1, 1.2, 0.4)$

$$\begin{aligned} \text{Cost}(a|b|c|d) &= 0 \\ \text{Cost}(b|a|c|d) &= 1.2 \\ \text{Cost}(c|b|a|d) &= 1.2 + 2 \times 0.1 \end{aligned}$$

$$\begin{aligned} P(a|b|c|d) &\propto e^{0} \\ P(b|a|c|d) &\propto e^{-1.2} \\ P(c|b|a|d) &\propto e^{-1.2-2\times 0.1} \end{aligned}$$

RIM distribution $P_{\tau,\vec{\theta}}$

Let v_i = number of inversions of π at node i

$$P_{\tau,\vec{\theta}}(\pi) \propto \prod_{i \in nodes} \exp\left(-\theta_i v_i\right)$$

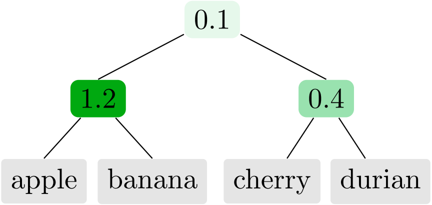
Normalization constant

$$Z(\tau, \theta) = \prod_{i \in nodes} G(L_i, R_i, \exp(-\theta_i))$$

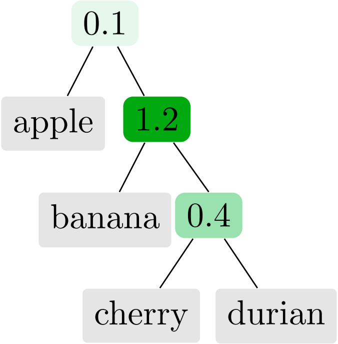
$$\text{with} \;\; G(L, R, q) = \frac{(q)_{L+R}}{(q)_L (q)_L}, \;\; (q)_n = \prod_{i=1}^{n} (1 - q^i).$$

Structure τ known as Riffle Independence model [Huang,Guestrin 12]

The RIM is a general flexible model



- any tree structure
- any parameters (but $\theta_j \geq 0$ suffices)
- includes the Mallows and Generalized Mallows models



Max Likelihood Estimation for RIM

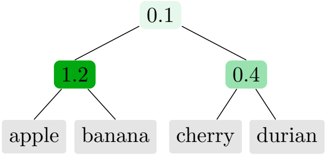
[M,Meek 14]

- Problem Given permutations $\pi_1, \ldots \pi_N$, infer τ, θ

Max Likelihood Estimation for RIM

[M,Meek 14]

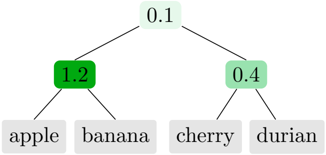
- Problem Given permutations $\pi_1, \ldots \pi_N$, infer τ, θ
- Identifiability of θ
  - reorder to obtain cannonical representation, with $\theta_i \geq 0$ for all $i \in nodes$
  - given τ, θ_i can be estimated by convex univariate minimization



Max Likelihood Estimation for RIM

[M,Meek 14]

- Problem Given permutations $\pi_1, \ldots \pi_N$, infer τ, θ
- Identifiability of θ
  - reorder to obtain cannonical representation, with $\theta_i \geq 0$ for all $i \in nodes$
  - given τ, θ_i can be estimated by convex univariate minimization



- Identifiability of τ

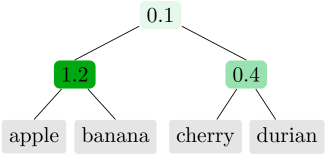
Theorem[M, Meek 14] A model τ, θ is identifiable iff

1. $\theta_i > 0$ for all $i \in nodes$
2. $\theta_i \neq \theta_{pa(i)}$ for all $i \in nodes$ ($pa(i)$ is the parent of node i in τ)

Max Likelihood Estimation for RIM

[M,Meek 14]

