3: RANK AGGREGATION

Ravi Kumar
Yahoo! Research
Sunnyvale, CA
ravikumar@yahoo-inc.com

Outline of lecture

- Metasearch problem and rank aggregation
- Voting and social choice
- Kemeny-optimal/-approximate aggregation
- Simple voting algorithms
- Median rank aggregation and implications
- Improved algorithms
- Heuristics and results
- Other approaches to metasearch
- Distance metrics for IR applications

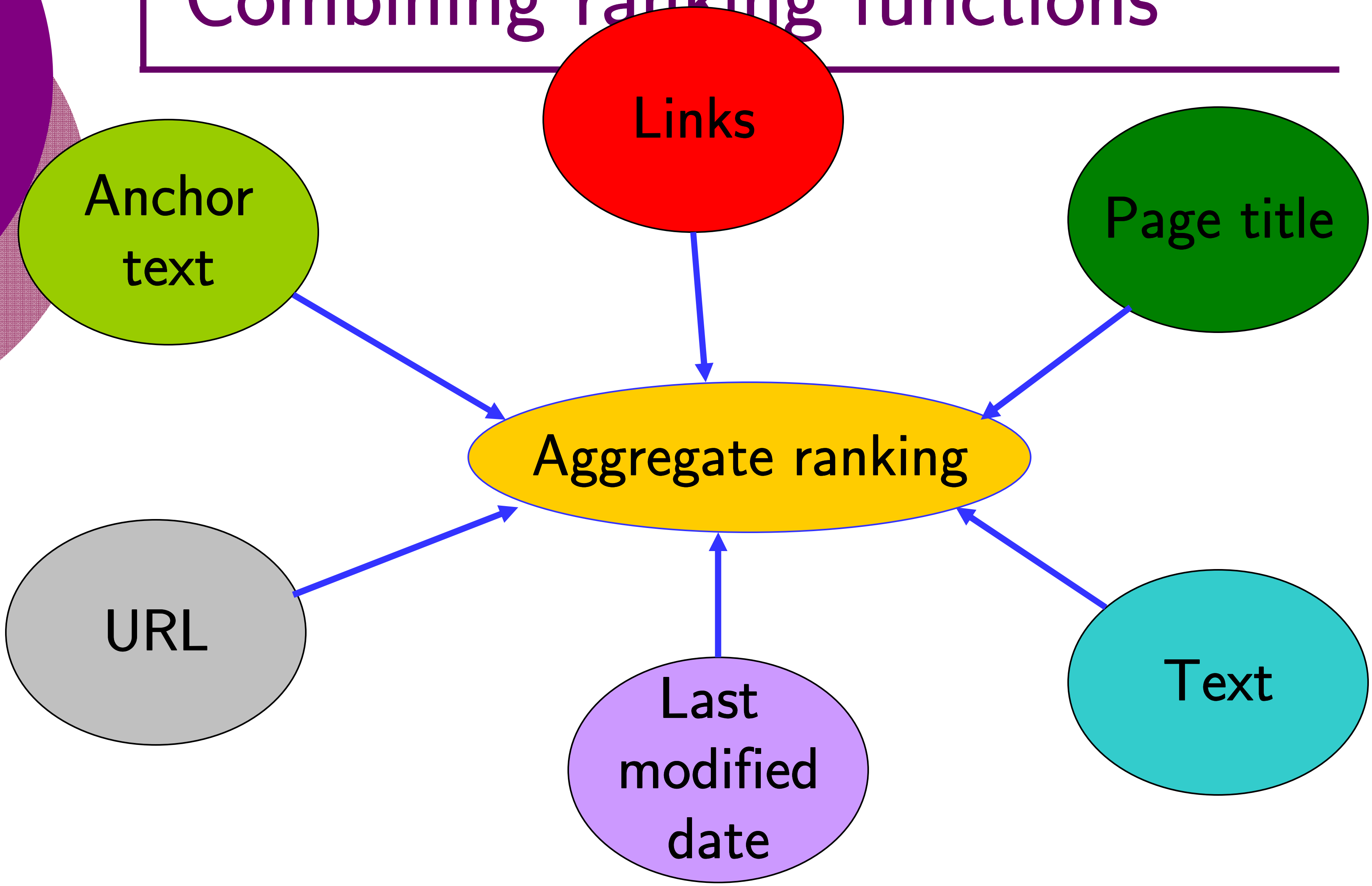
Metasearch

For a given query, combine the results from different search engines

Why metasearch?

