Ranking, Aggregation, and You

Lester Mackey†

Collaborators: John C. Duchi† and Michael I. Jordan*

†Stanford University    *UC Berkeley

October 5, 2014
A simple question
A simple question

On a scale of 1 (very white) to 10 (very black), how black is this box?
A simple question

On a scale of 1 (very white) to 10 (very black), how black is this box?

Which box is blacker?
Another question

On a scale of 1 to 10, how relevant is this result for the query *flowers*?
Another question

On a scale of 1 to 10, how relevant is this result for the query flowers?
Another question

Google

Search

About 849,000,000 results (0.31 seconds)

**Flower - Wikipedia, the free encyclopedia**
en.wikipedia.org/wiki/Flower
A flower, sometimes known as a bloom or blossom, is the reproductive structure found in flowering plants (plants of the division Magnoliophyta, also called ...

**Church Street Flowers**
www.churchstreetflowers.com/
Florist specializing in contemporary custom designs for everyday occasions and weddings. Includes image galleries, business hours and location map.

**Flowers | Same Day Flower Delivery, Send Flowers | FromYouFlow…**
www.fromyouflowers.com/
Order flowers for delivery today! Nationwide flower delivery, starting at $25.49. Send flowers to celebrate every occasion with same day flower delivery.

**Flowers Online, Send Roses, Florist | 1-800-FLOWERS.COM Delivery**
www.1800flowers.com/
Order flowers, roses, and gift baskets online & send same day flower delivery for birthdays and anniversaries from trusted florist 1-800-Flowers.com.
What have we learned?

1. We are good at pairwise comparisons ▶ Much worse at absolute relevance judgments [Miller, 1956, Shiffrin and Nosofsky, 1994, Stewart, Brown, and Chater, 2005]

2. We are good at expressing sparse, partial preferences ▶ Much worse at expressing complete preferences

Complete preferences:
- ftd.com
- en.wikipedia.org/
- 1800flowers.com

What you express:
- ftd.com
- en.wikipedia.org/
- 1800flowers.com
What have we learned?

1. We are good at **pairwise** comparisons
   - Much worse at **absolute** relevance judgments

What have we learned?

1. We are good at **pairwise** comparisons
   ▶ Much worse at **absolute** relevance judgments

2. We are good at expressing **sparse, partial** preferences
   ▶ Much worse at expressing **complete** preferences

- Complete preferences:
  - ftd.com
  - 1800flowers.com
  - en.wikipedia.org/...

- What you express:
  - ftd.com
  - 1800flowers.com
  - en.wikipedia.org/...
**Ranking**

**Goal:** Order set of items/results to best match your preferences
Ranking

**Goal:** Order set of items/results to best match your preferences

- Web search: Return most relevant URLs for user queries
**Goal:** Order set of items/results to best match your preferences

- **Web search:** Return most relevant URLs for user queries
- **Recommendation systems:**
  - Movies to watch based on user’s past ratings
  - News articles to read based on past browsing history
  - Items to buy based on patron’s or other patrons’ purchases
Ranking procedures

**Goal:** Order set of items/results to best match your preferences

1. **Tractable:** Run in polynomial time
Ranking procedures

**Goal:** Order set of items/results to best match your preferences

1. **Tractable:** Run in polynomial time
2. **Consistent:** Recover true preferences given sufficient data

**Past work:**
1+2 are possible given complete preference data

- [Ravikumar, Tewari, and Yang, 2011, Buffoni, Calauzenes, Gallinari, and Usunier, 2011]
- This work: [Duchi, Mackey, and Jordan, 2013]

- Standard (tractable) procedures for ranking with partial preferences are inconsistent
- Aggregating partial preferences into more complete preferences can restore consistency
- New estimators based on $U$-statistics achieve 1+2+3
Ranking procedures

Goal: Order set of items/results to best match your preferences

1. **Tractable:** Run in polynomial time
2. **Consistent:** Recover true preferences given sufficient data
3. **Realistic:** Make use of ubiquitous partial preference data

Past work:

1+2 are possible given complete preference data

[Ravikumar, Tewari, and Yang, 2011, Buffoni, Calauzenes, Gallinari, and Usunier, 2011]

This work

[Duchi, Mackey, and Jordan, 2013]

Standard (tractable) procedures for ranking with partial preferences are inconsistent

Aggregating partial preferences into more complete preferences can restore consistency

New estimators based on $U$-statistics achieve 1+2+3
Ranking procedures

**Goal:** Order set of items/results to best match your preferences

1. **Tractable:** Run in polynomial time
2. **Consistent:** Recover true preferences given sufficient data
3. **Realistic:** Make use of ubiquitous partial preference data

**Past work:** 1+2 are possible given *complete* preference data

[Ravikumar, Tewari, and Yang, 2011, Buffoni, Calauzenes, Gallinari, and Usunier, 2011]
Ranking procedures

**Goal:** Order set of items/results to best match your preferences

1. **Tractable:** Run in polynomial time
2. **Consistent:** Recover true preferences given sufficient data
3. **Realistic:** Make use of ubiquitous partial preference data

**Past work:** 1+2 are possible given complete preference data

[Ravikumar, Tewari, and Yang, 2011, Buffoni, Calauzenes, Gallinari, and Usunier, 2011]

**This work** [Duchi, Mackey, and Jordan, 2013]
Ranking procedures

**Goal:** Order set of items/results to best match your preferences

1. **Tractable:** Run in polynomial time
2. **Consistent:** Recover true preferences given sufficient data
3. **Realistic:** Make use of ubiquitous partial preference data

**Past work:** 1+2 are possible given *complete* preference data

[Ravikumar, Tewari, and Yang, 2011, Buffoni, Calauzenes, Gallinari, and Usunier, 2011]

**This work**  [Duchi, Mackey, and Jordan, 2013]

- Standard (tractable) procedures for ranking with partial preferences are *inconsistent*
Ranking procedures

**Goal:** Order set of items/results to best match your preferences

1. **Tractable:** Run in polynomial time
2. **Consistent:** Recover true preferences given sufficient data
3. **Realistic:** Make use of ubiquitous partial preference data

**Past work:** 1+2 are possible given complete preference data

[Ravikumar, Tewari, and Yang, 2011, Buffoni, Calauzenes, Gallinari, and Usunier, 2011]

**This work** [Duchi, Mackey, and Jordan, 2013]

- Standard (tractable) procedures for ranking with partial preferences are *inconsistent*
- **Aggregating** partial preferences into more complete preferences can restore consistency
Ranking procedures

**Goal:** Order set of items/results to best match your preferences

1. **Tractable:** Run in polynomial time
2. **Consistent:** Recover true preferences given sufficient data
3. **Realistic:** Make use of ubiquitous partial preference data

**Past work:** 1+2 are possible given complete preference data

[Ravikumar, Tewari, and Yang, 2011, Buffoni, Calauzenes, Gallinari, and Usunier, 2011]

**This work** [Duchi, Mackey, and Jordan, 2013]

- Standard (tractable) procedures for ranking with partial preferences are inconsistent
- **Aggregating** partial preferences into more complete preferences can restore consistency
- New estimators based on $U$-statistics achieve $1+2+3$
Outline

Supervised Ranking
  Formal definition
  Tractable surrogates
  Pairwise inconsistency