- Problem Given permutations $\pi_1, \ldots \pi_N$, infer τ, θ
- Identifiability of θ
  - reorder to obtain cannonical representation, with $\theta_i \geq 0$ for all $i \in nodes$
  - given τ, θ_i can be estimated by convex univariate minimization



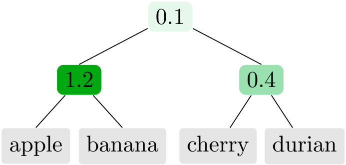
- Identifiability of τ

Theorem[M, Meek 14] A model τ, θ is identifiable iff

1. $\theta_i > 0$ for all $i \in nodes$
2. $\theta_i \neq \theta_{pa(i)}$ for all $i \in nodes$ ($pa(i)$ is the parent of node i in τ)

- Hardness of τ estimation
  - Estimating π_0 is NP-hard [Duchi, Mackey, Jordan 13]
  - Estimating τ structure given π_0 is tractable

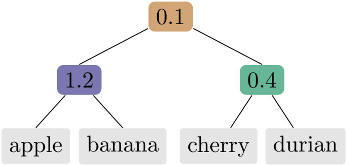
Sufficient statistics



$Q(d|a|b|c) =$

a	b	c	d	
–	1	0	0	a
0	–	1	0	b
0	0	–	0	c
1	1	1	–	d

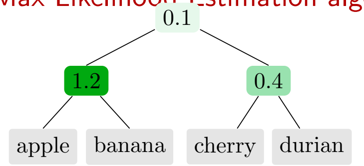
Sufficient statistics



$$Q(d|a|b|c) = \begin{array}{|cccc|c} a & b & c & d & \\ \hline - & 1 & 1 & 0 & a \\ 0 & - & 1 & 0 & b \\ 0 & 0 & - & 0 & c \\ 1 & 1 & 1 & - & d \\ \hline \end{array}$$

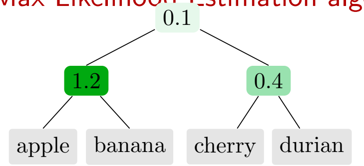
$$\text{Cost}(d|a|b|c) = 0.1 \times 2 + 1.2 \times 0 + 0.4 \times 1$$

Max Likelihood Estimation algorithm(s)



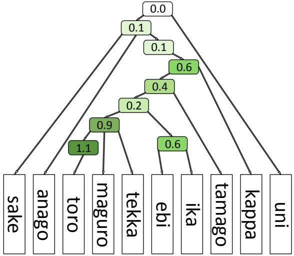
- Estimating τ given π_0 is tractable

Max Likelihood Estimation algorithm(s)



- Estimating τ given π_0 is tractable
  - by Dynamic Programming (DP) algorithm, similar to Matrix Chain Multiplication, Inside(-Outside) algorithm $\mathcal{O}(n^4)$
  - contains θ_j estimation at each DP "partial solution"
- Estimating π_0: Stochastic local search over π_0 space, similar to Simulated Annealing
  1. Sample ${\pi_0}^{new}$ from proposal distribution current $P_{\tau,\theta}$
  2. Given ${\pi_0}^{new}$, find τ^{opt}, θ^{opt} by Dynamic Programming
  3. Bring to cannonical form $\Rightarrow \tau^{new}, \theta^{new} \succeq 0$
  4. Compute *log-likelihood score*, accept/reject like in Metropolis-Hastings, return to step 1

Experiments - Sushi preferences data



Data
$N = 5000$ permutations of $n = 10$ items
Compared with:

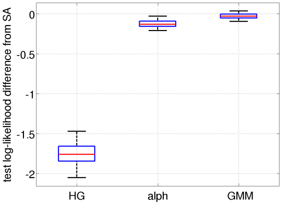
alph π_0 fixed, $\tau, \theta|\pi_0$ optimize

GM fixed τ, optimize π_0, θ

HG fixed τ from [Huang,Guestrin,12], optimize θ

SA Simulated Annealing

Test set log-likelihood w.r.t SA

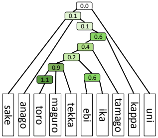


$N_{test} = 300$, $N_{train} = 4700$, 30 replicates

Partial rankings

"Sushi preference" data $n = 12$
types of sushi

"My top 3 preferences are ika, maguro, tekka, in this order"
"I like uni least of all"
"I prefer fish to non-fish"
. . .