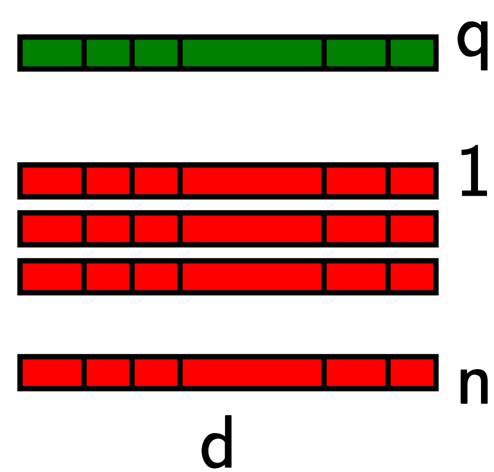
- Coverage: Search engines may not overlap much
- Consensus ranking: Get the best out of several ranking heuristics
- Spam resistance: Hard to fool many search engines
- Query robustness: Work for both broad-topic and specific queries
- Feedback: Reflects the effectiveness of a particular search engine

Combining ranking functions



Similarity search in databases

Given collection of n database elements (each is a d-tuple of attributes) and given at run-time a query element q (another d-tuple of attributes) find the database element that best matches q



Each of the d attributes is a voter

Database elements = candidates

Each voter ranks all candidates

Database elements ranked by voter i, based on similarity to the query q in attribute i

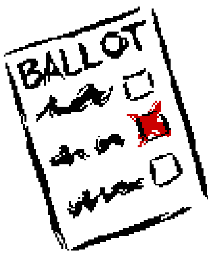
Find top winners of this election by aggregation

Basic theme: Rank aggregation

Input: n candidates and k voters

Preferential voting: Each voter gives a (partial) list of the candidates in order of preference

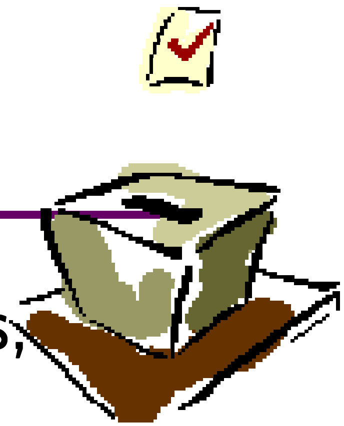
1	3	...	10
3	19	...	17
7	n	...	1
	...		
n	10	...	



Goal: Produce a good consensus ordering of all n candidates

Deja vu: Voting/elections

Voting



- Political decision making, jury decisions, pooling expert opinions, ...
- More than balance subjective opinions
  Seek the truth
  Find the “best” candidate, second “best”, ...
- What is “best”?
- Majority opinion represents (objectively) best?

Voting in CS: Some scenarios

- Meta-search
- Aggregating ranking functions in search engines
- Comparing search engine quality
- Spam reduction
- Nearest-neighbor and similarity search
- Multi-criteria selection (eg, travel, restaurant)
- Word association techniques (AND queries)

CS vs SC

- Small number of voters
- Large number of candidates
- Algorithmic efficiency
- Input could be partial lists/top k lists
- Output might have to be a ranking

Desiderata (CS)

- Simple algorithm
- Fast algorithm (near-linear time)
- Provable quality of solution
- If approximation, factor should be independent of number of candidates/voters

Borda's proposal (1770)

Jean-Charles Borda

Election by order of merit

First place is worth 1 point,
second place is worth 2 points
...

Candidate's score = Sum of points

Borda winner: Lowest scoring candidate

Eg, MVP in MLB

Condorcet's proposal (1785)

Marie J. A. N. Caritat, Marquis de Condorcet

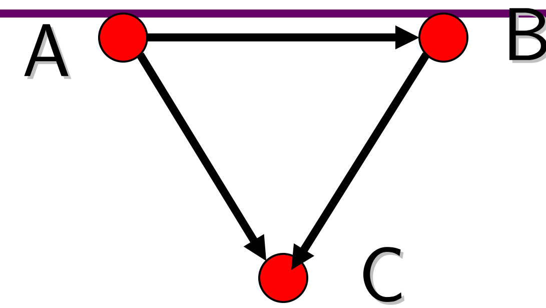
Partition candidates into A, B

If for every a ∈ A and b ∈ B, majority ranks a ahead of b then aggregation must place all elements in A ahead of all elements in B

Condorcet winner: A candidate who defeats every other candidate in pairwise majority-rule election

