AN INTRODUCTION TO PREFERENCE LEARNING

Eyke Hüllermeier
Department of Mathematics and Computer Science
Marburg University, Germany
AGENDA

1. Introduction
   - An example: The Choquet integral
   - Preference modeling, elicitation, and learning
   - Preference Learning: A first glimpse
   - Machine Learning

2. Ranking Problems

3. Model-based Preference Learning

4. Summary & Outlook
## AGGREGATION OF CRITERIA

<table>
<thead>
<tr>
<th>$x_1$</th>
<th>Math</th>
<th>CS</th>
<th>Statistics</th>
<th>English</th>
<th>score</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>16</td>
<td>17</td>
<td>12</td>
<td>19</td>
<td>0.7</td>
</tr>
<tr>
<td>$x_2$</td>
<td>10</td>
<td>12</td>
<td>18</td>
<td>9</td>
<td>0.4</td>
</tr>
<tr>
<td>$x_3$</td>
<td>19</td>
<td>10</td>
<td>18</td>
<td>10</td>
<td>0.6</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>$x_n$</td>
<td>8</td>
<td>18</td>
<td>10</td>
<td>18</td>
<td>0.5</td>
</tr>
</tbody>
</table>
Non-additive measure (capacity) \( \mu : 2^X \rightarrow [0, 1] \):

- \( \mu(\emptyset) = 0, \mu(X) = 1 \)
- \( \mu(A) \leq \mu(B) \) for \( A \subset B \subset X \)
- \( \mu(A \cup B) = \mu(A) + \mu(B) \) for \( A \cap B = \emptyset \)

We require...
- normalization
- monotonicity
- not necessarily additivity
Non-additive measures allow for modeling interaction between criteria:

- Positive (synergy): \(\mu(A \cup B) > \mu(A) + \mu(B)\)
- Negative (redundancy): \(\mu(A \cup B) < \mu(A) + \mu(B)\)

In a machine learning context: criteria = attributes/features

\(\mu(A) = \text{joint importance} \) of the feature subset \(A\)

\(\neq\) sum of individual importance degrees
The **discrete Choquet integral** of $f : X \to \mathbb{R}_+$ with respect to $\mu$ is defined as follows:

$$C_\mu(f) = \sum_{i=1}^{m} \left( f(x_{(i)}) - f(x_{(i-1)}) \right) \cdot \mu(A_{(i)}),$$

where $(\cdot)$ is a permutation of $\{1, \ldots, m\}$ such that $0 \leq f(x_{(1)}) \leq f(x_{(2)}) \leq \ldots \leq f(x_{(m)})$, and $A_{(i)} = \{x_{(i)}, \ldots, x_{(m)}\}$.

**Special cases:**
- min
- max
- weighted mean (additive measure)
- OWA
\[
C_\mu(f) = \sum_{i=1}^{4} w_i \cdot f(c_i) = \sum_{i=1}^{4} \mu(\{c_i\}) \cdot f(c_i)
\]

\[
C_\mu(f) = \sum_{i=1}^{4} \mu(A_{(i)}) \cdot \left( f(c_{(i)}) - f(c_{(i-1)}) \right)
\]
DECISION BOUNDARY IN TWO DIMENSIONS

\[(x, y) \mapsto \mathbb{I}\left(\alpha x + (1 - \alpha)y > c\right)\]
(x, y) \mapsto \mathbb{1}\left(\min(x, y) > c\right)
DECISION BOUNDARY IN TWO DIMENSIONS

\[(x, y) \mapsto \mathbb{I}\left(\max(x, y) > c\right)\]
DECISION BOUNDARY IN TWO DIMENSIONS
DECISION BOUNDARY IN TWO DIMENSIONS
DECISION BOUNDARY IN TWO DIMENSIONS
1. Introduction
   - An example: The Choquet integral
   - Preference modeling, elicitation, and learning
   - Preference Learning: A first glimpse
   - Machine Learning
2. Ranking Problems
3. Model-based Preference Learning
4. Summary & Outlook
CONDITIONAL PREFERENCE NETWORKS:
Compact representation of a partial order relation, exploiting conditional independence of preferences on attribute values.

meat: red wine > white wine
veggie: red wine > white wine
fish: white wine > red wine

meat: Italian > Chinese
veggie: Chinese > Italian
fish: Chinese > Italian
Preference modeling: The model is completely specified by an expert or the decision maker himself.

- Example Choquet integral: Normalization of features, specification of fuzzy measure, ...
- Example CP network: Qualitative structure, conditional preferences, ...

Preference elicitation/learning: Parts of the model are specified with the help of external information.
- typically no user interaction
- holistic judgements
- fixed preferences but noisy data
- regularized models
- weak model assumptions, flexible (instead of axiomatically justified) model classes
- diverse types of training information
- computational aspects: massive data, scalable methods
- focus on predictive accuracy (expected loss)

MACHINE LEARNING

computer science
artificial intelligence

Preference Learning
Preference Elicitation

PREFERENCE MODELING and DECISION ANALYSIS

operations research
social sciences (voting and choice theory)
economics and decision theory
TRAINING DATA:

\[
(18, 17, 12, 10) \succ (16, 19, 10, 10) \\
(17, 12, 18, 12) \succ (15, 14, 16, 11) \\
(12, 19, 18, 18) \succ (16, 16, 15, 17) \\
\ldots \succ \ldots
\]

The goal might be to find a Choquet integral whose utility degrees tend to agree with the observed pairwise preferences!
ORDINAL CLASSIFICATION / SORTING

TRAINING DATA:

\[
\begin{align*}
(12, 17, 11, 8) & \rightarrow ** \\
(19, 15, 17, 16) & \rightarrow *** \\
(9, 12, 14, 10) & \rightarrow * \\
\ldots & \rightarrow \ldots
\end{align*}
\]

The goal might be to find a Choquet integral whose utility degrees tend to agree with the observed classifications!
TRAINING DATA:

\[
\begin{align*}
(12, 17, 11, 8) & \rightarrow 0.6 \\
(19, 15, 17, 16) & \rightarrow 0.9 \\
(9, 12, 14, 10) & \rightarrow 0.2 \\
\ldots & \rightarrow \ldots
\end{align*}
\]

The goal might be to find a Choquet integral whose utility degrees tend to agree with the observed scores!
AGENDA

1. Introduction
   - An example: The Choquet integral
   - Preference modeling, elicitation, and learning
   - **Preference Learning: A first glimpse**
   - Machine learning

2. Ranking Problems

3. Model-based Preference Learning

4. Summary & Outlook

21
PREFERENCES ARE UBIQUITOUS

---

9 of 10 people found the following review helpful:

⭐⭐⭐⭐⭐ A wonderful textbook for machine learning over the web,
September 8, 2004

By Ari Rappoport - See all my reviews

This review is from: Mining the Web: Discovering Knowledge from Hypertext Data (Hardcover)

This book is one of the best computer science textbooks I have ever seen. Apart from the wealth of information and discussion on specific WEB crawling and data mining (chapters 2, 3, 7, 8), chapters 4, 5 and 6 constitute a wonderful summary of machine learning in general.