Aggregation
  Restoring consistency
  Estimating complete preferences

U-statistics
  Practical procedures
  Experimental results
Outline

**Supervised Ranking**
- Formal definition
- Tractable surrogates
- Pairwise inconsistency

**Aggregation**
- Restoring consistency
- Estimating complete preferences

**U-statistics**
- Practical procedures
- Experimental results
Supervised ranking

**Observe:** Sequence of training examples
Supervised ranking

Observe: Sequence of training examples
  - Query $Q$: e.g., search term “flowers”
Supervised ranking

Observe: Sequence of training examples

- Query $Q$: e.g., search term “flowers”
- Set of $m$ items $\mathcal{I}_Q$ to rank
  - e.g., websites $\{1, 2, 3, 4\}$
Supervised ranking

**Observe:** Sequence of training examples

- **Query** $Q$: e.g., search term “flowers”
- Set of $m$ items $I_Q$ to rank
  - e.g., websites $\{1, 2, 3, 4\}$
- **Label** $Y$ representing some preference structure over items
Supervised ranking

**Observe:** Sequence of training examples

- **Query** $Q$: e.g., search term “flowers”
- Set of $m$ items $\mathcal{I}_Q$ to rank
  - e.g., websites $\{1, 2, 3, 4\}$
- **Label** $Y$ representing some preference structure over items
  - Item 1 preferred to $\{2, 3\}$ and item 3 to 4

Example: $Y$ is a graph on items $\{1, 2, 3, 4\}$
Supervised ranking

Observe: \((Q_1, Y_1), \ldots, (Q_n, Y_n)\)

Learn: Scoring function \(f\) to induce item rankings for each query

\[\alpha_i = f_i(Q)\]

Real-valued score for each item \(i\) in item set \(I_Q\)

Vector of scores \(f(Q)\) induces ranking over \(I_Q\) where \(i\) ranked above \(j\) if \(\alpha_i > \alpha_j\)
Supervised ranking

Observe: \((Q_1, Y_1), \ldots, (Q_n, Y_n)\)

Learn: Scoring function \(f\) to induce item rankings for each query
  - Real-valued score for each item \(i\) in item set \(I_Q\)
    \[
    \alpha_i := f_i(Q)
    \]
  - Vector of scores \(f(Q)\) induces ranking over \(I_Q\)
    \[
    i \text{ ranked above } j \iff \alpha_i > \alpha_j
    \]
Supervised ranking

Example: Scoring function $f$ with scores

$$f_1(Q) > f_2(Q) > f_3(Q)$$

induces same ranking as preference graph $Y$

\[ \begin{align*}
1 & \rightarrow 2 \\
2 & \rightarrow 3 \\
\end{align*} \]

\[ f_1(Q) > f_2(Q) \]

\[ f_2(Q) > f_3(Q) \]
Supervised ranking

Observe: \((Q_1, Y_1), \ldots, (Q_n, Y_n)\)

Learn: Scoring function \(f\) to predict item ranking

Suffer loss: \(L(f(Q), Y)\)
  - Encodes discord between observed label \(Y\) and prediction \(f(Q)\)
  - Depends on specific ranking task and available data
Supervised ranking

**Example:** Pairwise loss
Supervised ranking

**Example:** Pairwise loss

- Let $Y = \text{(weighted) adjacency matrix for a preference graph}$
  - $Y_{ij} = \text{the preference weight on edge (}i, j\text{)}$

```
L(\alpha, Y) = Y_{12} \mathbb{1}(\alpha_1 \leq \alpha_2) + Y_{13} \mathbb{1}(\alpha_1 \leq \alpha_3) + Y_{34} \mathbb{1}(\alpha_3 \leq \alpha_4)
```
Supervised ranking

Example: Pairwise loss

- Let $Y = (\text{weighted})$ adjacency matrix for a preference graph
- $Y_{ij} =$ the preference weight on edge $(i, j)$
- Let $\alpha = f(Q)$ be the predicted scores for query $Q$

\[
L(\alpha, Y) = \sum_{i \neq j} Y_{ij} 1(\alpha_i \leq \alpha_j)
\]

Imposes penalty for each misordered edge
Supervised ranking

Example: Pairwise loss

- Let $Y = \text{(weighted) adjacency matrix for a preference graph}$
  - $Y_{ij}$ = the preference weight on edge $(i, j)$
- Let $\alpha = f(Q)$ be the predicted scores for query $Q$
- Then, $L(\alpha, Y) = \sum_{i \neq j} Y_{ij} 1(\alpha_i \leq \alpha_j)$
- Imposes penalty for each misordered edge

$$L(\alpha, Y) = Y_{12} 1(\alpha_1 \leq \alpha_2) + Y_{13} 1(\alpha_1 \leq \alpha_3) + Y_{34} 1(\alpha_3 \leq \alpha_4)$$
Supervised ranking

Observe: \((Q_1, Y_1), \ldots (Q_n, Y_n)\)

Learn: Scoring function \(f\) to rank items

Suffer loss: \(L(f(Q), Y)\)

Goal: Minimize the risk \(R(f) := \mathbb{E}[L(f(Q), Y)]\)
Supervised ranking

**Observe:** $\ (Q_1, Y_1), \ldots (Q_n, Y_n) \$

**Learn:** Scoring function $f$ to rank items

**Suffer loss:** $L(f(Q), Y)$

**Goal:** Minimize the risk $R(f) := \mathbb{E} [L(f(Q), Y)]$

**Main Question:** Are there **tractable** ranking procedures that minimize $R$ as $n \to \infty$?
Tractable ranking

**First try:** Empirical risk minimization

\[
\min_f \hat{R}_n(f) := \hat{E}_n [L(f(Q), Y)] = \frac{1}{n} \sum_{k=1}^n L(f(Q_k), Y_k)
\]
Tractable ranking

**First try:** Empirical risk minimization ← Intractable!

$$\min_f \hat{R}_n(f) := \mathbb{E}_n [L(f(Q), Y)] = \frac{1}{n} \sum_{k=1}^{n} L(f(Q_k), Y_k)$$
Tractable ranking

**First try:** Empirical risk minimization $\leftarrow$ Intractable!

$$
\min_f \tilde{R}_n(f) := \hat{E}_n [L(f(Q), Y)] = \frac{1}{n} \sum_{k=1}^{n} L(f(Q_k), Y_k)
$$

$$
L(\alpha, Y) = \sum_{i \neq j} Y_{ij} 1(\alpha_i \leq \alpha_j)
$$

Hard
Tractable ranking

First try: Empirical risk minimization $\leftarrow$ Intractable!

$$\min_f \hat{R}_n(f) := \hat{E}_n [L(f(Q), Y)] = \frac{1}{n} \sum_{k=1}^{n} L(f(Q_k), Y_k)$$

Idea: Replace loss $L(\alpha, Y)$ with convex surrogate $\varphi(\alpha, Y)$

$$L(\alpha, Y) = \sum_{i \neq j} Y_{ij} 1_{(\alpha_i \leq \alpha_j)}$$
Tractable ranking

First try: Empirical risk minimization $\leftarrow$ Intractable!

$$
\min_f \hat{R}_n(f) := \hat{E}_n [L(f(Q), Y)] = \frac{1}{n} \sum_{k=1}^{n} L(f(Q_k), Y_k)
$$