Condorcet ≠ Borda

(6)	(4)
A	B
B	C
C	A

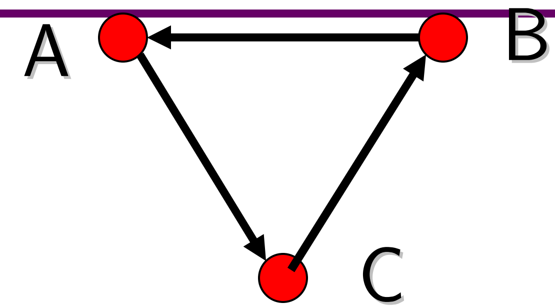


Borda scores: A $(1*6 + 3*4 = 18)$,
B $(2*6 + 1*4 = 16)$, C $= (3*6 + 2*4 = 26)$
B is the Borda winner

Condorcet criterion: A beat both B and C in pair-wise majority
A is the Condorcet winner

Condorcet paradox

A	B	C
B	C	A
C	A	B



Condorcet winner may not exist!

Black (1950s): Choose Condorcet winner; if none, choose Borda winner

Copeland (1951): Choose candidate with highest outdegree – indegree in the majority graph

Many other voting schemes

- Plurality vote
  - Candidate with most # first positions is winner
- Instant runoff vote
  - If there is a majority winner, choose
  - Otherwise, eliminate least popular, repeat
  - President of Ireland, Australian parliament, many US university student elections
- Single-transferable vote
  - Malta, Republic of Ireland, Australian Senate
- ...

Arrow's theorem (1951)

The following are irreconcilable

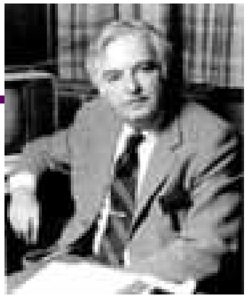
- Every result must be achievable somehow
- Monotonicity: Ranking higher should not hurt a candidate
- Independence of irrelevant attributes: Changes in rankings of "irrelevant alternatives" should have no impact on ranking of "relevant" subset
- Non-dictatorship

Conclusion: $\nexists$ satisfactory rank aggregation function

Borda vs. Condorcet debate

- Borda
  - Score-based
  - Consistent: two separate set of voters yield same ranking $\Rightarrow$ their union yields same ranking
  - Theorem: Any score-based method not Condorcet
- Condorcet
  - Majority-based
  - Meet Arrow's criteria where "independence of irrelevant attributes" criterion is modified
  - Winner may not exist

Kemeny's proposal (1959)



Axiomatic approach

- "Distance" between two preference orderings

  Distance = number of pair-wise disagreements
- Obtain ordering that is "least-distant" from the individual orderings

Theorem [Young Levenglick 1988]: Kemeny's rule is the unique preference function that is neutral, consistent, and Condorcet

- Reconciles Borda and Condorcet
- Satisfies additional properties (Pareto, anonymity)
- Maximum likelihood interpretation: [Young 1988]

Metrics on permutations

- Domain: $[n] = \{ 1, 2, \ldots, n \}$
- $\sigma \in S_n$
- $\sigma(i) < \sigma(j)$ means that "σ ranks i above j"

Kendall τ distance

Spearman's footrule distance

Kendall τ distance

$K(\sigma, \tau)$ = Number of pairs (i, j) such that σ ranks (i, j) in one order and τ ranks them in the opposite order

- Bubble-sort distance
- K is a metric
- K is right invariant: $K(\sigma, \tau) = K(\sigma \tau^{-1}, 1)$
- Eg

A	B
B	D
C	A
D	C

number of disagreements: 3

(AB, AD, CD)

Spearman's footrule distance

$$F(\sigma, \tau) = \sum_{i = 1, n} |\sigma(i) - \tau(i)|$$

- F is a metric (L_1-norm)
- F is right invariant: $F(\sigma, \tau) = F(\sigma\, \tau^{-1}, 1)$
- Eg,

A	B
B	D
C	A
D	C

shift(A) = 2 shift(B) = 1, etc, so

footrule distance: 6

There are several others, but...