The book's discussion of unsupervised learning (the EM algorithm, advanced algorithms in which the number of clusters is not known in advance), supervised learning (Bayesian networks, entropian methods, SVMs), semisupervised learning, co-training and rule induction is extraordinary in that it is short, intuitive, does not sacrifice mathematical rigor, and accompanied by examples (all taken from information retrieval over the web).

Help other customers find the most helpful reviews Report this Permalink

Was this review helpful to you? Yes No

---

Mozilla Firefox for Windows 3.6

Download Now

File size: 9.15 MB

PC Compatibility?

This browser provides pop-up control and context sensitive page controls that can be used in your theme.

---

Epinions.com

Popular

Rating

Now Playing DIVINE (Midnight Juggernauts Remix) Sebastien Tellier

---

T-Meter Critics Top Critics RT Community My Critics My Friends DVD

84%

How does the Tomatometer work

Consensus: Hal Ashby's comedy is too dark and twisted for some, and occasionally oversteps its bounds, but there's no denying the film's warm humor and big heart.

Like 70 people like this.
PREFERENCES ARE UBQUITOUS

<table>
<thead>
<tr>
<th>Offizielle Homepage</th>
<th>Daniel Baier</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="http://www.daniel-baier.com/">www.daniel-baier.com/</a></td>
<td></td>
</tr>
</tbody>
</table>
Willkommen auf der offiziellen Homepage von Fußballprofi Daniel Baier - TSV 1860 München.

Prof. Dr. Daniel Baier - Brandenburgische Technische Universität ...
www.tu-cottbus.de/fakultaet3/de/.../team/.../prof-dr-daniel-baier.html
Vökler, Sascha; Krausche, Daniel; Baier, Daniel: Product Design Optimization Using Ant Colony And Bee Algorithms: A Comparison, erscheint in: Studies in ...

Daniel Baier
www.welfussball.de/spieler_profil/daniel-baier/

Daniel Baier - aktuelle Themen & Nachrichten - sueddeutsche.de
www.sueddeutsche.de/thema/Daniel_Baier
Aktuelle Nachrichten, Informationen und Bilder zum Thema Daniel Baier auf sueddeutsche.de.

Daniel Baier | Facebook
de-de.facebook.com/daniel.baier.589
Tritt Facebook bei, um dich mit Daniel Baier und anderen Nutzern, die du kennst, zu vernetzen. Facebook ermöglicht den Menschen das Teilen von Inhalten mit ...

FC Augsburg: Mein Tag in Bad Gögging: Daniel Baier
www.fcaugsburg.de/cms/website.php?id=/index/aktuell/news/...
2. Aug. 2012 – Daniel Baier berichtet heute, was für die Profis auf dem Programm stand. Hi FCA-Fans, heute liegen wieder zwei intensive Trainingseinheiten ...
Companies (and agencies) are collecting lots of data ...

... hoping to discover something useful in it!
Neues Patent: Amazon will schon vor der Bestellung liefern


Noch bevor ein Kunde überhaupt den Button "Kaufen" anklickt, soll die für ihn passende Ware schon auf dem Weg in Richtung seiner Wohnung sein. Dem Versandhändler Amazon wurde ein Patent (PDF) zugesprochen, das einen "vorausschauenden Versand" ("anticipatory shipping") ermöglichen soll. Das heißt: Bestimmte Waren werden schon einmal an ein Versandzentrum geschickt, in dessen Nähe sich ein oder mehrere Kunden höchstwahrscheinlich für das Produkt interessieren. Wird es dann schließlich bestellt, ist es umso schneller beim Empfänger.

Wie Amazon das herausfinden will, erklärt das "Wall Street Journal": Ausgewertet werden könnten demnach frühere Bestellungen, Umtäusche, Wunschzettel bei Amazon, der Inhalt der Einkaufswagen - und sogar, wie lange ein Kunde mit dem Mauszeiger auf einer Produktbeschreibung verweilt.
Fostered by the availability of large amounts of data, **PREFERENCE LEARNING** has recently emerged as a new subfield of machine learning, dealing with the learning of (predictive) preference models from observed, revealed or automatically extracted preference information.
PREFERENCE LEARNING IS AN ACTIVE FIELD

- NIPS–01: New Methods for Preference Elicitation
- NIPS–02: Beyond Classification and Regression: Learning Rankings, Preferences, Equality Predicates, and Other Structures
- KI–03: Preference Learning: Models, Methods, Applications
- NIPS–04: Learning with Structured Outputs
- NIPS–05: Workshop on Learning to Rank
- IJCAI–05: Advances in Preference Handling
- SIGIR 07–10: Workshop on Learning to Rank for Information Retrieval
- **ECML/PDKK 08–10: Workshop on Preference Learning**
- NIPS–09: Workshop on Advances in Ranking
- American Institute of Mathematics Workshop in Summer 2010: The Mathematics of Ranking
- NIPS-11: Workshop on Choice Models and Preference Learning
- EURO-12: Special Track on Preference Learning
- **ECAI-12: Workshop on Preference Learning: Problems and Applications in AI**
PREFERENCE LEARNING IS AN ACTIVE FIELD

Tutorials:
- European Conf. on Machine Learning, 2010
- Int. Conf. Discovery Science, 2011
- Int. Conf. Algorithmic Decision Theory, 2011
- European Conf. on Artificial Intelligence, 2012

Special Issue on Representing, Processing, and Learning Preferences: Theoretical and Practical Challenges (2011)

J. Fürnkranz & E. Hüllermeier (eds.) Preference Learning Springer-Verlag 2011

Special Issue on Preference Learning Forthcoming
User preferences play a key role in various fields of application:

- **Computational Advertising**
- **Recommender Systems**
- **Computer Games**
- **Electronic Commerce**
- **Adaptive Retrieval Systems**
- **Autonomous Agents**

