A New Rule-based Method for Label Ranking

M. Gurrieri\textsuperscript{a}    P. Fortemps\textsuperscript{a}    Xavier Siebert\textsuperscript{a}
Salvatore Greco\textsuperscript{b}    Roman Słowiński\textsuperscript{c}

\textsuperscript{a} UMONS, Faculté Polytechnique, Mons, Belgium
\textsuperscript{b} Faculty of Economics, University of Catania, Italy
\textsuperscript{c} Institute of Computing Science, Poznan University of Technology, Poland

July 10th, 2012
## Label Ranking (toy example)

<table>
<thead>
<tr>
<th>Customers</th>
<th>Customers’ preferences on books</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer 1</td>
<td>Fiction ≻ History ≻ Business ≻ Hobbies</td>
</tr>
<tr>
<td>Customer 2</td>
<td>Business ≻ History ≻ Fiction ≻ Hobbies</td>
</tr>
<tr>
<td>Customer 3</td>
<td>Hobbies ≻ Fiction ≻ Business ≻ History</td>
</tr>
<tr>
<td>Customer 4</td>
<td>History ≻ Business ≻ Hobbies ≻ Fiction</td>
</tr>
<tr>
<td>NEW Customer</td>
<td>?</td>
</tr>
</tbody>
</table>

**How to predict the final order of books for a new customer?**
### Label Ranking (toy example)

#### Customers’ Profile

<table>
<thead>
<tr>
<th>Profile</th>
<th>Fiction</th>
<th>History</th>
<th>Business</th>
<th>Hobbies</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5 0.3 -2.1 -0.5</td>
<td>1 2</td>
<td>3 4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-0.2 0.1 2.7 1.5</td>
<td>3 2</td>
<td>1 4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.5 0.4 0.1 0.7</td>
<td>2 4</td>
<td>3 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-1.5 1.3 2.4 -1.6</td>
<td>4 1</td>
<td>2 3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.3 -0.3 0.1 0.1</td>
<td>? ?</td>
<td>? ?</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- A feature vector (net salary, height,...)
- Position of each label in the ranking (linear order)
Label Ranking (more formally)

Input:
- An instance space $X$
- A set of labels $L = \{\lambda_1; \lambda_2; \ldots; \lambda_k\}$
- $\forall x \in X$ a total order $\succ_x: X \rightarrow L$ in the form of $\lambda_i \succ_x \lambda_j$

Output:
- A map $\Gamma : X \rightarrow \Omega_L$ (permutation space of $\{1, 2, \ldots, k\}$)
- Equivalent to the search of a directed acyclic preference graph:

$$x \rightarrow \lambda_{\pi_x^{-1}(1)} \succ_x \lambda_{\pi_x^{-1}(2)} \succ_x \ldots \succ_x \lambda_{\pi_x^{-1}(k)}$$
Label Ranking (more formally)

Some measures of the prediction quality:

- **Kendall’s tau**:
  \[
  \tau_k = \frac{n_c - n_d}{k(k-1) \frac{2}{2}}
  \]

- **Spearman’s rank correlation**:
  \[
  \rho = 1 - \frac{6 \sum_{i=1}^{k} (\pi_i - \pi'_i)^2}{k(k^2 - 1)}
  \]
Adaptation of existing classification algorithms:

- Decision Trees & Instance-based for label ranking
  - Cheng & Hüllermeier, 2009
- Log-linear models for label ranking
  - Dekel et al., 2003
- APRIORI-LR
  - Sá et al., 2011
- Label Ranking Support Vector Machine
  - Shwartz and Singer, 2003
Label ranking is reduced to binary classification:

- Ranking by Pairwise Comparison
  - Hüllermeier and Fürnkranz, 2008
- Constraints Classification
  - Har-Peled et al., 2002
Most of existing approaches:

- Work as **oracles** or **black boxes**:  
  - Lack of interpretability on results
- Provide **local models** (lazy methods):  
  - Cannot provide description of the model (i.e. not suitable for *association analysis*)
- Do not treat preference information **at once**:
  - There could be loss of information
1. Label Ranking
   - An example
   - More formally
   - Approaches

2. Classification Rules
   - Definition
   - Rough Sets

3. A Rule-based Method
   - Preprocessing
   - Rules Generation
   - Classification
   - Ranking
   - Results

4. Experimental Results
   - Data Sets
   - Set-up

5. Conclusion
Classification rules are sentences of the form:

\[ \Phi \rightarrow \Psi \]

Let \( H, K \subseteq X \) be the subsets verifying \( \Phi \) and \( \Psi \)

Some interestingness measures:

- **Support** \( S = |H \cap K| \)
- **Confidence** \( c = \frac{|H \cap K|}{|H|} \)
- **Strength** \( s = \frac{|H \cap K|}{|X|} \)
Rough Sets approach (RSA): **approximation** of a crisp set (lower, upper approximations)

- **Lower approximation**
  \[ B(Cl_t) = \{ x \in U : I_B(x) \subseteq Cl_t \} \]
  - generating **certain rules**

- **Upper approximation**
  \[ \overline{B}(Cl_t) = \bigcup_{x \in Cl_t} I_B(x) \]
  - generating **possible rules**

- **Boundary**
  \[ BN_B(Cl_t) = B(Cl_t) - \overline{B}(Cl_t) \]
  - generating **approximated rules**
Dominance-based rough set approach (DRSA) and Variable-consistency DRSA (VCDRSA):
- ordered classes
- criteria instead of simple attributes
- inconsistencies w.r.t. the dominance principle

Some rule learners based on RSA:
- AQ, LEM2, MLEM2, DomLEM, VC-DomLEM
  - based on a sequential covering strategy
  - heuristically generate a minimal set of rules
How to use a **rule learner** without treating a permutation as a class?

How to use **pairwise information** without independent binary classifiers?

How to take into account **criteria**?
A learning reduction technique is applied:

\[(x, \pi_x) = \{q_1, q_2, \ldots, q_l, \pi_x\}\]

\[\downarrow\]

\[\{x_{1,2}, x_{1,3}, \ldots, x_{i,j}, \ldots\}\]

- \(x_{i,j} = (q_{1}^{\geq}, q_{1}^{\leq}, q_{2}^{\geq}, q_{2}^{\leq}, \ldots, q_{i}^{\geq}, q_{i}^{\leq}, (\lambda_i, \lambda_j), d)\)
- \(i, j \in \{1, 2, \ldots, k\}, i < j\) and \(d \in \{GT, LT\}\)

VC-DRSA as rule-learner:

\[[(q_{r_1}^{\geq} \geq h_1) \text{ and } (q_{r_2}^{\leq} \leq h_2) \text{ and } \ldots] \text{ and } (\lambda_i, \lambda_j) \rightarrow d\]
Certain, possible and approximate rules
Minimal and non-redundant set of decision rules
Complexity of VC-DomLEM: polynomial
The Training Process: Inferring Rules
Scores

For a given unknown instance \( x' \) (testing instance), by using **interestingness** measures of rules (support, strength,...):

\[
\forall (\lambda_i, \lambda_j) \rightarrow \Gamma^+_{(i,j)}, \Gamma^-_{(i,j)}
\]

- \( \Gamma^+_{(i,j)} = \Gamma^-_{(j,i)} \)
- \( \Gamma^+_{(i,j)}, \Gamma^-_{(i,j)} \in [0, 1] \)
- \( \Gamma^+_{(i,j)} + \Gamma^-_{(i,j)} = 1 \)
- \( \Gamma^+_{(i,j)} \approx P(\lambda_i \succ_{x'} \lambda_j) \)
- \( \Gamma^-_{(i,j)} \approx P(\lambda_j \succ_{x'} \lambda_i) \)
For an unknown instance \( x' \), the decision about a pair \( (\lambda_i, \lambda_j) \) is given by:

\[
\text{If } \Gamma^+_{(i,j)} = \Gamma^-_{(i,j)} = 0 : \\
\quad d(x') = \begin{cases} 
  \text{GT} & \text{if } W^+_{(i,j)} \geq W^-_{(i,j)}; \\
  \text{LT} & \text{otherwise}
\end{cases}
\]

\[
\text{Else:} \\
\quad d(x') = \begin{cases} 
  \text{GT} & \text{if } \Gamma^+_{(i,j)} \geq \Gamma^-_{(i,j)}; \\
  \text{LT} & \text{otherwise}
\end{cases}
\]
Another strategy is based on the **One Nearest Neighbor**:

\[ \text{If } \Gamma^+_{(i,j)} = \Gamma^-_{(i,j)} = 0 : \]

\[ d(x') \rightarrow \text{nearest neighbor (with same scores)} \]

Else:

\[ d(x') = \begin{cases} 
GT & \text{if } \Gamma^+_{(i,j)} \geq \Gamma^-_{(i,j)}; \\
LT & \text{otherwise}
\end{cases} \]
For an unknown instance \( x' \), a set of pairwise preferences is provided.

Each pair of labels is associated with two scores: \( \Gamma^+(i,j) \), \( \Gamma^-(i,j) \).

Preference relation: total, asymmetric, irreflexive but, in general, **not transitive** (i.e. cycles are likely to happen).
The final ranking (preferences aggregation) is obtained by using a **Net Flow Scores** procedure:

$$S(\lambda_i) = \sum_{j \neq i} (\Gamma^+_{(i,j)} - \Gamma^-_{(i,j)})$$

- $\Gamma^+_{(i,j)} = \Gamma^-_{(j,i)}$
- $\Gamma^+_{(i,j)}, \Gamma^-_{(i,j)} \in [0, 1]$
- $\Gamma^+_{(i,j)} + \Gamma^-_{(i,j)} = 1$

The final ranking is obtained by ordering labels according to these scores.
Ranking Generation Process

\[ S(L_1) = 0.67 - 0.33 + 0.11 - 0.89 = -0.44 \]

\[ S(L_2) = 0.33 - 0.67 + 0.55 - 0.45 = -0.24 \]

\[ S(L_3) = 0.89 - 0.11 + 0.45 - 0.55 = 0.68 \]

\[ L_3 > L_2 > L_1 \]
WHAT ABOUT USING ANOTHER BINARY CLASSIFIER?
A learning reduction technique is applied:

\[(x, \pi_x) = \{q_1, q_2, \ldots, q_l, \pi_x\}\]

\[\downarrow\]

\[\{x_{1,2}, x_{1,3}, \ldots, x_{i,j}, \ldots\}\]

- \(x_{i,j} = (q_1, q_2, \ldots, q_l, (\lambda_i, \lambda_j), d)\)
- \(i, j \in \{1, 2, \ldots, k\}, i < j\) and \(d \in \{LT, GT\}\) or \(d \in \{0, 1\}\)

Any binary classifier:

- \(\Gamma^+_{(i,j)} = P(\lambda_i \succ_x \lambda_j)\)
- \(\Gamma^-_{(i,j)} = P(\lambda_j \succ_x \lambda_i)\)

The net flow score is given by:

\[S(i) = \sum_{j \neq i}(P(\lambda_i \succ_x \lambda_j) - P(\lambda_j \succ_x \lambda_i))\]
1. Label Ranking
   - An example
   - More formally
   - Approaches

2. Classification Rules
   - Definition
   - Rough Sets

3. A Rule-based Method

4. Experimental Results
   - Data Sets
   - Set-up
   - Results

5. Conclusion
Data sets used in our experiment were taken from KEBI Data Repository \(^1\) in the Philipps University of Marburg