Many of the other metrics are computationally expensive (some NP-hard, some not known to be polynomial-time computable, etc.)
[Diaconis; Group Representation in Probability and Statistics]

Also these two are perhaps the most natural for many applications

Diaconis-Graham inequality

$$K(\sigma, \tau) \leq F(\sigma, \tau) \leq 2\, K(\sigma, \tau)$$

$F(\sigma) \leq 2\, K(\sigma)$

$$
\begin{aligned}
F(\sigma) &= \textstyle\sum_i |\sigma(i) - i| \\
&= \textstyle\sum_i \mid \sum_j [\sigma(i) > \sigma(j)] - [i > j] \mid \\
&\leq \textstyle\sum_i \sum_j \mid [\sigma(i) > \sigma(j)] - [i > j] \mid \\
&= \textstyle\sum_{i,\, j} [\sigma(i) > \sigma(j),\ i < j] \\
&= 2\, K(\sigma)
\end{aligned}
$$

$K(\sigma) \leq F(\sigma)$

- $[i\colon j]$ = inversion $i < j$, $\sigma(i) > \sigma(j)$
  - Type 1 inversion if $\sigma(i) \geq j$
    $\Rightarrow i < j \leq \sigma(i)$
    $\Rightarrow \forall\, i,\ \#\{\, j \mid [i;\, j] \text{ is type 1 inversion } \} \leq \sigma(i) - i$
  - Type 2 inversion if $\sigma(i) \leq j$
    $\Rightarrow \sigma(j) < \sigma(i) \leq j$
    $\Rightarrow \forall\, j,\ \#\{\, i \mid [i;\, j] \text{ is type 2 inversion } \} \leq j - \sigma(j)$
- Each inversion is type 1, or type 2, or both

$K(\sigma) \leq$ type 1 inversions + type 2 inversions

$$\leq \textstyle\sum_{i \mid \sigma(i) > i} (\sigma(i) - i) + \sum_{j \mid j > \sigma(j)} (j - \sigma(j))$$

$$\leq F(\sigma)$$

Optimal aggregation

Given metric $d(\cdot,\cdot)$ and input permutations σ_1, …, σ_k, find permutation $\pi*$ such that

$$\sum_{i = 1, k} d(\sigma_i, \pi*)$$

is minimized

Kemeny (Kendall) optimal aggregation: $d = K$

Spearman footrule optimal aggregation: $d = F$

Kemeny optimal aggregation

Theorem [Bartholdi Tovey Trick 1989]: Kemeny optimal aggregation is NP-hard

Theorem: Kemeny optimal aggregation is NP-hard even for 4 lists

- Reduction using feedback arc set

c-approximate aggregation

Given metric $d(\cdot,\cdot)$ and input permutations σ_1, …, σ_k, find permutation π such that

$$\sum_{i=1,k} d(\sigma_i, \pi) \le c \cdot \sum_{i=1,k} d(\sigma_i, \pi^*)$$

Trivial approximation

Theorem: $2(1 - 1/k)$-approximation can be computed easily

Proof: K, F are metrics and simple geometry

$\pi^* =$ Optimal aggregation wrt. $d(\cdot,\cdot)$

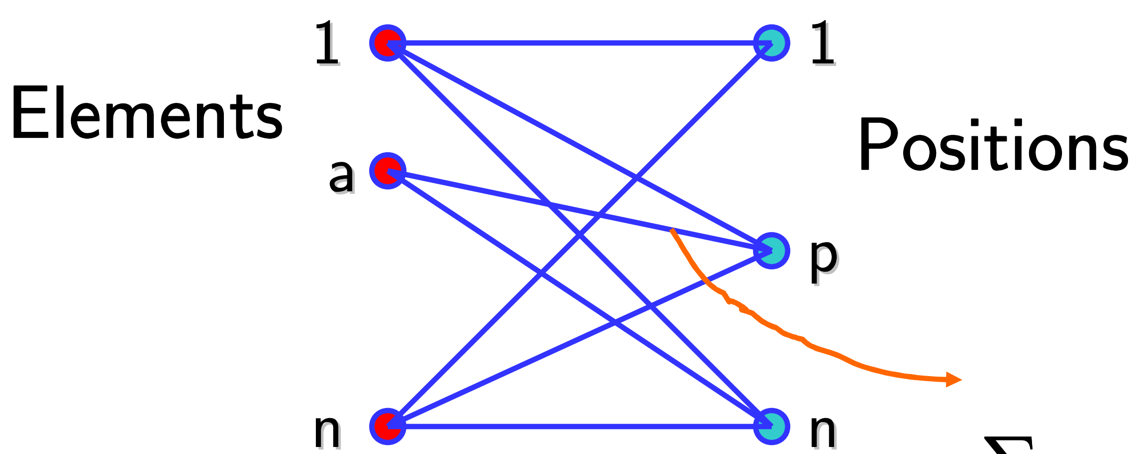
$i^* = \arg\min_i \sum_j d(\sigma_i, \sigma_j)$

$$\sum_j d(\sigma_j, \sigma_{i*}) \leq (1/k) \sum_{j, j'} d(\sigma_j, \sigma_{j'})$$
$$\leq (1/k) \sum_{j, j'} (d(\sigma_j, \pi^*) + d(\pi^*, \sigma_{j'}))$$
$$\leq 2 \sum_j d(\sigma_j, \pi^*)$$