**Preferences in AI Research:**

- **Preference representation** (CP nets, GAU networks, logical representations, fuzzy constraints, ...)
- **Reasoning** with preferences (decision theory, constraint satisfaction, non-monotonic reasoning, ...)
- **Preference acquisition** (preference elicitation, preference learning, ...)
preferences learning settings

- **binary vs. graded** (e.g., relevance judgements vs. ratings)
- **absolute vs. relative** (e.g., assessing single alternatives vs. comparing pairs)
  - **explicit vs. implicit** (e.g., direct feedback vs. click-through data)
  - **structured vs. unstructured** (e.g., ratings on a given scale vs. free text)
- **single user vs. multiple users** (e.g., document keywords vs. social tagging)
- **single vs. multi-dimensional**

A wide spectrum of learning problems!
Preference learning problems can be distinguished along several problem dimensions, including

- **representation of preferences, type of preference model:**
  - utility function (ordinal, numeric),
  - preference relation (partial order, ranking, ...),
  - logical representation, ...

- **description of individuals/users and alternatives/items:**
  - identifier, feature vector, structured object, ...

- **type of training input:**
  - direct or indirect feedback,
  - complete or incomplete relations,
  - utilities, ...

- ...
PREFERENCE LEARNING

Preferences

assessing

absolute

binary

A B C D
1 1 0 0

numeric

A B C D
.9 .8 .1 .3

gradual

ordinal

A B C D
+ + -- 0

comparing

relative

total order

A > B > C > D

partial order

A ≈ B ≈ C ≈ D

→ (ordinal) regression

→ classification/ranking
OBJECT RANKING [Cohen et al., 1999]

TRAINING

(0.74, 1, 25, 165) \succ (0.45, 0, 35, 155)
(0.47, 1, 46, 183) \succ (0.57, 1, 61, 177)
(0.25, 0, 26, 199) \succ (0.73, 0, 46, 185)

Pairwise preferences between objects
OBJECT RANKING [Cohen et al. 99]

PREDICTION (ranking a new set of objects)

\[ Q = \{ x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}, x_{12}, x_{13} \} \]

\[ x_{10} \succ x_4 \succ x_7 \succ x_1 \succ x_{11} \succ x_2 \succ x_8 \succ x_{13} \succ x_9 \succ x_3 \succ x_{12} \succ x_5 \succ x_6 \]
**Collaborative Filtering** [Goldberg et al., 1992]

<table>
<thead>
<tr>
<th>Users</th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>...</th>
<th>P38</th>
<th>...</th>
<th>P88</th>
<th>P89</th>
<th>P90</th>
</tr>
</thead>
<tbody>
<tr>
<td>U1</td>
<td>⭐️</td>
<td>⭐️⭐⭐⭐⭐</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⭐️⭐⭐⭐⭐</td>
<td></td>
<td></td>
</tr>
<tr>
<td>U2</td>
<td>⭐⭐⭐</td>
<td>⭐⭐</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⭐️</td>
<td></td>
<td></td>
</tr>
<tr>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td></td>
<td></td>
</tr>
<tr>
<td>U46</td>
<td>⭐️</td>
<td>⭐⭐⭐</td>
<td>⭐️</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
</tr>
<tr>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td></td>
<td></td>
</tr>
<tr>
<td>U98</td>
<td>⭐⭐⭐</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⭐⭐⭐⭐⭐</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>U99</td>
<td>⭐⭐</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⭐⭐⭐⭐⭐</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## PREFERENCE LEARNING TASKS

<table>
<thead>
<tr>
<th>OBJECT RANKING</th>
<th>COLLABORATIVE FILTERING</th>
</tr>
</thead>
<tbody>
<tr>
<td>product description</td>
<td>features</td>
</tr>
<tr>
<td>preference description</td>
<td>relative</td>
</tr>
<tr>
<td>predictions</td>
<td>ranking</td>
</tr>
<tr>
<td>number of users/models</td>
<td>single</td>
</tr>
<tr>
<td></td>
<td>identifier</td>
</tr>
<tr>
<td></td>
<td>absolute</td>
</tr>
<tr>
<td></td>
<td>utility degrees</td>
</tr>
<tr>
<td></td>
<td>many</td>
</tr>
</tbody>
</table>
AGENDA

1. Introduction
   - An example: The Choquet integral
   - Preference modeling, elicitation, and learning
   - Preference Learning: A first glimpse
   - Machine Learning

2. Ranking Problems
3. Model-based Preference Learning
4. Summary & Outlook
SUPERVISED LEARNING: Algorithms and methods for discovering (alleged) dependencies and regularities in a domain of interest, expressed through appropriate models, from specific observations or examples.
What kind of **training data** is offered to the learning algorithm?

What **type of model** (prediction) is the learner supposed to produce?

\[ h : \mathcal{X} \rightarrow \mathcal{Y} \]

What is the nature of the **ground truth**, and how is a model assessed?

\[ \mathcal{L} : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}_+ \]

\[(y, y^*) \mapsto \text{penalty for predicting } y \text{ if the true outcome is } y^*\]
**SPECIFICATION OF A MACHINE LEARNING PROBLEM**

- What kind of **training data** is offered to the learning algorithm?

- What **type of model** (prediction) is the learner supposed to produce?

  \[ h : \mathcal{X} \rightarrow \mathcal{Y} \]

- What is the nature of the **ground truth**, and how is a model assessed?

  \[ \mathcal{R}(h) = \int_{\mathcal{X} \times \mathcal{Y}} \mathcal{L}(h(x), y) \, d\mathbf{P}(X, Y) \]

  - **risk**: average penalty caused by the model’s predictions
  - **unknown data-generating process**

- Other criteria, such as complexity...
A simple setting of supervised learning: Given (i.i.d.) training data

\[ D = \{ (x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \} \subset (\mathcal{X} \times \mathcal{Y})^n \]

and a hypothesis space \( \mathcal{H} \subset \{ h : \mathcal{X} \to \mathcal{Y} \} \), induce a model

\[ h^* \in \arg\min_{h \in \mathcal{H}} \int_{\mathcal{X} \times \mathcal{Y}} \ell(h(x), y) \ d P(X, Y) \]

Many other settings exist, such as online learning, semi-supervised learning, active learning, multi-task learning, etc.
\( \mathcal{X} \subseteq \mathbb{R}^d, \quad \mathcal{Y} = \{y_1, y_2, \ldots, y_k\}, \)

\[
\ell(\hat{y}, y) = \begin{cases} 
0 & \hat{y} = y \\
1 & \hat{y} \neq y
\end{cases}
\]

\[
\mathcal{H} = \left\{ \mathbf{x} \mapsto \mathbb{I}(\mathbf{x}^\top \alpha + \beta > 0) \mid \alpha \in \mathbb{R}^d, \beta \in \mathbb{R} \right\}
\]
A key to successful learning/generalization is a proper **capacity control**: The model class must be flexible enough (to allow approximation of the pointwise loss-minimizer) but not too flexible (to prevent overfitting the data).