Idea: Replace loss $L(\alpha, Y)$ with convex surrogate $\varphi(\alpha, Y)$

$$
L(\alpha, Y) = \sum_{i \neq j} Y_{ij} 1(\alpha_i \leq \alpha_j) \\
\varphi(\alpha, Y) = \sum_{i \neq j} Y_{ij} \phi(\alpha_i - \alpha_j)
$$

Hard

Tractable
Surrogate ranking

**Idea:** Empirical *surrogate* risk minimization

$$\min_f \hat{R}_{\varphi,n}(f) := \hat{E}_n [\varphi(f(Q), Y)] = \frac{1}{n} \sum_{k=1}^n \varphi(f(Q_k), Y_k)$$
Surrogate ranking

**Idea:** Empirical *surrogate* risk minimization

\[
\min_f \hat{R}_{\varphi,n}(f) := \hat{E}_n [\varphi(f(Q), Y)] = \frac{1}{n} \sum_{k=1}^{n} \varphi(f(Q_k), Y_k)
\]

- If \( \varphi \) convex, then minimization is *tractable*

\[
\Rightarrow \text{Main Question: Are these tractable ranking procedures consistent?}
\]

\[
\Rightarrow \text{Does } \arg\min_f R_{\varphi}(f) = \arg\min_f R_{\varphi,n}(f) \Rightarrow \text{Does the true risk also minimize?}
\]
Surrogate ranking

**Idea:** Empirical *surrogate* risk minimization

\[
\min_{f} \hat{R}_{\varphi,n}(f) := \mathbb{E}_n [\varphi(f(Q), Y)] = \frac{1}{n} \sum_{k=1}^{n} \varphi(f(Q_k), Y_k)
\]

- If \( \varphi \) convex, then minimization is tractable
- \( \arg\min_{f} \hat{R}_{\varphi,n}(f) \xrightarrow{n \to \infty} \arg\min_{f} R_{\varphi}(f) := \mathbb{E} [\varphi(f(Q), Y)] \)
Surrogate ranking

**Idea:** Empirical *surrogate* risk minimization

\[
\min_f \hat{R}_{\varphi,n}(f) := \hat{E}_n [\varphi(f(Q), Y)] = \frac{1}{n} \sum_{k=1}^n \varphi(f(Q_k), Y_k)
\]

- If \( \varphi \) convex, then minimization is tractable
- \( \arg\min_f \hat{R}_{\varphi,n}(f) \xrightarrow{n \to \infty} \arg\min_f R_{\varphi}(f) := \mathbb{E} [\varphi(f(Q), Y)] \)

**Main Question:**
Are these tractable ranking procedures *consistent*?
Surrogate ranking

**Idea:** Empirical *surrogate* risk minimization

\[
\min_f \hat{R}_{\varphi,n}(f) := \mathbb{E}_n [\varphi(f(Q), Y)] = \frac{1}{n} \sum_{k=1}^{n} \varphi(f(Q_k), Y_k)
\]

- If \(\varphi\) convex, then minimization is **tractable**
- \(\arg\min_f \hat{R}_{\varphi,n}(f) \xrightarrow{n \to \infty} \arg\min_f R_{\varphi}(f) := \mathbb{E} [\varphi(f(Q), Y)]\)

**Main Question:**
Are these tractable ranking procedures **consistent**?

\(\iff\)
Does \(\arg\min_f R_{\varphi}(f)\) also minimize the true risk \(R(f)\)?
Classification consistency

Consider the special case of classification

\[ \text{Pairwise loss:} \quad L(\alpha, Y) = Y_{01} \max(\alpha_0 - \alpha_1) + Y_{10} \max(\alpha_1 - \alpha_0) \]

\[ \text{Surrogate loss:} \quad \phi(\alpha, Y) = Y_{01} \phi(\alpha_0 - \alpha_1) + Y_{10} \phi(\alpha_1 - \alpha_0) \]

Theorem: If \( \phi \) is convex, procedure based on minimizing \( \phi \) is consistent if and only if \( \phi'(0) < 0 \).

[Bartlett, Jordan, and McAuliffe, 2006]}

⇒ Tractable consistency for boosting, SVMs, logistic regression
Classification consistency

Consider the special case of classification

- Observe: query $X$, items $\{0, 1\}$, label $Y_{01} = 1$ or $Y_{10} = 1$
Classification consistency

Consider the special case of classification

- Observe: query $X$, items $\{0, 1\}$, label $Y_{01} = 1$ or $Y_{10} = 1$
- Pairwise loss: $L(\alpha, Y) = Y_{01}1_{(\alpha_0 \leq \alpha_1)} + Y_{10}1_{(\alpha_1 \leq \alpha_0)}$
Classification consistency

Consider the special case of classification

- Observe: query $X$, items $\{0, 1\}$, label $Y_{01} = 1$ or $Y_{10} = 1$
- Pairwise loss: $L(\alpha, Y) = Y_{01} \mathbb{1}_{(\alpha_0 \leq \alpha_1)} + Y_{10} \mathbb{1}_{(\alpha_1 \leq \alpha_0)}$
- Surrogate loss: $\varphi(\alpha, Y) = Y_{01} \phi(\alpha_0 - \alpha_1) + Y_{10} \phi(\alpha_1 - \alpha_0)$

Theorem: If $\varphi$ is convex, procedure based on minimizing $\varphi$ is consistent if and only if $\varphi'(0) < 0$.

[Bartlett, Jordan, and McAuliffe, 2006] ⇒ Tractable consistency for boosting, SVMs, logistic regression
Classification consistency

Consider the special case of classification

- Observe: query $X$, items $\{0, 1\}$, label $Y_{01} = 1$ or $Y_{10} = 1$
- Pairwise loss: $L(\alpha, Y) = Y_{01}1(\alpha_0 \leq \alpha_1) + Y_{10}1(\alpha_1 \leq \alpha_0)$
- Surrogate loss: $\varphi(\alpha, Y) = Y_{01}\phi(\alpha_0 - \alpha_1) + Y_{10}\phi(\alpha_1 - \alpha_0)$

**Theorem:** If $\phi$ is convex, procedure based on minimizing $\phi$ is consistent if and only if $\phi'(0) < 0$. [Bartlett, Jordan, and McAuliffe, 2006]

$\Rightarrow$ **Tractable consistency** for boosting, SVMs, logistic regression
Good news: Can characterize surrogate ranking consistency

1 [Duchi, Mackey, and Jordan, 2013]
Ranking consistency?

**Good news:** Can characterize surrogate ranking consistency

**Theorem:** Procedure based on minimizing $\varphi$ is consistent $\iff$

$$\min_{\alpha} \left\{ \mathbb{E}[\varphi(\alpha, Y) \mid q] \mid \alpha \notin \arg\min_{\alpha'} \mathbb{E}[L(\alpha', Y) \mid q] \right\}$$

$$> \min_{\alpha} \mathbb{E}[\varphi(\alpha, Y) \mid q].$$

- Translation: $\varphi$ is consistent if and only if minimizing conditional surrogate risk gives correct ranking for every query

---

1[Duchi, Mackey, and Jordan, 2013]
Bad news: The consequences are dire...
Ranking consistency?

**Bad news:** The consequences are dire...

Consider the pairwise loss:

\[ L(\alpha, Y) = \sum_{i \neq j} Y_{ij} 1(\alpha_i \leq \alpha_j) \]
Ranking consistency?