Footrule optimal aggregation

Theorem [DKNS]: F-optimal aggregation can be computed in polynomial time

Proof: Via minimum cost perfect matching



$$\sum_{i = 1, k} |\sigma_i(a) - p|$$

2-approximation to K-optimum

Use Diaconis-Graham inequality

π = Footrule optimal aggregation

π^* = Kendall-optimal aggregation

$$\begin{aligned} \sum_i K(\sigma_i, \pi) &\leq \sum_i F(\sigma_i, \pi) \\ &\leq \sum_i F(\sigma_i, \pi^*) \\ &\leq 2 \sum_i K(\sigma_i, \pi^*) \end{aligned}$$

Heuristic: Median rank aggregation

Given $\sigma_1, \ldots, \sigma_k$,

$$\mu'(i) = \text{median}\ (\sigma_1(i), \ldots, \sigma_k(i))$$

Order μ' to obtain a permutation μ

Eg,

A	B	C
B	D	D
C	A	B
D	C	A

$\mu'(A) = 3$, $\mu'(B) = 2$, $\mu'(C) = 3$, $\mu'(D) = 2$

$\mu =$ B D A C

Median ranking is used in Olympic figure skating

Median rank aggregation

Theorem [DKNS]: If the median ranks of the candidates are unique (ie, form a permutation), then this permutation is a footrule optimal aggregation

What about using the median itself for ranking, even if it is not unique?

Median is a good approximation

Theorem [FKMSV]: Median rank aggregation is a 3-approximation to footrule optimal aggregation

Consistent permutations

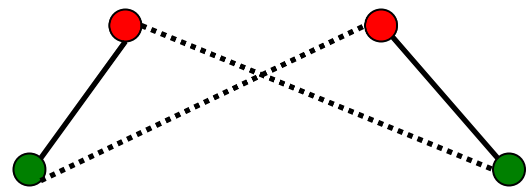
Given $\sigma' = \sigma'_1, \ldots, \sigma'_n$ where $\sigma'_i \in R$, call a permutation $\sigma \in S_n$ to be consistent with σ' if $\sigma'_i < \sigma'_j \Rightarrow \sigma(i) < \sigma(j)$

Consistency lemma: If σ is consistent with σ', then for any other permutation τ, $F(\sigma, \sigma') \leq F(\tau, \sigma')$

Proof of consistency lemma

Fact: $a' \leq b'$ and $a < b \Rightarrow$

$$|a - a'| + |b - b'| \leq |a - b'| + |a' - b|$$



If $\tau \neq \sigma$, apply this fact repeatedly to differing pairs until τ becomes σ

Each time $F(\tau, \sigma')$ can only improve

Median lemma

Fact: Given x_1, …, x_n where $x_i \in R$,

$$\text{median}(x_1, \ldots, x_n) = \arg\min_y \sum_i |x_i - y|$$

Median lemma: Given permutations σ_1, …, σ_k, let μ' denote their median function. Then, for any permutation τ,

$$\sum_i F(\mu', \sigma_i) \le \sum_i F(\tau, \sigma_i)$$

Proof of median theorem

Let τ be any permutation

$$
\begin{aligned}
\sum_i F(\mu, \sigma_i) &\le \sum_i F(\mu, \mu') + \sum_i F(\mu', \sigma_i) \quad \text{(triangle)} \\
&\le \sum_i F(\tau, \mu') + \sum_i F(\mu', \sigma_i) \quad \text{(consistency)} \\
&\le \sum_i F(\tau, \sigma_i) + 2 \sum_i F(\mu', \sigma_i) \quad \text{(triangle)} \\
&\le \sum_i F(\tau, \sigma_i) + 2 \sum_i F(\tau, \sigma_i) \quad \text{(median)} \\
&= 3 \sum_i F(\tau, \sigma_i)
\end{aligned}
$$

Merits of median

- Simple to implement
- Admits instance optimal algorithms [FLN]: among all algorithms that do sequential and random access to pre-sorted preference orders, the run-time of this median-finding algorithm is optimal up to a factor of 2 on every instance
- A good method for nearest-neighbor applications

Borda rank aggregation

Given σ_1, …, σ_k,

$$\beta'(i) = \sigma_1(i) + \cdots + \sigma_k(i)$$

Order β' to obtain a permutation β

Eg,

A	B	C
B	D	D
C	A	B
D	C	A

$\beta'(A) = 8$, $\beta'(B) = 6$, $\beta'(C) = 8$, $\beta'(D) = 8$

$\beta =$ B A C D

Borda is a good approximation

Theorem [FKMSV]: Borda rank aggregation is a 5-approximation to footrule optimal aggregation