Typical bound on the true risk: With probability $1 - \delta$

$$
R(h) \leq R_{emp}(h) + \sqrt{\frac{2 \cdot \text{VC}(\mathcal{H}) \log \left( \frac{2e|\mathcal{D}|}{\text{VC}(\mathcal{H})} \right) + \log \left( \frac{2}{\delta} \right)}{|\mathcal{D}|}}
$$

true risk  empirical risk  correction depending on capacity and sample size
CHOICE OF THE MODEL SPACE

INPUT

OUTPUT
CHOICE OF THE MODEL SPACE

underfitting
CHOICE OF THE MODEL SPACE

overfitting

OUTPUT

INPUT
overfitting and underfitting leads to poor generalization
**LOGISTIC REGRESSION**

- **Logistic regression** modifies linear regression for the purpose of predicting (probabilities of) a **binary class label** instead of real-valued responses.

- The basic model:

\[
\log \left( \frac{P(y = 1 | x)}{P(y = 0 | x)} \right) = w_0 + \sum_{i=1}^{m} w_i \cdot x_i \\
= w_0 + \mathbf{w}^\top \mathbf{x},
\]

where

- \( \mathbf{x} = (x_1, x_2, \ldots, x_m)^\top \in \mathbb{R}^m \) is an instance to be classified,

- \( \mathbf{w} = (w_1, w_2, \ldots, w_m)^\top \in \mathbb{R}^m \) is a vector of regression coefficients,

- \( w_0 \in \mathbb{R} \) is a constant bias (the intercept).
LOGISTIC REGRESSION: CLASS PREDICTION

- Equivalently, this can be expressed in terms of posterior probabilities:

\[
P(y = 1 \mid x) = \left( 1 + \exp(-w_0 - \mathbf{w}^\top \mathbf{x}) \right)^{-1}
\]

\[
P(y = 0 \mid x) = 1 - P(y = 1 \mid x)
\]

- Predictions are typically made using the following decision rule:

\[
\hat{y} = \begin{cases} 
0 & \text{if } P(y = 1 \mid x) < 1/2 \\
1 & \text{if } P(y = 1 \mid x) \geq 1/2 
\end{cases}
\]
LOGISTIC REGRESSION: PARAMETER ESTIMATION

- The parameters of the model (bias, regression coefficients) can be obtained through **Maximum Likelihood (ML)** estimation.
- Given a sample of i.i.d. data

\[
D = \left\{ (x^{(i)}, y^{(i)}) \right\}_{i=1}^{n} \subset (\mathbb{R}^m \times \{0, 1\})^n, 
\]

the likelihood function is given by

\[
\prod_{i=1}^{n} P \left( y = y^{(i)} \mid x^{(i)} \right),
\]

and the **ML estimate** is the maximizer of (the log of) this function:

\[
(\hat{w}_0, \hat{w}) = \arg \max_{(w_0, w)} \sum_{i=1}^{n} y^{(i)} \log \theta^{(i)}(w_0, w) + (1 - y^{(i)}) \log \left( 1 - \theta^{(i)}(w_0, w) \right)
\]

with

\[
\theta^{(i)}(w_0, w) = \left( 1 + \exp(-w_0 - w^\top x^{(i)}) \right)^{-1}
\]
• Logistic regression is very popular and widely used in practice.

• It is comprehensible and easy to interpret, especially since the influence of each variable can easily be captured from the model:

\[
\log \left( \frac{P(y = 1 \mid x)}{P(y = 0 \mid x)} \right) = w_0 + w_1 \cdot x_1 + w_2 \cdot x_2 + \ldots + w_m \cdot x_m
\]

direction and strength of influence of the first variable on the log-odds ratio (probability of positive class)
AGENDA

1. Introduction

2. Ranking Problems
   - Label ranking
   - Object ranking
   - Instance ranking

3. Model-based Preference Learning

4. Summary & Outlook
LABEL RANKING

... mapping instances to **TOTAL ORDERS** over a fixed set of alternatives/labels:

\[(28, 0, 187, 0.4) \rightarrow \rightarrow \rightarrow\]

instance \(x \in \mathcal{X}\) (e.g., features of a person)

ranking of labels/alternatives
\[
\mathcal{Y} = \{y_1, y_2, \ldots, y_k\}
\]
\[
\mathcal{Y} = \{A, B, C, \ldots\}
\]
LABEL RANKING

... mapping instances to **TOTAL ORDERS** over a fixed set of alternatives/labels:

\[(28, 0, 187, 0.4) \rightarrow \text{ROME} \succ \text{PARIS} \succ \text{LONDON} \]

instance \( x \in \mathcal{X} \)

(e.g., features of a person)

ranking of labels/alternatives

\[\mathcal{Y} = \{y_1, y_2, \ldots, y_k\}\]

\[\mathcal{Y} = \{A, B, C, \ldots\}\]
THE SUSHI DATA

Rankings of 10 types of sushi by 5000 customers. Each customer is characterized by 11 features.
Collected by Kamishima et al., reprocessed by Grbovic.

http://www.kamishima.net/sushi/
**LABEL RANKING: TRAINING DATA**

**TRAINING**

<table>
<thead>
<tr>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>preferences</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.34</td>
<td>0</td>
<td>10</td>
<td>174</td>
<td>$A \succ B, C \succ D$</td>
</tr>
<tr>
<td>1.45</td>
<td>0</td>
<td>32</td>
<td>277</td>
<td>$B \succ C \succ A$</td>
</tr>
<tr>
<td>1.22</td>
<td>1</td>
<td>46</td>
<td>421</td>
<td>$B \succ D, A \succ D, C \succ D, A \succ C$</td>
</tr>
<tr>
<td>0.74</td>
<td>1</td>
<td>25</td>
<td>165</td>
<td>$C \succ A \succ D, A \succ B$</td>
</tr>
<tr>
<td>0.95</td>
<td>1</td>
<td>72</td>
<td>273</td>
<td>$B \succ D, A \succ D$</td>
</tr>
<tr>
<td>1.04</td>
<td>0</td>
<td>33</td>
<td>158</td>
<td>$D \succ A \succ B, C \succ B, A \succ C$</td>
</tr>
</tbody>
</table>

Instances are associated with preferences between labels.