**Bad news:** The consequences are dire...

Consider the pairwise loss:

\[
L(\alpha, Y) = \sum_{i \neq j} Y_{ij} 1(\alpha_i \leq \alpha_j)
\]

**Task:** Find \( \arg\min_\alpha \mathbb{E}[L(\alpha, Y) \mid q] \)

Task diagram:

1 -- \( y_{12} \) -- 2

1 -- \( y_{13} \) -- 3

2 -- \( y_{34} \) -- 4
Ranking consistency?

Bad news: The consequences are dire...

Consider the pairwise loss:

\[ L(\alpha, Y) = \sum_{i \neq j} Y_{ij} 1(\alpha_i \leq \alpha_j) \]

**Task:** Find \( \text{argmin}_\alpha \mathbb{E}[L(\alpha, Y) \mid q] \)

- Classification (two node) case: Easy
  - Choose \( \alpha_0 > \alpha_1 \iff \mathbb{P}[\text{Class 0} \mid q] > \mathbb{P}[\text{Class 1} \mid q] \)
Ranking consistency?

**Bad news:** The consequences are dire...

Consider the pairwise loss:

\[ L(\alpha, Y) = \sum_{i \neq j} Y_{ij} 1(\alpha_i \leq \alpha_j) \]

**Task:** Find \( \text{argmin}_\alpha \mathbb{E}[L(\alpha, Y) \mid q] \)

- Classification (two node) case: Easy
  - Choose \( \alpha_0 > \alpha_1 \iff \mathbb{P}[	ext{Class 0} \mid q] > \mathbb{P}[	ext{Class 1} \mid q] \)

- General case: NP hard
  - Unless \( P = NP \), must restrict problem for tractable consistency
Low noise distribution

**Define:** Average preference for item $i$ over item $j$:

$$s_{ij} = \mathbb{E}[Y_{ij} \mid q]$$

- We say $i \succ j$ on average if $s_{ij} > s_{ji}$

**Definition (Low noise distribution):** If $i \succ j$ on average and $j \succ k$ on average, then $i \succ k$ on average.

Low noise $\Rightarrow s_{13} > s_{31}$

- No cyclic preferences on average

- Find $\arg\min \alpha \mathbb{E}[L(\alpha, Y) \mid q]$: Very easy

- Choose $\alpha_i > \alpha_j \iff s_{ij} > s_{ji}$
Low noise distribution

**Define:** Average preference for item $i$ over item $j$:

$$s_{ij} = \mathbb{E}[Y_{ij} \mid q]$$

- We say $i \succ j$ on average if $s_{ij} > s_{ji}$

**Definition** *(Low noise distribution)*: If $i \succ j$ on average and $j \succ k$ on average, then $i \succ k$ on average.

- No cyclic preferences on average

Low noise

$\Rightarrow s_{13} > s_{31}$
Low noise distribution

**Define:** Average preference for item $i$ over item $j$:

$$ s_{ij} = \mathbb{E}[Y_{ij} | q] $$

- We say $i \succ j$ on average if $s_{ij} > s_{ji}$

**Definition (Low noise distribution):** If $i \succ j$ on average and $j \succ k$ on average, then $i \succ k$ on average.

- No cyclic preferences on average
- Find $\arg\min_\alpha \mathbb{E}[L(\alpha, Y) | q]$: Very easy
  - Choose $\alpha_i > \alpha_j \iff s_{ij} > s_{ji}$
Ranking consistency?

**Pairwise ranking surrogate:**


\[
\varphi(\alpha, Y) = \sum_{ij} Y_{ij} \phi(\alpha_i - \alpha_j)
\]

for \( \phi \) convex with \( \phi'(0) < 0 \). Common in ranking literature.

Theorem:

\( \varphi \) is not consistent, even in low noise settings.

[Duchi, Mackey, and Jordan, 2013] \( \Rightarrow \) Inconsistency for RankBoost, RankSVM, Logistic Ranking...
Ranking consistency?

**Pairwise ranking surrogate:**


\[ \phi(\alpha, Y) = \sum_{ij} Y_{ij} \phi(\alpha_i - \alpha_j) \]

for \( \phi \) convex with \( \phi'(0) < 0 \). Common in ranking literature.

**Theorem:** \( \varphi \) is not consistent, even in low noise settings.

[Duchi, Mackey, and Jordan, 2013]
Ranking consistency?

Pairwise ranking surrogate:


\[ \varphi(\alpha, Y) = \sum_{ij} Y_{ij} \phi(\alpha_i - \alpha_j) \]

for \( \phi \) convex with \( \phi'(0) < 0 \). Common in ranking literature.

**Theorem:** \( \varphi \) is not consistent, even in low noise settings.

[Duchi, Mackey, and Jordan, 2013]

\[ \Rightarrow \text{Inconsistency} \] for RankBoost, RankSVM, Logistic Ranking...
Ranking with pairwise data is challenging

▶ Inconsistent in general (unless $P=NP$)
▶ Low noise distributions
  $\phi(\alpha,Y) = \sum_{ij} Y_{ij} \phi(\alpha_i - \alpha_j)$
  Inconsistent for standard convex losses
  $\phi(\alpha,Y) = \sum_{ij} \phi(\alpha_i - \alpha_j - Y_{ij})$
  Inconsistent for margin-based convex losses

Question: Do tractable consistent losses exist for partial preference data?

Yes!
Ranking with pairwise data is challenging

- Inconsistent in general (unless $P = NP$)

Question: Do tractable consistent losses exist for partial preference data? Yes!
Ranking with pairwise data is challenging

- Inconsistent in general (unless $P = NP$)
- Low noise distributions

Question:
Do tractable consistent losses exist for partial preference data?

Yes!
Ranking with pairwise data is challenging

- Inconsistent in general (unless $P = NP$)
- Low noise distributions
  - Inconsistent for standard convex losses

\[ \varphi(\alpha, Y) = \sum_{ij} Y_{ij} \phi(\alpha_i - \alpha_j) \]

Question: Do tractable consistent losses exist for partial preference data? Yes!
Ranking with pairwise data is challenging

- Inconsistent in general (unless $P = NP$)
- Low noise distributions
  - Inconsistent for standard convex losses
    \[
    \varphi(\alpha, Y) = \sum_{ij} Y_{ij} \phi(\alpha_i - \alpha_j)
    \]
  - Inconsistent for margin-based convex losses
    \[
    \varphi(\alpha, Y) = \sum_{ij} \phi(\alpha_i - \alpha_j - Y_{ij})
    \]
Ranking with pairwise data is challenging

- Inconsistent in general (unless $P = NP$)
- Low noise distributions
  - Inconsistent for standard convex losses
    \[
    \varphi(\alpha, Y) = \sum_{ij} Y_{ij} \phi(\alpha_i - \alpha_j)
    \]
  - Inconsistent for margin-based convex losses
    \[
    \varphi(\alpha, Y) = \sum_{ij} \phi(\alpha_i - \alpha_j - Y_{ij})
    \]

**Question:**
Do tractable consistent losses exist for partial preference data?
Ranking with pairwise data is challenging

- Inconsistent in general (unless $P = NP$)
- Low noise distributions
  - Inconsistent for standard convex losses
    \[ \varphi(\alpha, Y) = \sum_{ij} Y_{ij} \phi(\alpha_i - \alpha_j) \]
  - Inconsistent for margin-based convex losses
    \[ \varphi(\alpha, Y) = \sum_{ij} \phi(\alpha_i - \alpha_j - Y_{ij}) \]