Borda lemma: $\sum_i F(\beta', \sigma_i) \leq 2 \sum_i F(\mu', \sigma_i)$
Prove this point-wise for every j in the domain

Proof of Borda lemma

$\sum_i |\beta'(j) - \sigma_i(j)|$

$= \sum_i |(1/k \ \ \sum_{i'} \sigma_{i'}(j)) - \sigma_i(j) |$

$= (1/k) \sum_i |\sum_{i'}(\sigma_{i'}(j) - \sigma_i(j))|$

$\leq (1/k) \sum_{i,\ i'} | \sigma_{i'}(j) - \sigma_i(j) |$

$\leq (1/k) \sum_{i,\ i'} (| \sigma_{i'}(j) - \mu(j) | + | \sigma_i(j) - \mu(j) |)$

$= 2 \sum_i | \sigma_i(j) - \mu(j) |$

Proof of Borda theorem

Let τ be any permutation

$$
\begin{aligned}
\sum_i F(\beta, \sigma_i) &\le \sum_i F(\beta, \beta') + \sum_i F(\beta', \sigma_i) && \text{(triangle)}\\
&\le \sum_i F(\tau, \beta') + \sum_i F(\beta', \sigma_i) && \text{(consistency)}\\
&\le \sum_i F(\tau, \sigma_i) + 2 \sum_i F(\beta', \sigma_i) && \text{(triangle)}\\
&\le \sum_i F(\tau, \sigma_i) + 2 \sum_i F(\mu', \sigma_i) && \text{(Borda)}\\
&\le \sum_i F(\tau, \sigma_i) + 4 \sum_i F(\tau, \sigma_i) && \text{(median)}\\
&= 5 \sum_i F(\tau, \sigma_i)
\end{aligned}
$$

Copeland rank aggregation

Given $\sigma_1, \ldots, \sigma_k$,

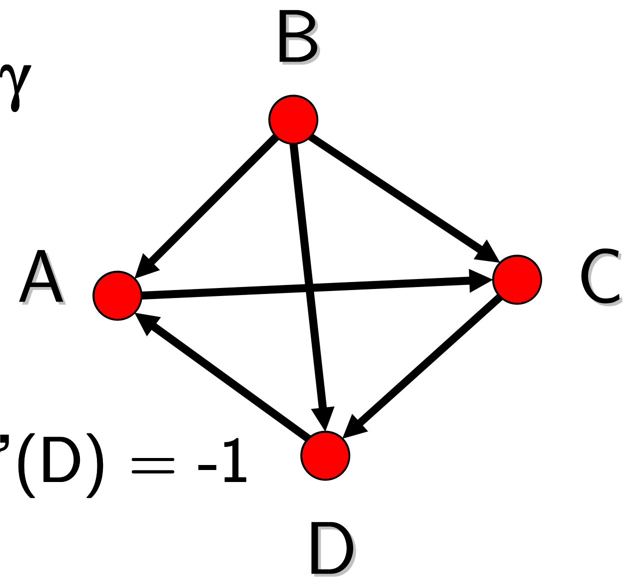
$\Gamma(i, j) = \text{majority} \{ \sigma_1(i) \text{ vs } \sigma_1(j), \ldots, \sigma_k(i) \text{ vs. } \sigma_k(j) \}$

$\gamma'(i) = \sum_i \Gamma(i, j) - \sum_j \Gamma(j, i)$

Order γ' to obtain a permutation γ

Eg,

A	B	C
B	D	D
C	A	B
D	C	A



$\gamma'(A) = -1, \gamma'(B) = 3, \gamma'(C) = -1, \gamma'(D) = -1$

$\gamma =$ B A C D

Copeland is a good approximation

Theorem [FKMSV]: Copeland rank aggregation is a 6-approximation to Kendall optimal aggregation

Proof: As before, but using K instead of F

Plurality method

Given $\sigma_1, \ldots, \sigma_k$,

$$\pi'(i) = \langle \ldots, \#\ \text{j-th place votes}, \ldots \rangle$$

Lexicographically order π' to obtain a permutation π

Eg,

A	B	C
B	D	D
C	A	B
D	C	A

$\pi'(A) = \langle\ 1\ 0\ 1\ 1\ \rangle$, $\pi'(B) = \langle\ 1\ 1\ 1\ 0\ \rangle$,