... no demand for full rankings!
# Label Ranking: Prediction

<table>
<thead>
<tr>
<th>PREDICTION</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.92</td>
<td>?</td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>1</td>
<td>?</td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>81</td>
<td>?</td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>382</td>
<td>?</td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
</tbody>
</table>

new instance ranking ?
## Label Ranking: Prediction

<table>
<thead>
<tr>
<th>PREDICTION</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.92</td>
<td>4</td>
<td>1</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>81</td>
<td>382</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**new instance**

$$\pi(i) = \text{position of } i\text{-th label}$$

A ranking of all labels
**LABEL RANKING: PREDICTION**

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>PREDICTION</td>
<td>0.92</td>
<td>1</td>
<td>81</td>
</tr>
<tr>
<td>GROUND TRUTH</td>
<td>0.92</td>
<td>1</td>
<td>81</td>
</tr>
</tbody>
</table>

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>4</td>
<td>1</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>4</td>
</tr>
</tbody>
</table>

A ranking of all labels

**SPEARMAN**

\[
\mathcal{L}(\pi, \pi^*) = \sum_{i=1}^{k} (\pi(i) - \pi^*(i))^2
\]

**LOSS**

\[
\rho = 1 - \frac{6D(\pi, \pi^*)}{k(k^2 - 1)}
\]

**RANK CORRELATION**
**LABEL RANKING: PREDICTION**

**PREDICTION**

<table>
<thead>
<tr>
<th>0.92</th>
<th>1</th>
<th>81</th>
<th>382</th>
<th>4</th>
<th>1</th>
<th>3</th>
<th>2</th>
</tr>
</thead>
</table>

**GROUND TRUTH**

| 0.92 | 1 | 81 | 382 | 2 | 1 | 3 | 4 |

**KENDALL**

\[
\mathcal{L}(\pi, \pi^*) = \sum_{1 \leq i < j \leq k} \left[ (\pi(i) - \pi(j))(\pi^*(i) - \pi^*(j)) < 0 \right]
\]

**Loss**

\[
\tau = 1 - \frac{4D(\pi, \pi^*)}{k(k - 1)}
\]

**Rank Correlation**

A ranking of all labels
LEARNING TECHNIQUES

How to learn a label ranker \( h : \mathcal{X} \to S_k \) ?

The output space is complex ...
Our output space is the class of permutations (symmetric group):

Symmetric group $S_3$

- 123
- 213
- 231
- 312
- 321

a single inversion
Symmetric group $S_4$
Treating each permutation as a (meta) class and applying polychotomous classification methods is obviously a bad idea:

- too many classes
- no exploitation of the structure of the space
- ...
LEARNING TECHNIQUES

How to learn a label ranker $h : X \rightarrow S_k$?

DIFFERENT APPROACHES:

- Reduction to simpler problems (binary classification)
  Transform the problem, so as to make it amenable to standard ML algorithms.

- Extension of (classification) algorithms
  Generalize standard ML algorithms, so as to make them applicable to label ranking data.

- Probabilistic modeling and statistical inference
  Make use of statistical models for rank data and parameter estimation methods.
RANKING BY PAIRWISE COMPARISON

Ranking by Pairwise Comparison (RPC) trains models

\[ M_{i,j} : \mathcal{X} \rightarrow [0, 1] \quad (1 \leq i < j \leq k) \]

Given a query instance \( x \), \( M_{i,j} \) is supposed to predict the probability that \( y_i \succ y_j \):

\[ M_{i,j}(x) = P(y_i \succ y_j) = 1 - P(y_j \succ y_i) \]

\( \rightarrow \) decomposition into \( k(k - 1)/2 \) binary classification problems
## Ranking by Pairwise Comparison

Training data (for the label pair A and B):

<table>
<thead>
<tr>
<th>X1</th>
<th>X2</th>
<th>X3</th>
<th>X4</th>
<th>preferences</th>
<th>class</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.34</td>
<td>0</td>
<td>10</td>
<td>0.34</td>
<td>0 10 174</td>
<td>1</td>
</tr>
<tr>
<td>1.45</td>
<td>0</td>
<td>32</td>
<td>1.22</td>
<td>1  46 421</td>
<td>0</td>
</tr>
<tr>
<td>1.22</td>
<td>1</td>
<td>46</td>
<td>0.74</td>
<td>1  25 165</td>
<td>1</td>
</tr>
<tr>
<td>0.74</td>
<td>1</td>
<td>25</td>
<td>1.04</td>
<td>0  33 158</td>
<td>1</td>
</tr>
<tr>
<td>0.95</td>
<td>1</td>
<td>72</td>
<td>1.04</td>
<td>0  33 158</td>
<td>1</td>
</tr>
</tbody>
</table>

For the label pair A and C:

<table>
<thead>
<tr>
<th>X1</th>
<th>X2</th>
<th>X3</th>
<th>X4</th>
<th>preferences</th>
<th>class</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.04</td>
<td>0</td>
<td>33</td>
<td>158</td>
<td>D &gt; A, A &gt; B, C &gt; B, A &gt; C</td>
<td>1</td>
</tr>
</tbody>
</table>

---

A: A, B: B, C: C, D: D
At prediction time, a query instance is submitted to all models, and the predictions are combined into a binary preference relation

\[ P(i, j) = \begin{cases} 
M_{i,j}(x), & i < j \\
1 - M_{i,j}(x), & i > j 
\end{cases} \]

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.3</td>
<td>0.8</td>
<td>0.4</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>0.7</td>
<td>0.7</td>
<td>0.9</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>0.2</td>
<td>0.3</td>
<td>0.3</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>0.6</td>
<td>0.1</td>
<td>0.7</td>
<td></td>
</tr>
</tbody>
</table>

How to produce a ranking on the basis of this preference relation?
Recall our original goal

\[ R(h) = \int_{X \times Y} \mathcal{L}(h(x), y) \, dP(X, Y) \rightarrow \min \]

and our representation:

\[ h = \text{AGG} \left( \mathcal{M}_{1,2}, \mathcal{M}_{1,3}, \ldots, \mathcal{M}_{k-1,k} \right) \]