**Question:**
Do tractable consistent losses exist for partial preference data?

**Yes!**
Ranking with pairwise data is challenging

- Inconsistent in general (unless $P = NP$)
- Low noise distributions
  - Inconsistent for standard convex losses
    \[ \phi(\alpha, Y) = \sum_{i,j} Y_{ij} \phi(\alpha_i - \alpha_j) \]
  - Inconsistent for margin-based convex losses
    \[ \phi(\alpha, Y) = \sum_{i,j} \phi(\alpha_i - \alpha_j - Y_{ij}) \]

**Question:**
Do tractable consistent losses exist for partial preference data?

**Yes,** if we aggregate!
Outline

Supervised Ranking
  Formal definition
  Tractable surrogates
  Pairwise inconsistency

Aggregation
  Restoring consistency
  Estimating complete preferences

U-statistics
  Practical procedures
  Experimental results
An observation

Can rewrite risk of pairwise loss

\[ \mathbb{E}[L(\alpha, Y) \mid q] = \sum_{i \neq j} s_{ij} 1(\alpha_i \leq \alpha_j) \]

where \( s_{ij} = \mathbb{E}[Y_{ij} \mid q] \).
An observation

Can rewrite risk of pairwise loss

\[ \mathbb{E}[L(\alpha, Y) \mid q] = \sum_{i \neq j} s_{ij} 1(\alpha_i \leq \alpha_j) = \sum_{i \neq j} \max\{s_{ij} - s_{ji}, 0\} 1(\alpha_i \leq \alpha_j) \]

where \( s_{ij} = \mathbb{E}[Y_{ij} \mid q] \).

- Only depends on net expected preferences: \( s_{ij} - s_{ji} \)
An observation

Can rewrite risk of pairwise loss

\[ \mathbb{E}[L(\alpha, Y) \mid q] = \sum_{i \neq j} s_{ij} 1(\alpha_i \leq \alpha_j) = \sum_{i \neq j} \max\{s_{ij} - s_{ji}, 0\} 1(\alpha_i \leq \alpha_j) \]

where \( s_{ij} = \mathbb{E}[Y_{ij} \mid q] \).

- Only depends on net expected preferences: \( s_{ij} - s_{ji} \)

Consider the surrogate

\[ \varphi(\alpha, s) := \sum_{i \neq j} \max\{s_{ij} - s_{ji}, 0\} \phi(\alpha_i - \alpha_j) \]

for \( \phi \) non-increasing and convex, with \( \phi'(0) < 0 \).
An observation

Can rewrite risk of pairwise loss

\[
\mathbb{E}[L(\alpha, Y) \mid q] = \sum_{i \neq j} s_{ij}1_{(\alpha_i \leq \alpha_j)} = \sum_{i \neq j} \max\{s_{ij} - s_{ji}, 0\}1_{(\alpha_i \leq \alpha_j)}
\]

where \( s_{ij} = \mathbb{E}[Y_{ij} \mid q] \).

- Only depends on net expected preferences: \( s_{ij} - s_{ji} \)

Consider the surrogate

\[
\varphi(\alpha, s) := \sum_{i \neq j} \max\{s_{ij} - s_{ji}, 0\}\phi(\alpha_i - \alpha_j) \neq \sum_{i \neq j} s_{ij}\phi(\alpha_i - \alpha_j)
\]

for \( \phi \) non-increasing and convex, with \( \phi'(0) < 0 \).

- Either \( i \rightarrow j \) penalized or \( j \rightarrow i \) but not both
An observation

Can rewrite risk of pairwise loss

\[ \mathbb{E}[L(\alpha, Y) \mid q] = \sum_{i \neq j} s_{ij} 1(\alpha_i \leq \alpha_j) = \sum_{i \neq j} \max\{s_{ij} - s_{ji}, 0\} 1(\alpha_i \leq \alpha_j) \]

where \( s_{ij} = \mathbb{E}[Y_{ij} \mid q] \).

- Only depends on net expected preferences: \( s_{ij} - s_{ji} \)

Consider the surrogate

\[
\varphi(\alpha, s) := \sum_{i \neq j} \max\{s_{ij} - s_{ji}, 0\} \phi(\alpha_i - \alpha_j) \neq \sum_{i \neq j} s_{ij} \phi(\alpha_i - \alpha_j)
\]

for \( \phi \) non-increasing and convex, with \( \phi'(0) < 0 \).

- Either \( i \rightarrow j \) penalized or \( j \rightarrow i \) but not both
- **Consistent** whenever average preferences are acyclic
What happened?

Old surrogates: \[ \mathbb{E}[\varphi(\alpha, Y) \mid q] = \lim_{k \to \infty} \frac{1}{k} \sum_k \varphi(\alpha, Y_k) \]

- Loss \( \varphi(\alpha, Y) \) applied to a single datapoint
What happened?

Old surrogates: \( \mathbb{E}[\varphi(\alpha, Y) \mid q] = \lim_{k \to \infty} \frac{1}{k} \sum_k \varphi(\alpha, Y_k) \)

- Loss \( \varphi(\alpha, Y) \) applied to a single datapoint

New surrogates: \( \varphi(\alpha, \mathbb{E}[Y \mid q]) = \lim_{k \to \infty} \varphi(\alpha, \frac{1}{k} \sum_k Y_k) \)

- Loss applied to aggregation of many datapoints

New framework: Ranking with aggregate losses

\( L(\alpha, s_k(Y_1, \ldots, Y_k)) \) and \( \varphi(\alpha, s_k(Y_1, \ldots, Y_k)) \) where \( s_k \) is a structure function that aggregates first \( k \) datapoints

- \( s_k \) combines partial preferences into more complete estimates

- Consistency characterization extends to this setting
What happened?

Old surrogates:  \( \mathbb{E}[\varphi(\alpha, Y) \mid q] = \lim_{k \to \infty} \frac{1}{k} \sum_k \varphi(\alpha, Y_k) \)
- Loss \( \varphi(\alpha, Y) \) applied to a single datapoint

New surrogates:  \( \varphi(\alpha, \mathbb{E}[Y \mid q]) = \lim_{k \to \infty} \varphi(\alpha, \frac{1}{k} \sum_k Y_k) \)
- Loss applied to aggregation of many datapoints

New framework: Ranking with aggregate losses

\[ L(\alpha, s_k(Y_1, \ldots, Y_k)) \quad \text{and} \quad \varphi(\alpha, s_k(Y_1, \ldots, Y_k)) \]

where \( s_k \) is a structure function that aggregates first \( k \) datapoints
What happened?

Old surrogates: \( \mathbb{E}[\varphi(\alpha, Y) \mid q] = \lim_{k \to \infty} \frac{1}{k} \sum_k \varphi(\alpha, Y_k) \)

- Loss \( \varphi(\alpha, Y) \) applied to a single datapoint

New surrogates: \( \varphi(\alpha, \mathbb{E}[Y \mid q]) = \lim_{k \to \infty} \varphi(\alpha, \frac{1}{k} \sum_k Y_k) \)

- Loss applied to aggregation of many datapoints

New framework: Ranking with aggregate losses

\[
L(\alpha, s_k(Y_1, \ldots, Y_k)) \quad \text{and} \quad \varphi(\alpha, s_k(Y_1, \ldots, Y_k))
\]

where \( s_k \) is a structure function that aggregates first \( k \) datapoints

- \( s_k \) combines partial preferences into more complete estimates
What happened?