$\pi'(C) = \langle\ 1\ 0\ 1\ 1\ \rangle$, $\pi'(D) = \langle\ 0\ 2\ 0\ 1\ \rangle$

$\pi =$ B A C D

Plurality is not a good approximation

Theorem [FKMSV]: Plurality rank aggregation is not a good to approximation to Kendall optimal aggregation

Proof: n candidates, k voters, n >> k

1	1	2	3	4	…	k-1
2	2	3	2	2	…	2
3	3	4	4	3	…	3
			…			
n	n	1	1	1	…	1

$\pi = 1\ 2\ \ldots\ n$

$\sum_i F(\pi, \sigma_i) \geq (k\text{-}2)(n\text{-}1)$

$\beta = 2\ 3\ \ldots\ n\ 1$

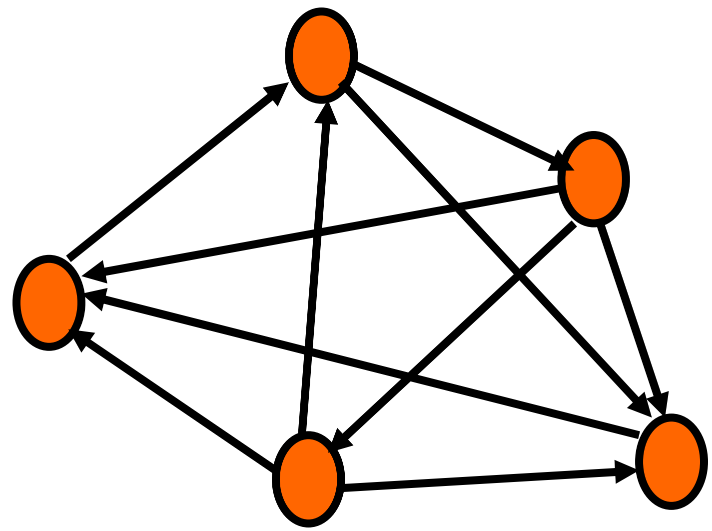
$\sum_i F(\beta, \sigma_i) \leq k^3 + n$

$n \uparrow \Rightarrow \text{Ratio} = \Omega(k)$

Rank agg vs min feedback arc set

- $G = (V, E)$, directed
- Tournament: each pair (i, j) has an edge

- Min feedback arc set (FAS): Find smallest $E' \subseteq E$ such that graph $(V, E - E')$ is acyclic



- V = candidates
- $(i, j) \in E$ = fraction of voters who rank i above j
- Acyclic = linear ordering

New approximation algorithm

Theorem [Ailon, Charikar, Newman]: There is a 11/7-approximation algorithm for rank aggregation

Proof: Consists of combining two approximation algorithms

1. Pick input closest to all other inputs (2-approx)
2. Construct a tournament and approximate FAS on a weighted tournament (2-approx)
3. Take the best of both solutions

FAS on unweighted tournaments

RQS(V, E)

- Pick a node $u \in V$ at random
- $V1 = \{ v \mid (v, u) \text{ in } E \}$, $E1 =$ edges in V1
- $V2 = \{ v \mid (u, v) \text{ in } E \}$, $E2 =$ edges in V2
- Output RQS(V1, E1) ° u ° RQS(V2, E2)

Randomized quicksort!

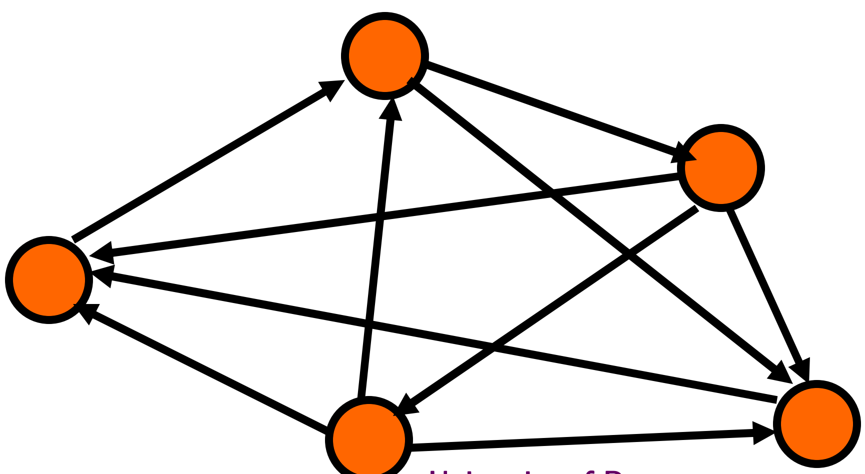
Theorem [ACN]: RQS is 3-approximation algorithm

The final word on this ...