Three good things about the RIM

- RIM is a general model (includes Mallows, generalized Mallows)
- likelihood $P(\pi|\tau(\vec{\theta}))$ factors according to tree (and partition function Z tractable)
- RIM has sufficient statistics

Partial rankings

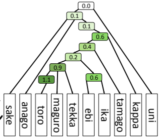
"Sushi preference" data $n = 12$

types of sushi

ika|maguro|tekka|{all other types}

{all but ebi}|ebi

$\underbrace{\{\text{sake,anago,}\ldots\}}_{E_1} \,|\, \underbrace{\{\text{tamago,ika,}\ldots\}}_{E_2}$



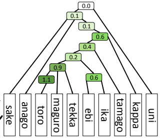
Partial rankings

"Sushi preference" data $n = 12$ types of sushi

ika|maguro|tekka|{all other types}

{all but ebi}|ebi

$\underbrace{\{\text{sake,anago,}\ldots\}}_{E_1} \,|\, \underbrace{\{\text{tamago,ika,}\ldots\}}_{E_2}$



Partial ranking σ [Huang & al, 10]

$\sigma = (E_1|E_2|\ldots|E_K)$ with

- $E_1 \cup E_2 \cup \ldots E_K =$ set of items
- shape $(n_1, \ldots n_K)$, $n_k = |E_k|$, $\sum n_k = n$

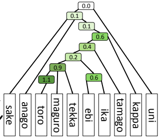
Partial rankings

"Sushi preference" data $n = 12$ types of sushi

ika|maguro|tekka|{all other types}

{all but ebi}|ebi

$\underbrace{\{\text{sake,anago,}\ldots\}}_{E_1} \mid \underbrace{\{\text{tamago,ika,}\ldots\}}_{E_2}$



Partial ranking σ [Huang & al, 10]

$\sigma = (E_1|E_2|\ldots|E_K)$ with

- $E_1 \cup E_2 \cup \ldots E_K$ = set of items
- shape $(n_1, \ldots n_K)$, $n_k = |E_k|$, $\sum n_k = n$

Three good things about the RIM

- RIM is a general model (includes Mallows, generalized Mallows)
- likelihood $P(\pi|\tau(\vec{\theta}))$ factors according to tree **?** **YES** [Huang et al, 10]
- RIM has sufficient statistics **?** **NO**

Inferences with partial rankings in the RIM. Are they tractable?

The meaning of "tractable"

- Estimation of π_0 for RIM is intractable in the worst case
- We define tractable as $\mathcal{O}(N\,poly(n))\times$ time (memory) for complete data

Inferences with partial rankings in the RIM. Are they tractable?

The meaning of "tractable"

- Estimation of π_0 for RIM is intractable in the worst case
- We define tractable as $\mathcal{O}(N\, poly(n))\times$ time (memory) for complete data

Main technical difficulty

- marginal probability of a partial ranking σ

$$P(\sigma|\tau(\vec{\theta})) = \sum_{\pi\sim\sigma} P(\pi|\tau(\vec{\theta}))$$

where linear extension $\{\pi \sim \sigma\}$ of σ can have exponential size

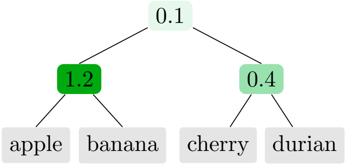
Contributions

1. for marginal probability $P(\sigma|\tau(\vec{\theta}))$
   - exact formula and polynomial algorithm
   - proved algorithm no more than $2Nn$ more costly than for complete permutations (and sometimes much faster)
2. for pairwise marginals $E[Q_{ab}] = Pr[a \text{ precedes } b \,|\, \sigma, \tau(\vec{\theta})]$
   - exact recursive (polynomial) algorithm
   - proved algorithm no more costly than for complete permutations
3. for parameter $\vec{\theta}$ estimation (Maximum Likelihood)
   - convex univariate minimization algorithm for each θi
   - proved algorithm is $\mathcal{O}(Nn)$ more costly than for complete permutations
4. for structure search (Maximum Likelihood)