**Old surrogates:** \( \mathbb{E}[\varphi(\alpha, Y) \mid q] = \lim_{k \to \infty} \frac{1}{k} \sum_k \varphi(\alpha, Y_k) \)

- Loss \( \varphi(\alpha, Y) \) applied to a single datapoint

**New surrogates:** \( \varphi(\alpha, \mathbb{E}[Y \mid q]) = \lim_{k \to \infty} \varphi(\alpha, \frac{1}{k} \sum_k Y_k) \)

- Loss applied to aggregation of many datapoints

**New framework:** Ranking with aggregate losses

\[
L(\alpha, s_k(Y_1, \ldots, Y_k)) \quad \text{and} \quad \varphi(\alpha, s_k(Y_1, \ldots, Y_k))
\]

where \( s_k \) is a **structure function** that aggregates first \( k \) datapoints

- \( s_k \) combines partial preferences into more complete estimates
- Consistency characterization extends to this setting
Aggregation via structure function

Question: When does aggregation help?
Aggregation via structure function

\[ Y_1, Y_2, \ldots, Y_k \]

\[ s_k(Y_1, \ldots, Y_k) \]

**Question:** When does aggregation help?
Complete data losses

- Normalized Discounted Cumulative Gain (NDCG)
- Precision, Precision@$k$
- Expected reciprocal rank (ERR)

**Pros:** Popular, well-motivated, admit tractable consistent surrogates
- e.g., Penalize mistakes at top of ranked list more heavily
Complete data losses

- Normalized Discounted Cumulative Gain (NDCG)
- Precision, Precision@$k$
- Expected reciprocal rank (ERR)

**Pros:** Popular, well-motivated, admit tractable consistent surrogates
  - e.g., Penalize mistakes at top of ranked list more heavily

**Cons:** Require complete preference data
Complete data losses

- Normalized Discounted Cumulative Gain (NDCG)
- Precision, Precision@$k$
- Expected reciprocal rank (ERR)

**Pros:** Popular, well-motivated, admit tractable consistent surrogates
  - e.g., Penalize mistakes at top of ranked list more heavily

**Cons:** Require complete preference data

**Idea:**
- Use aggregation to estimate complete preferences from partial preferences
Complete data losses

- Normalized Discounted Cumulative Gain (NDCG)
- Precision, Precision@k
- Expected reciprocal rank (ERR)

**Pros:** Popular, well-motivated, admit tractable consistent surrogates
  - e.g., Penalize mistakes at top of ranked list more heavily

**Cons:** Require complete preference data

**Idea:**
- Use aggregation to estimate complete preferences from partial preferences
- Plug estimates into consistent surrogates
Complete data losses
- Normalized Discounted Cumulative Gain (NDCG)
- Precision, Precision@\(k\)
- Expected reciprocal rank (ERR)

**Pros:** Popular, well-motivated, admit tractable consistent surrogates
  - e.g., Penalize mistakes at top of ranked list more heavily

**Cons:** Require complete preference data

**Idea:**
- Use aggregation to estimate complete preferences from partial preferences
- Plug estimates into consistent surrogates
- Check that aggregation + surrogacy retains consistency
Cascade model for click data

[Craswell, Zoeter, Taylor, and Ramsey, 2008, Chapelle, Metzler, Zhang, and Grinspan, 2009]

- Person $i$ clicks on first relevant result, $k(i)$
Cascade model for click data

[Craswell, Zoeter, Taylor, and Ramsey, 2008, Chapelle, Metzler, Zhang, and Grinspan, 2009]

- Person $i$ clicks on first relevant result, $k(i)$
- Relevance probability of item $k$ is $p_k$

\[
p_{k} = \prod_{j=1}^{k-1} (1 - p_j)
\]

ERR loss assumes $p$ is known

Estimate $p$ via maximum likelihood on $n$ clicks:

\[
s = \arg\max_{p \in [0, 1]} \frac{1}{m} \sum_{i=1}^{n} \log p_k(i) + \sum_{j=1}^{k} \log(1 - p_j)
\]

⇒ Consistent ERR minimization under our framework
Cascade model for click data

[Craswell, Zoeter, Taylor, and Ramsey, 2008, Chapelle, Metzler, Zhang, and Grinspan, 2009]

- Person \( i \) clicks on first relevant result, \( k(i) \)
- Relevance probability of item \( k \) is \( p_k \)
- Probability of a click on item \( k \) is

\[
p_k \prod_{j=1}^{k-1} (1 - p_j)
\]
Cascade model for click data

[Craswell, Zoeter, Taylor, and Ramsey, 2008, Chapelle, Metzler, Zhang, and Grinspan, 2009]

- Person $i$ clicks on first relevant result, $k(i)$
- Relevance probability of item $k$ is $p_k$
- Probability of a click on item $k$ is

$$p_k \prod_{j=1}^{k-1}(1 - p_j)$$

- ERR loss assumes $p$ is known
Cascade model for click data

[Craswell, Zoeter, Taylor, and Ramsey, 2008, Chapelle, Metzler, Zhang, and Grinspan, 2009]

- Person $i$ clicks on first relevant result, $k(i)$
- Relevance probability of item $k$ is $p_k$
- Probability of a click on item $k$ is

$$p_k \prod_{j=1}^{k-1}(1 - p_j)$$

- ERR loss assumes $p$ is known

Estimate $p$ via maximum likelihood on $n$ clicks:

$$s = \arg\max_{p \in [0,1]^m} \sum_{i=1}^{n} \log p_{k(i)} + \sum_{j=1}^{k(i)} \log (1 - p_j).$$

⇒ **Consistent** ERR minimization under our framework
Benefits of aggregation

- Tractable consistency for partial preference losses

\[
\begin{align*}
\arg\min_f \lim_{k \to \infty} \mathbb{E}[\varphi(f(Q), s_k(Y_1, \ldots, Y_k))] \\
\Rightarrow \\
\arg\min_f \lim_{k \to \infty} \mathbb{E}[L(f(Q), s_k(Y_1, \ldots, Y_k))]
\end{align*}
\]

- Use complete data losses with realistic partial preference data
  - Models process of generating relevance scores from clicks/comparisons
What remains?

Before aggregation, we had

$$\arg\min_f \frac{1}{n} \sum_{k=1}^n \varphi(f(Q_k), Y_k) \rightarrow \arg\min_f \mathbb{E}[\varphi(f(Q), Y)]$$

\underline{empirical} \hspace{5cm} \underline{population}
What remains?

Before aggregation, we had

\[
\arg\min_f \frac{1}{n} \sum_{k=1}^{n} \varphi(f(Q_k), Y_k) \rightarrow \arg\min_f \mathbb{E}[\varphi(f(Q), Y)]
\]

What’s a suitable empirical analogue \( \hat{R}_{\varphi,n}(f) \) with aggregation?
What remains?