**Loss decomposition problem:** Is it possible to find a suitable loss \( \mathcal{L}_p \), to be minimized (in expectation) by the pairwise learners, and an aggregation function \( \text{AGG} \), such that \( h = \text{AGG}(\mathcal{M}_{1,2}, \ldots, \mathcal{M}_{k-1,k}) \) minimizes \( \mathcal{L} \) (in expectation)?
## MINIMIZING SPEARMAN LOSS

### Predictions $\mathcal{M}_{i,j}(x)$

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.3</td>
<td>0.8</td>
<td>0.4</td>
<td>1.5</td>
</tr>
<tr>
<td>B</td>
<td>0.7</td>
<td>0.7</td>
<td>0.9</td>
<td>2.3</td>
</tr>
<tr>
<td>C</td>
<td>0.2</td>
<td>0.3</td>
<td>0.3</td>
<td>0.8</td>
</tr>
<tr>
<td>D</td>
<td>0.6</td>
<td>0.1</td>
<td>0.7</td>
<td>1.4</td>
</tr>
</tbody>
</table>

The order based on loss is $B \succ A \succ D \succ C$. 

*Note: Spearman loss minimization focuses on ranking agreement.*
Theorem: Suppose the pairwise learners $\mathcal{M}_{i,j}$ yield unbiased probability estimates and let $\pi$ be a ranking such that

$$\left( \sum_q P(i, q) > \sum_q P(j, q) \right) \Rightarrow (\pi(i) < \pi(j)).$$

Then $\pi$ minimizes risk w.r.t. to the Spearman loss

$$\mathcal{L}(\pi, \pi^*) = \sum_{i=1}^{k} (\pi(i) - \pi^*(i))^2.$$
**Theorem:** Risk w.r.t. Kendall loss

\[ \mathcal{L}(\pi, \pi^*) = \sum_{1 \leq i < j \leq k} \left[ (\pi(i) - \pi(j))(\pi^*(i) - \pi^*(j)) < 0 \right] \]

is minimized by

\[ \pi^* = \arg \min_{\pi} \sum_{1 \leq i < j \leq k} \mathcal{P}(\pi^{-1}(j), \pi^{-1}(i)) \]

\[ \rightarrow \text{linear ordering problem for weighted tournaments} \]
MINIMIZING KENDALL LOSS

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.3</td>
<td>0.8</td>
<td>0.4</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>0.7</td>
<td>0.7</td>
<td>0.9</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>0.2</td>
<td>0.3</td>
<td>0.3</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>0.6</td>
<td>0.1</td>
<td>0.7</td>
<td></td>
</tr>
</tbody>
</table>
MINIMIZING KENDALL LOSS

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.3</td>
<td>0.8</td>
<td>0.4</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>0.7</td>
<td>0.7</td>
<td>0.9</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>0.2</td>
<td>0.3</td>
<td>0.3</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>0.6</td>
<td>0.1</td>
<td>0.7</td>
<td></td>
</tr>
</tbody>
</table>
## Minimizing Kendall Loss

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.3</td>
<td>0.8</td>
<td>0.4</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>0.7</td>
<td>0.7</td>
<td>0.9</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>0.2</td>
<td>0.3</td>
<td>0.3</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>0.6</td>
<td>0.1</td>
<td>0.7</td>
<td></td>
</tr>
</tbody>
</table>

![Diagram showing relationships between A, B, C, D with Kendall loss values](attachment:diagram.png)
# Minimizing Kendall Loss

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.3</td>
<td>0.8</td>
<td>0.4</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>0.7</td>
<td>0.7</td>
<td>0.9</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>0.2</td>
<td>0.3</td>
<td>0.3</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>0.6</td>
<td>0.1</td>
<td>0.7</td>
<td></td>
</tr>
</tbody>
</table>

Order: B A D C
**MINIMIZING KENDALL LOSS**

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.3</td>
<td>0.8</td>
<td>0.4</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>0.7</td>
<td></td>
<td>0.7</td>
<td>0.9</td>
</tr>
<tr>
<td>C</td>
<td>0.2</td>
<td>0.3</td>
<td></td>
<td>0.3</td>
</tr>
<tr>
<td>D</td>
<td>0.6</td>
<td>0.1</td>
<td>0.7</td>
<td></td>
</tr>
</tbody>
</table>

Order:

- B
- A
- D
- C

*forward arcs*

**Diagram**
MINIMIZING KENDALL LOSS

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.3</td>
<td>0.8</td>
<td>0.4</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>0.7</td>
<td></td>
<td>0.7</td>
<td>0.9</td>
</tr>
<tr>
<td>C</td>
<td>0.2</td>
<td>0.3</td>
<td></td>
<td>0.3</td>
</tr>
<tr>
<td>D</td>
<td>0.6</td>
<td>0.1</td>
<td>0.7</td>
<td></td>
</tr>
</tbody>
</table>

Order:

B → A → D → C

Forward arcs:
- B to A: 0.7
- A to D: 0.8
- D to C: 0.7

Backward arcs:
- B to D: 0.3
- C to B: 0.3

COST:

0.3 + 0.2 + 0.3 +
0.3 + 0.6 + 0.1 = 1.8
LIMITATIONS OF RPC

Proposition: For the following losses, RPC can not guarantee a risk minimizing prediction:

- 0/1 loss
  \[ \mathcal{L}(\pi, \pi^*) = \left[ \pi \neq \pi^* \right] \]

- Hamming distance
  \[ \mathcal{L}(\pi, \pi^*) = \sum_{i=1}^{k} \left[ \pi(i) \neq \pi^*(i) \right] \]

- Cayley distance (minimal number of transpositions of any pair of labels needed to turn the first ranking into the second one)

- Ulam distance (minimal number of position changes of labels needed to turn the first ranking into the second one)
RANKING BY PAIRWISE COMPARISON [E.H. et al., 2008]

Train

\[ \mathcal{T}_{1,2} \rightarrow \mathcal{M}_{1,2} \]
\[ \mathcal{T}_{1,3} \rightarrow \mathcal{M}_{1,3} \]
\[ \mathcal{T}_{1,4} \rightarrow \mathcal{M}_{1,4} \]
\[ \mathcal{T}_{2,3} \rightarrow \mathcal{M}_{2,3} \]
\[ \mathcal{T}_{2,4} \rightarrow \mathcal{M}_{2,4} \]
\[ \mathcal{T}_{3,4} \rightarrow \mathcal{M}_{3,4} \]

Test

\[ \mathcal{M}_{1,2} \]
\[ \mathcal{M}_{1,3} \]
\[ \mathcal{M}_{1,4} \]
\[ \mathcal{M}_{2,3} \]
\[ \mathcal{M}_{2,4} \]
\[ \mathcal{M}_{3,4} \]

Decom-position

Binary Classification

RANKING

\[ \mathcal{P} \rightarrow \pi \]

Binary Prediction
LEARNING TECHNIQUES

How to learn a label ranker $h : \mathcal{X} \rightarrow \mathcal{S}_k$?