   previous work
   - complete data: local (simulated annealing) search algorithm with exact, tractable steps [Meek M 14]
   - partial rankings: EM algorithm with approximate (or exponential) E step [Huang & al 10]

   our contributions
   - new "E step" based on completing the pairwise marginals $E[Q_{ab}]$
   - algorithms above can use the completed pairwise marginals as if they were complete data

Computing the marginal probability $P(\sigma|\tau, \vec{\theta})$



$$P(a|b|c|d) \propto e^{0}$$
$$P(b|a|c|d) \propto e^{-1.2}$$
$$P(c|b|a|d) \propto e^{-1.2-2\times 0.1}$$

RIM probability for complete data $P(\pi|\tau, \vec{\theta})$

(with v_i = number of inversions of π_0 at node i)

$$P_{\tau,\vec{\theta}}(\pi) = \prod_{i \in nodes} \frac{e^{-\theta_i v_i}}{G_{L_i,R_i}(\exp(-\theta_i))}$$

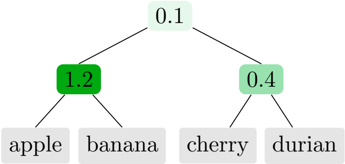
$$\text{with } G_{L,R}(q) = \frac{(q)_{L+R}}{(q)_L (q)_R}, \quad (q)_n = \prod_{i=1}^{n} (1 - q^i).$$

RIM probability for partial ranking σ

[M, Meek in prep]

$$P_{\tau,\vec{\theta}}(\sigma) = \prod_{i \in nodes} (\text{factor at node } i)$$

Marginal $P(\pi|\tau,\vec{\theta})$ for partial ranking σ



Sufficient to consider root node

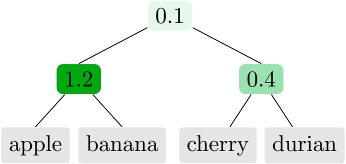
Complete ranking $\pi = (c|a|b|d)$

$$\text{factor} = \frac{e^{-2\theta}}{G_{2,2}(e^{-\theta})}$$

Partial ranking $\sigma = (c|\{a, b, d\})$

$$\text{factor} = \frac{e^{-2\theta} G_{0,1}(e^{-\theta}) G_{2,1}(e^{-\theta})}{G_{2,2}(e^{-\theta})}$$

Marginal $P(\pi|\tau,\vec{\theta})$ for partial ranking σ



Sufficient to consider root node

Complete ranking $\pi = (c|a|b|d)$

$$\text{factor} = \frac{e^{-2\theta}}{G_{2,2}(e^{-\theta})}$$

Partial ranking $\sigma = (c|\{a,b,d\})$

$$\text{factor} = \frac{e^{-2\theta} G_{0,1}(e^{-\theta}) G_{2,1}(e^{-\theta})}{G_{2,2}(e^{-\theta})}$$

In general, at some internal node where

- set $\mathcal{L}$ is merged with set $\mathcal{R}$
- partial ranking σ restricted to $\mathcal{L} \cup \mathcal{R}$ is $E_1|E_2|\ldots|E_K$ with $E_k = L_k \cup R_k$, $L_k \subseteq \mathcal{L}$, $r_k \subseteq \mathcal{R}$
- factor of $P(\sigma|\tau(\vec{\theta}))$ at this node is

$$g(l_{1:K}, r_{1:K}, \theta) = \frac{e^{-\theta v} G_{l_1,r_1}(e^{-\theta}) G_{l_2,r_2}(e^{-\theta}) \ldots G_{l_K,r_K}(e^{-\theta})}{G_{|\mathcal{L}|,|\mathcal{R}|}(e^{-\theta})}$$

where $v = \#$ inversions in σ at node $\leq \#$ inversions in $\pi \sim \sigma$

Marginal $P(\pi|\tau, \vec{\theta})$ – how much extra computation?