Theorem [Kenyon-Mathieu, Schudy]: There is a PTAS for rank aggregation

Heuristics: Markov chains

- States = candidates
- Transitions = function of voting preferencess
  Probabilistically switch to a better candidate
- Final ranking = order of stationary probabilities



Advantages of Markov chains

- Handling partial lists and top k lists using available information to infer new ones
- Handling uneven comparisons and list lengths
- Motivation from PageRank---more wins better, more wins against good players even better
- With O(nk) preprocessing, O(k) per step for about O(n) steps

Sample Markov chains

If current state is candidate P, next state is:

- MC1: Choose uniformly from the multiset of all candidates that were ranked higher than or equal to P by some voter that ranked P

…

- MC4: Choose unifomly a candidate Q from all candidates and switch if the majority preferred Q to P

Metasearch results

- Using top 100 from major search engines
- Queries: affirmative action, alcoholism, ...

	K	F
Borda	0.214	0.345
Footrule	0.111	0.167
MC1	0.130	0.213
MC2	0.128	0.210
MC3	0.114	0.183
MC4	0.104	0.149

Other approaches to Metasearch

- Support vector machines [Joachims 2002]
- Learning [Cohen Schapire Singer 1999]
  - Hedge algorithm—iterative weight update
- Condorcet fusion [Montague Aslam 2002]
  - Finding Hamiltonian paths in Condorcet graphs
- Bayesian [Aslam Montague 2001]

Equivalent distant measures

Distance measure = non-negative, symmetric, regular binary function

Two distance measures $d(\cdot, \cdot)$ and $D(\cdot, \cdot)$ are equivalent if there is a constant $b > 0$ such that forall x, y in domain,

$$D(x, y) \leq d(x, y) \leq b \cdot D(x, y)$$

Theorem [FKS]: If d is a metric, then D satisfies approximate triangle inequality

Theorem [FKS]: If there is a factor c-aggregation algorithm with respect to d, then there is a factor (cb)-aggregation algorithm with respect to D

Comparing top k lists in IR

- Define distance measures to compare top *k* lists
- Useful in ranking search engine results, …
- τ_1, τ_2 = top k lists, D = dom τ_1 U dom τ_2

  $d_{min}(\tau_1, \tau_2) = \min_{\sigma_1 \geq \tau_1, \sigma_2 \geq \tau_2, \sigma_1, \tau_1 \in S_{|D|}} \{ d(\sigma_1, \sigma_2) \}$

  $d_{avg}(\tau_1, \tau_2) = \text{avg} \ldots$

  $d_{haus}(\tau_1, \tau_2) = \max \{ \max_{\sigma_1} \min_{\sigma_2} \ldots \}$

Theorem [FKS]: The distance measures K_{min}, K_{avg}, K_{haus}, F_{min}, F_{avg}, F_{haus}, are all in the same equivalence class

Comparing bucket orders

Bucket order = Linear order with "ties"

Profile-based measures

- K-profile(σ): $p_{i,j} = 1$ if $\sigma(i) < \sigma(j)$, 0 if $\sigma(i) = \sigma(j)$, -1 o.w.
- F-profile(σ): p_i = average position of i in its bucket
- $d_{prof} = L_1$ distance between d-profiles

Hausdorff measures as before

Theorem [FKMSV]: The distance measures K_{prof}, K_{haus}, F_{prof}, F_{haus} can be computed in polynomial time

Theorem [FKMSV]: The distance measures K_{prof}, K_{haus}, F_{prof}, F_{haus}, are all in the same equivalence class

Some open questions

- Improve the constants for median, Borda
- Is rank aggregation NP-hard for three lists?
- Aggregating wrt other metrics on permutations
  - Borda is near-optimal wrt Spearman's rho
- Practical algorithms for PTAS

Some references

- Dwork, Kumar, Naor, Sivakumar. Rank aggregation methods for the web, *WWW*, 2001.
- Fagin, Kumar, Sivakumar. Comparing top-k lists, *SODA*, 2003.
- Fagin, Kumar, Mahdian, Sivakumar, Vee. Comparing partial rankings, *PODS*, 2004.
- Ailon, Charikar, Newman. Aggregating incosistent information: Ranking and clustering, *STOC*, 2005.
- Kenyon-Mathieu, Schudy. How to rank with few errors: A PTAS for weighted feedback arc set on tournaments, *STOC*, 2007.

Thank you all!

ravikumar@yahoo-inc.com