**DIFFERENT APPROACHES:**

- Reduction to simpler problems (binary classification)
  
  *Transform the problem, so as to make it amenable to standard ML algorithms.*

- Extension of (classification) algorithms
  
  *Generalize standard ML algorithms, so as to make them applicable to label ranking data.*

- **Probabilistic modeling and statistical inference**
  
  *Make use of statistical models for rank data and parameter estimation methods.*
**PROBABILISTIC LABEL RANKER**

<table>
<thead>
<tr>
<th>permutation</th>
<th>probability</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.jpg" alt="Permutation 1" /></td>
<td>0.2</td>
</tr>
<tr>
<td><img src="image2.jpg" alt="Permutation 2" /></td>
<td>0</td>
</tr>
<tr>
<td><img src="image3.jpg" alt="Permutation 3" /></td>
<td>0</td>
</tr>
<tr>
<td><img src="image4.jpg" alt="Permutation 4" /></td>
<td>0.1</td>
</tr>
</tbody>
</table>

**Need a parametrized family of distributions on the permutation space!**
... is a **stagewise** model specified by a vector \( \mathbf{v} = (v_1, v_2, \ldots, v_k) \in \mathbb{R}^k_+ \):

\[
P(\pi \mid \mathbf{v}) = \prod_{i=1}^{k} \frac{v_{\pi^{-1}(i)}}{v_{\pi^{-1}(i)} + v_{\pi^{-1}(i+1)} + \cdots + v_{\pi^{-1}(k)}}
\]

A ranking is produced by choosing labels one by one, with a probability proportional to their respective „skills“.
THE PLACKETT-LUCE MODEL

\[
v_\bullet = 10, \quad v_\cdot = 6, \quad v_\circ = 4
\]

\[
P(\bullet, \cdot, \circ) = \quad
\]
THE PLACKETT-LUCE MODEL

\[ \nu_1 = 10, \quad \nu_2 = 6, \quad \nu_3 = 4 \]

\[ P(\ , \ , \ ) = \frac{6}{20} \times \]
THE PLACKETT-LUCE MODEL

\[ v_1 = 10, \quad v_2 = 6, \quad v_3 = 4 \]

\[
P(\text{ }, \text{ }, \text{ }) = \frac{6}{20} \times \frac{10}{14} \times \text{ } \]
The Plackett-Luce Model

\[ v_1 = 10, \quad v_2 = 6, \quad v_3 = 4 \]

\[ P(\text{red}, \text{green}, \text{blue}) = \frac{6}{20} \times \frac{10}{14} \times \frac{4}{4} = \frac{3}{14} \]
THE PLACKETT-LUCE MODEL

<table>
<thead>
<tr>
<th></th>
<th>FIRST</th>
<th>SECOND</th>
<th>THIRD</th>
</tr>
</thead>
<tbody>
<tr>
<td>3/14</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6/70</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3/10</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1/5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3/40</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1/8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

First: 3/14, 6/70, 3/10, 1/5, 3/40, 1/8
Second: 3/14, 3/10, 1/5, 3/40, 1/8
Third: 3/14, 6/70, 3/10, 1/5, 3/40, 1/8
Observations are not complete rankings such as

\[ \pi : B \succ C \succ A \succ D \]

but \textbf{pairwise preferences} like

\[ \sigma : D \succ C \]

or \textbf{incomplete rankings} like

\[ \sigma : B \succ D \succ A \]

Given a probability $P(\cdot)$ on $S_k$, the probability of an **incomplete ranking** $\sigma$ is given by the probability of its linear extensions:

$$P(\sigma) = P(E(\sigma)) = \sum_{\pi \in E(\sigma)} P(\pi)$$
### Probability of Incomplete Rankings

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>0.14</td>
</tr>
<tr>
<td>A</td>
<td>B</td>
<td>D</td>
<td>C</td>
<td>0.00</td>
</tr>
<tr>
<td>A</td>
<td>C</td>
<td>B</td>
<td>D</td>
<td>0.08</td>
</tr>
<tr>
<td>A</td>
<td>C</td>
<td>D</td>
<td>B</td>
<td>0.01</td>
</tr>
<tr>
<td>A</td>
<td>D</td>
<td>B</td>
<td>C</td>
<td>0.10</td>
</tr>
<tr>
<td>A</td>
<td>D</td>
<td>C</td>
<td>B</td>
<td>0.10</td>
</tr>
<tr>
<td>B</td>
<td>A</td>
<td>C</td>
<td>D</td>
<td>0.00</td>
</tr>
<tr>
<td>B</td>
<td>A</td>
<td>D</td>
<td>C</td>
<td>0.05</td>
</tr>
<tr>
<td>B</td>
<td>C</td>
<td>A</td>
<td>D</td>
<td>0.00</td>
</tr>
<tr>
<td>B</td>
<td>C</td>
<td>D</td>
<td>A</td>
<td>0.00</td>
</tr>
<tr>
<td>B</td>
<td>D</td>
<td>A</td>
<td>C</td>
<td>0.15</td>
</tr>
<tr>
<td>B</td>
<td>D</td>
<td>C</td>
<td>A</td>
<td>0.00</td>
</tr>
<tr>
<td>C</td>
<td>A</td>
<td>B</td>
<td>D</td>
<td>0.00</td>
</tr>
<tr>
<td>C</td>
<td>A</td>
<td>D</td>
<td>B</td>
<td>0.03</td>
</tr>
<tr>
<td>C</td>
<td>B</td>
<td>A</td>
<td>D</td>
<td>0.00</td>
</tr>
<tr>
<td>C</td>
<td>B</td>
<td>D</td>
<td>A</td>
<td>0.16</td>
</tr>
<tr>
<td>C</td>
<td>D</td>
<td>A</td>
<td>B</td>
<td>0.00</td>
</tr>
<tr>
<td>C</td>
<td>D</td>
<td>B</td>
<td>A</td>
<td>0.00</td>
</tr>
<tr>
<td>D</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>0.00</td>
</tr>
<tr>
<td>D</td>
<td>A</td>
<td>C</td>
<td>B</td>
<td>0.02</td>
</tr>
<tr>
<td>D</td>
<td>B</td>
<td>A</td>
<td>C</td>
<td>0.00</td>
</tr>
<tr>
<td>D</td>
<td>B</td>
<td>C</td>
<td>A</td>
<td>0.17</td>
</tr>
<tr>
<td>D</td>
<td>C</td>
<td>A</td>
<td>B</td>
<td>0.00</td>
</tr>
<tr>
<td>D</td>
<td>C</td>
<td>B</td>
<td>A</td>
<td>0.09</td>
</tr>
</tbody>
</table>