Before aggregation, we had

$$\arg\min_f \frac{1}{n} \sum_{k=1}^{n} \varphi(f(Q_k), Y_k) \rightarrow \arg\min_f \mathbb{E}[\varphi(f(Q), Y)]$$

empirical

population

What’s a suitable empirical analogue $\widehat{R}_{\varphi,n}(f)$ with aggregation?

$$\iff$$

When does

$$\arg\min_f \widehat{R}_{\varphi,n}(f) \rightarrow \arg\min_f \lim_{k \to \infty} \mathbb{E}[\varphi(f(Q), s_k(Y_1, \ldots, Y_k))]$$

empirical

population
Outline

Supervised Ranking
  Formal definition
  Tractable surrogates
  Pairwise inconsistency

Aggregation
  Restoring consistency
  Estimating complete preferences

U-statistics
  Practical procedures
  Experimental results
Datapoint consists of query $q$ and preference judgment $Y$

$n_q$ datapoints for query $q$

Structure functions for aggregation:

$$s(Y_1, Y_2, \ldots, Y_k)$$
Data with aggregation

- **Simple idea:** for query $q$, aggregate all $Y_1, Y_2, \ldots, Y_{n_q}$

- **Loss** $\varphi$ for query $q$ is

$$n_q \cdot \varphi(\alpha, s(Y_1, \ldots, Y_{n_q}))$$
Data with aggregation

Simple idea: for query $q$, aggregate all $Y_1, Y_2, \ldots, Y_{n_q}$

Loss $\varphi$ for query $q$ is

$$n_q \cdot \varphi(\alpha, s(Y_1, \ldots, Y_{n_q}))$$

Cons:

- Requires detailed knowledge of $\varphi$ and $s_k(Y_1, \ldots, Y_k)$ as $k \to \infty$
Data with aggregation

| \( q_1 \) | \( Y_1 \ Y_2 \ Y_3 \ \ldots \ n_{q_1} \) |
| \( q_2 \) | \( n_{q_2} \) |
| \( q_3 \) | \( n_{q_3} \) |
| \( q_4 \) | \( n_{q_4} \) |
| \( q_5 \) | \( n_{q_5} \) |

- **Simple idea:** for query \( q \), aggregate all \( Y_1, Y_2, \ldots, Y_{n_q} \)
- **Loss** \( \varphi \) for query \( q \) is
  \[
  n_q \cdot \varphi(\alpha, s(Y_1, \ldots, Y_{n_q}))
  \]

**Cons:**
- Requires detailed knowledge of \( \varphi \) and \( s_k(Y_1, \ldots, Y_k) \) as \( k \to \infty \)

**Ideal procedure:**
- Agnostic to form of aggregation
- Take advantage of independence of \( Y_1, Y_2, \ldots \)
Digression: $U$-statistics

$U$-statistic: classical tool in statistics

- Given $X_1, \ldots, X_n$, estimate $\mathbb{E}[g(X_1, \ldots, X_k)]$ for $g$ symmetric

- Idea: Average all estimates based on $k$ datapoints

\[
U_n = \left( \frac{n}{k} \right)^{-1} \sum_{i_1 < \cdots < i_k} g(X_{i_1}, X_{i_2}, \ldots, X_{i_k})
\]
Data with aggregation: $U$-statistic in the loss

Target: $\mathbb{E}[\varphi(\alpha, s(Y_1, \ldots, Y_k)) \mid q]$
Data with aggregation: $U$-statistic in the loss

- **Target:** $\mathbb{E}[\varphi(\alpha, s(Y_1, \ldots, Y_k)) \mid q]$  
- **Idea:** Estimate with $U$-statistic:

$$\left(\frac{n_q}{k}\right)^{-1} \sum_{i_1 < \ldots < i_k} \varphi(\alpha, s(Y_{i_1}, \ldots, Y_{i_k}))$$
Data with aggregation: $U$-statistic in the loss

- **Target:** $\mathbb{E}[\varphi(\alpha, s(Y_1, \ldots, Y_k)) | q]$

- **Idea:** Estimate with $U$-statistic:

\[
\left( \begin{array}{c} n_q \\ k \end{array} \right)^{-1} \sum_{i_1 < \cdots < i_k} \varphi(\alpha, s(Y_{i_1}, \ldots, Y_{i_k}))
\]

- **Empirical risk for scoring function $f$:**

\[
\hat{R}_{\varphi,n}(f) = \frac{1}{n} \sum_{q} n_q \left( \begin{array}{c} n_q \\ k \end{array} \right)^{-1} \sum_{i_1 < \cdots < i_k} \varphi(f(q), s(Y_{i_1}, \ldots, Y_{i_k}))
\]
Empirical risk for scoring function $f$:

$$\hat{R}_{\varphi,n}(f) = \frac{1}{n} \sum_q n_q \left( \frac{n_q}{k} \right)^{-1} \sum_{i_1 < \cdots < i_k} \varphi(f(q), s(Y_{i_1}, \ldots, Y_{i_k}))$$

**Theorem:** If we choose $k_n = o(n)$ but $k_n \to \infty$, then uniformly in $f$

$$\hat{R}_{\varphi,n}(f) \to \lim_{k \to \infty} \mathbb{E}[\varphi(f(Q), s(Y_1, \ldots, Y_k))]$$

**Limiting aggregated loss**
New procedure for learning to rank

1. Use loss function that aggregates *per-query*:
   \[
   \hat{R}_{\varphi,n}(f) = \frac{1}{n} \sum_q n_q \left( \frac{n_q}{k} \right)^{-1} \sum_{i_1 < \cdots < i_k} \varphi(f(q), s(Y_{i_1}, \ldots, Y_{i_k}))
   \]

2. Learn ranking function by taking
   \[
   \hat{f} \in \arg\min_{f \in \mathcal{F}} \hat{R}_{\varphi,n}(f)
   \]

3. Can optimize by stochastic gradient descent over queries \( q \) and subsets \( (i_1, \ldots, i_k) \)
Experiments

- Web search
- Image ranking
Web search

- Microsoft Learning to Rank Web10K dataset
Web search

- Microsoft Learning to Rank Web10K dataset
  - 10,000 queries issued
  - 100 items per query
  - Estimated relevance score $r \in \mathbb{R}$ for each query/result pair

Generating pairwise preferences

- Choose query $q$ uniformly at random
- Choose pair $(i,j)$ of items, and set $i \succ j$ with probability $p_{ij} = \frac{1}{1 + \exp(r_j - r_i)}$

Aggregate scores by setting $s_i = \sum_{j \neq i} \log \frac{\hat{P}(j \prec i)}{\hat{P}(i \prec j)}$
Web search

- Microsoft Learning to Rank Web10K dataset
  - 10,000 queries issued
  - 100 items per query
  - Estimated relevance score $r \in \mathbb{R}$ for each query/result pair
- Generating pairwise preferences
  - Choose query $q$ uniformly at random
  - Choose pair $(i, j)$ of items, and set $i \succ j$ with probability

$$p_{ij} = \frac{1}{1 + \exp(r_j - r_i)}$$
Web search

- Microsoft Learning to Rank Web10K dataset
  - 10,000 queries issued
  - 100 items per query
  - Estimated relevance score $r \in \mathbb{R}$ for each query/result pair

- Generating pairwise preferences
  - Choose query $q$ uniformly at random
  - Choose pair $(i,j)$ of items, and set $i \succ j$ with probability
    $$ p_{ij} = \frac{1}{1 + \exp(r_j - r_i)} $$