How many additional factors?

Rem 1 $G_{0,r} = G_{l,0} = 1$

Marginal $P(\pi|\tau, \vec{\theta})$ – how much extra computation?

How many additional factors?

Rem 1 $G_{0,r} = G_{l,0} = 1$

Rem 2 at each node, at least one of L_k, R_k decreases (and their initial sum is n)

- Hence, no more than $n - 1$ extra factors (but sometimes much fewer)

Marginal $P(\pi|\tau,\vec{\theta})$ – how much extra computation?

How many additional factors?

Rem 1 $G_{0,r} = G_{l,0} = 1$

Rem 2 at each node, at least one of L_k, R_k decreases (and their initial sum is n)

- Hence, no more than $n-1$ extra factors (but sometimes much fewer)
- Example top-t rankings $\sigma = (\mathit{ika}|\mathit{maguro}|\mathit{sake}|\{\text{everything else}\})$ $P(\sigma|\tau,\vec{\theta})$ has at most $t-1$ non-trivial factors

Marginal $P(\pi|\tau,\vec{\theta})$ – how much extra computation?

How many additional factors?

Rem 1 $G_{0,r} = G_{l,0} = 1$

Rem 2 at each node, at least one of L_k, R_k decreases (and their initial sum is n)

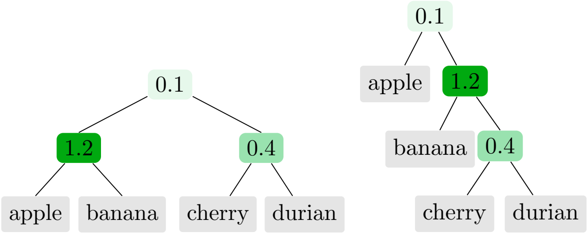
- Hence, no more than $n-1$ extra factors (but sometimes much fewer)
- Example top-t rankings $\sigma = (ika|maguro|sake|\{\text{everything else}\})$ $P(\sigma|\tau,\vec{\theta})$ has at most $t-1$ non-trivial factors

How much additional computation?

- $G_{L,R}$ is computed recursively over $l = 0,\ldots L$, $r = 1,\ldots R$
- Hence, all $G_{l,r}(\theta)$ in numerator are cached while computing the denominator

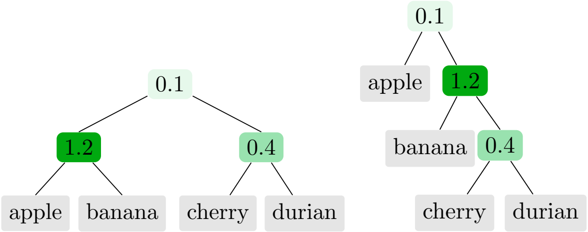
- Overhead for whole sample of size N is no more than nN lookups+multiplications
- For comparison, for a complete whole sample
  - computation of sufficient statistics is $\mathcal{O}(n^2 N)$
  - computation of Z given $\vec{\theta}$ is $\mathcal{O}(n^2 \log n)$

Independence properties



- define $Q_{ab} = 1$ iff a precedes b
- $Q_{ab} \perp Q_{cd}$ whenever $\text{path}(a, b) \cap \text{path}(c, d) = \emptyset$

Independence properties



- define $Q_{ab} = 1$ iff a precedes b
- $Q_{ab} \perp Q_{cd}$ whenever $\text{path}(a, b) \cap \text{path}(c, d) = \emptyset$
- Indepence checking can reveal the "branching structure" (but not π_0)
- In progress: combine independence tests with local search to estimate τ

Conclusion: No need to compromise!

Goals of inference in models on permutations

- Flexible w.r.t observation model (i.e. input data)
  - partial rankings, pairwise observations
- Flexible w.r.t generative model
  - RIMs are a class of flexible, identifyable, intepretable models
- Exact and tractable algorithms, closed form expression