\[
P(A \succeq C) =
\]
PROBABILITY OF INCOMPLETE RANKINGS

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>0.14</td>
</tr>
<tr>
<td>A</td>
<td>B</td>
<td>D</td>
<td>C</td>
<td>0.00</td>
</tr>
<tr>
<td>A</td>
<td>C</td>
<td>B</td>
<td>D</td>
<td>0.08</td>
</tr>
<tr>
<td>A</td>
<td>C</td>
<td>D</td>
<td>B</td>
<td>0.00</td>
</tr>
<tr>
<td>A</td>
<td>D</td>
<td>B</td>
<td>C</td>
<td>0.10</td>
</tr>
<tr>
<td>A</td>
<td>D</td>
<td>C</td>
<td>B</td>
<td>0.00</td>
</tr>
<tr>
<td>B</td>
<td>A</td>
<td>C</td>
<td>D</td>
<td>0.00</td>
</tr>
<tr>
<td>B</td>
<td>A</td>
<td>D</td>
<td>C</td>
<td>0.05</td>
</tr>
<tr>
<td>B</td>
<td>C</td>
<td>A</td>
<td>D</td>
<td>0.00</td>
</tr>
<tr>
<td>B</td>
<td>C</td>
<td>D</td>
<td>A</td>
<td>0.00</td>
</tr>
<tr>
<td>B</td>
<td>D</td>
<td>A</td>
<td>C</td>
<td>0.15</td>
</tr>
<tr>
<td>B</td>
<td>D</td>
<td>C</td>
<td>A</td>
<td>0.00</td>
</tr>
<tr>
<td>C</td>
<td>A</td>
<td>B</td>
<td>D</td>
<td>0.00</td>
</tr>
<tr>
<td>C</td>
<td>A</td>
<td>D</td>
<td>B</td>
<td>0.03</td>
</tr>
<tr>
<td>C</td>
<td>B</td>
<td>A</td>
<td>D</td>
<td>0.00</td>
</tr>
<tr>
<td>C</td>
<td>B</td>
<td>D</td>
<td>A</td>
<td>0.16</td>
</tr>
<tr>
<td>C</td>
<td>D</td>
<td>A</td>
<td>B</td>
<td>0.00</td>
</tr>
<tr>
<td>C</td>
<td>D</td>
<td>B</td>
<td>A</td>
<td>0.00</td>
</tr>
<tr>
<td>D</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>0.00</td>
</tr>
<tr>
<td>D</td>
<td>A</td>
<td>C</td>
<td>B</td>
<td>0.02</td>
</tr>
<tr>
<td>D</td>
<td>B</td>
<td>A</td>
<td>C</td>
<td>0.00</td>
</tr>
<tr>
<td>D</td>
<td>B</td>
<td>C</td>
<td>A</td>
<td>0.17</td>
</tr>
<tr>
<td>D</td>
<td>C</td>
<td>A</td>
<td>B</td>
<td>0.00</td>
</tr>
<tr>
<td>D</td>
<td>C</td>
<td>B</td>
<td>A</td>
<td>0.09</td>
</tr>
</tbody>
</table>

\[ P(A \succeq C) = 0.54 \]
The probability to observe a set of (incomplete) rankings $\mathcal{D} = \{\sigma_n\}_{n=1}^N$, assuming independence, is

$$P(\mathcal{D}) = \prod_{n=1}^{N} P(E(\sigma_n))$$

The **likelihood** of a set of parameter values is the probability of the data under these values:

$$L(\nu) = P(\mathcal{D} \mid \nu) = \prod_{n=1}^{N} P_{\nu}(E(\sigma_n))$$

Maximum Likelihood (ML) Inference
Since rankings $\pi$ are “contextualized” by instances $x$, we need to model $P(\pi | x)$ instead of $P(\pi)$.

Assuming the PL model, this can be done by expressing $\nu = (\nu_1, \ldots, \nu_k)$ as function of $x$:  

$$v_i = f_i(x) = f_i(x_1, \ldots, x_m)$$
Assuming $\mathbf{x} = (x_1, \ldots, x_m) \in \mathbb{R}^m$, the $v_i$ can be expressed as log-linear functions:

$$v_i = f_i(\mathbf{x}) = \exp \left( \sum_{j=1}^{m} \alpha_j^{(i)} \cdot x_j \right)$$

$\rightarrow$ estimation of parameter set $\left\{ \alpha_j^{(i)} | 1 \leq i \leq k, 1 \leq j \leq m \right\}$

$\rightarrow$ label ranking model defined by $k \cdot m$ real parameters
Given training data \( \mathcal{D} = \{(\mathbf{x}^{(n)}, \pi^{(n)})\}_{n=1}^{N} \) with \( \mathbf{x}^{(n)} = (x_1^{(n)}, \ldots, x_m^{(n)}) \), the log-likelihood is given by

\[
\ell = \sum_{n=1}^{N} \left[ \sum_{j=1}^{K_n} \log \left( v(\pi^{(n)}(j), n) \right) - \log \sum_{k=j}^{K_n} v(\pi^{(n)}(k), n) \right],
\]

where \( K_n \) is the number of labels in the ranking \( \pi^{(n)} \), and

\[
v(j, n) = \exp \left( \sum_{i=1}^{m} \alpha_m^{(j)} \cdot x_m^{(n)} \right).
\]

Algorithm based on MM (minorization and maximization) construction principle [Hunter 2004].
THE SUSHI DATA

Rankings of 10 types of sushi by 5000 customers. Each customer is characterized by 11 features. Collected by Kamishima et al., reprocessed by Grbovic.

http://www.kamishima.net/sushi/
EXPERIMENTAL STUDIES

![Graph showing Kendall tau values for Plackett-Luce, PRC (LR), and Constraint Classification.](Image)
SELECTED LITERATURE