- Aggregate scores by setting
  $$ s_i = \sum_{j \neq i} \log \frac{\hat{P}(j \prec i)}{\hat{P}(i \prec j)} $$
Benefits of aggregation

NDCG risk as a function of aggregation level $k$
for $n = 10^6$ samples

<table>
<thead>
<tr>
<th>Order $k$</th>
<th>NDCG@10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aggregate</td>
<td>0.65</td>
</tr>
<tr>
<td>Pairwise</td>
<td>0.7</td>
</tr>
<tr>
<td>Score-based</td>
<td>0.8</td>
</tr>
</tbody>
</table>
Image ranking

- **Setup** [Grangier and Bengio 2008]
  - Take most common image search queries on google.com
  - Train an independent ranker based on aggregated preference statistics for each query
  - Compare with standard, disaggregated image-ranking approaches
Image ranking experiments

Highly ranked items from Corel Image Database for query tree car:

Aggregated

SVM

PLSA
Conclusions

1. Partial preference data is abundant and (more) reliable

2. General theory of ranking consistency: When is
   \[ \arg\min_f E[\phi(f(Q,s))] \subseteq \arg\min_f E[L(f(Q),s)] \]?
   ▶ Tractable consistency difficult with partial preference data
   ▶ Possible with complete preference data

3. Aggregation can bridge the gap
   ▶ Can transform pairwise preferences/click data into scores

4. Practical consistent procedures via \( U \)-statistic aggregation
   ▶ Allows for arbitrary aggregation
   ▶ High-probability convergence of the learned ranking function
Conclusions

1. Partial preference data is abundant and (more) reliable

Tractable consistency difficult with partial preference data
Possible with complete preference data

Aggregation can bridge the gap
Can transform pairwise preferences/click data into scores

Practical consistent procedures via $U$-statistic aggregation
Allows for arbitrary aggregation
High-probability convergence of the learned ranking function
Conclusions

1. Partial preference data is abundant and (more) reliable
2. General theory of ranking consistency: When is

\[
\arg\min_f \mathbb{E}[\varphi(f(Q), s)] \subseteq \arg\min_f \mathbb{E}[L(f(Q), s)]?
\]

- Tractable consistency difficult with partial preference data
- Possible with complete preference data
Conclusions

1. Partial preference data is abundant and (more) reliable
2. General theory of ranking consistency: When is

\[ \arg\min_f \mathbb{E}[\varphi(f(Q), s)] \subseteq \arg\min_f \mathbb{E}[L(f(Q), s)]? \]

- Tractable consistency difficult with partial preference data
- Possible with complete preference data
3. Aggregation can bridge the gap
- Can transform pairwise preferences/click data into scores \( s \)
Conclusions

1. Partial preference data is abundant and (more) reliable
2. General theory of ranking consistency: When is

$$\arg\min_{f} \mathbb{E}[\varphi(f(Q), s)] \subseteq \arg\min_{f} \mathbb{E}[L(f(Q), s)]?$$

- Tractable consistency difficult with partial preference data
- Possible with complete preference data

3. Aggregation can bridge the gap
   - Can transform pairwise preferences/click data into scores $s$

4. Practical consistent procedures via $U$-statistic aggregation
   - Allows for arbitrary aggregation $s$
   - High-probability convergence of the learned ranking function
Future work

- Empirical directions
  - Apply to more ranking problems!
  - Which aggregation procedures perform best?
  - How much aggregation is enough?
- Statistical questions: beyond consistency
  - How does aggregation impact rate of convergence?
  - Can we design statistically efficient ranking procedures?
- Other ways of dealing with realistic partial preference data?
Future work

- Empirical directions
  - Apply to more ranking problems!
  - Which aggregation procedures perform best?
  - How much aggregation is enough?
Future work

- **Empirical directions**
  - Apply to more ranking problems!
  - Which aggregation procedures perform best?
  - How much aggregation is enough?

- **Statistical questions: beyond consistency**
  - How does aggregation impact rate of convergence?
  - Can we design statistically efficient ranking procedures?
Future work

- **Empirical directions**
  - Apply to more ranking problems!
  - Which aggregation procedures perform best?
  - How much aggregation is enough?

- **Statistical questions: beyond consistency**
  - How does aggregation impact rate of convergence?
  - Can we design statistically efficient ranking procedures?

- Other ways of dealing with realistic partial preference data?


What is the problem?

Surrogate loss \( \varphi(\alpha, s) = \sum_{ij} s_{ij} \phi(\alpha_i - \alpha_j) \)

\[
\begin{align*}
1 & \rightarrow 2 \\
& \quad \downarrow s_{12} \quad s_{13} \\
& \quad \downarrow s_{23} \\
& \quad \rightarrow 3
\end{align*}
\]

\[ p(s) = .5 \]

\[
\begin{align*}
1 & \rightarrow 2 \\
& \quad \downarrow s_{31} \\
& \quad \rightarrow 3
\end{align*}
\]

\[ p(s') = .5 \]

Aggregate

\[
\begin{align*}
1 & \rightarrow 2 \\
& \quad \downarrow s_{12} \quad s_{31} \\
& \quad \downarrow s_{23} \\
& \quad \rightarrow 3 \\
& \quad \downarrow s_{13}
\end{align*}
\]
What is the problem?

Surrogate loss $\varphi(\alpha, s) = \sum_{ij} s_{ij} \phi(\alpha_i - \alpha_j)$

\[ p(s) = .5 \quad p(s') = .5 \]

Aggregate

\[ \sum_s p(s) \varphi(\alpha, s) = \frac{1}{2} \varphi(\alpha, s') + \frac{1}{2} \varphi(\alpha, s') \]

\[ \propto s_{12} \phi(\alpha_1 - \alpha_2) + s_{13} \phi(\alpha_1 - \alpha_3) + s_{23} \phi(\alpha_2 - \alpha_3) + s_{31} \phi(\alpha_3 - \alpha_1) \]
What is the problem?

\[ s_{12}\phi(\alpha_1 - \alpha_2) + s_{13}\phi(\alpha_1 - \alpha_3) + s_{23}\phi(\alpha_2 - \alpha_3) + s_{31}\phi(\alpha_3 - \alpha_1) \]
What is the problem?

\[ s_{12} \phi(\alpha_1 - \alpha_2) + s_{13} \phi(\alpha_1 - \alpha_3) + s_{23} \phi(\alpha_2 - \alpha_3) + s_{31} \phi(\alpha_3 - \alpha_1) \]
What is the problem?

\[ s_{12}\phi(\alpha_1 - \alpha_2) + s_{13}\phi(\alpha_1 - \alpha_3) + s_{23}\phi(\alpha_2 - \alpha_3) + s_{31}\phi(\alpha_3 - \alpha_1) \]
What is the problem?

\[ s_{12}\phi(\alpha_1 - \alpha_2) + s_{13}\phi(\alpha_1 - \alpha_3) + s_{23}\phi(\alpha_2 - \alpha_3) + s_{31}\phi(\alpha_3 - \alpha_1) \]

More bang for your $$ by increasing to 0 from left: \alpha_1 \downarrow. \text{ Result:} \]

\[ \alpha^* = \arg\min_\alpha \sum_{ij} s_{ij}\phi(\alpha_i - \alpha_j) \]

can have \( \alpha_2^* > \alpha_1^* \), even if \( s_{13} - s_{31} > s_{12} + s_{23} \).