<table>
<thead>
<tr>
<th>DATA SETS</th>
<th>#Instances</th>
<th>#Labels</th>
<th>#Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Glass</td>
<td>214</td>
<td>6</td>
<td>9</td>
</tr>
<tr>
<td>Iris</td>
<td>150</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Vehicle</td>
<td>846</td>
<td>4</td>
<td>18</td>
</tr>
<tr>
<td>Vowel</td>
<td>528</td>
<td>11</td>
<td>10</td>
</tr>
<tr>
<td>Wine</td>
<td>178</td>
<td>3</td>
<td>13</td>
</tr>
</tbody>
</table>

\(^1\) see http://www.uni-marburg.de/fb12/kebi/research/repository
Evaluation measures: *Kendall’s tau* and the *Spearman Rank Correlation coefficient*

Three different configurations:
- Basic version (**RBLR**): VC-DomLEM, certain rules, consistency level 0.98, majority class strategy
- One Nearest Neighbor (**RBLR+**): VC-DomLEM, certain rules, consistency level 0.98, ONN class strategy
- Standard binary classifier (**BCLR**): Multilayer perceptron
Other methods: pairwise comparison (RPC), constraints classification (CC), log-linear (LL), association rules for label ranking (ARLR), instance based learning (IBLR), decision tree for label ranking (LRT)

A cross validation study with 10-fold, 5 repeats
## Results

<table>
<thead>
<tr>
<th>Kendall’s Tau</th>
<th>RBLR</th>
<th>RBLR+</th>
<th>BCLR</th>
<th>ARLR</th>
<th>RPC</th>
<th>CC</th>
<th>LL</th>
<th>IBLR</th>
<th>LRT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Glass</td>
<td>.882(3)</td>
<td>.906(1)</td>
<td>.863(4)</td>
<td>.850(5)</td>
<td>.882(3)</td>
<td>.846(6)</td>
<td>.817(8)</td>
<td>.841(7)</td>
<td>.883(2)</td>
</tr>
<tr>
<td>Iris</td>
<td>.956(4)</td>
<td>.961(2)</td>
<td>.971(1)</td>
<td>.960(3)</td>
<td>.885(6)</td>
<td>.836(7)</td>
<td>.818(8)</td>
<td>.960(3)</td>
<td>.947(5)</td>
</tr>
<tr>
<td>Vehicle</td>
<td>.812(7)</td>
<td>.863(2)</td>
<td>.870(1)</td>
<td>.750(8)</td>
<td>.854(5)</td>
<td>.855(4)</td>
<td>.601(9)</td>
<td>.859(3)</td>
<td>.827(6)</td>
</tr>
<tr>
<td>Vowel</td>
<td>.776(5)</td>
<td>.897(1)</td>
<td>.858(2)</td>
<td>.720(7)</td>
<td>.647(8)</td>
<td>.623(9)</td>
<td>.770(6)</td>
<td>.851(3)</td>
<td>.794(4)</td>
</tr>
<tr>
<td>Wine</td>
<td>.883(8)</td>
<td>.901(7)</td>
<td>.931(4)</td>
<td>.910(6)</td>
<td>.921(5)</td>
<td>.933(3)</td>
<td>.942(2)</td>
<td>.947(1)</td>
<td>.882(9)</td>
</tr>
<tr>
<td>Average Rank</td>
<td>.861(5)</td>
<td>.905(1)</td>
<td>.898(2)</td>
<td>.838(6)</td>
<td>.837(7)</td>
<td>.818(8)</td>
<td>.789(9)</td>
<td>.891(3)</td>
<td>.866(4)</td>
</tr>
</tbody>
</table>

| Table 1. Comparison of RBLR, RBLR+, BCLR with state-of-the-art methods (Kendall’s Tau) |
Table 2. Performance of RBLR+, BCLR in terms of Kendall’s tau (mean and standard deviation)

<table>
<thead>
<tr>
<th></th>
<th>RBLR+</th>
<th>BCLR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Glass</td>
<td>.906 ±.006</td>
<td>.863 ±.044</td>
</tr>
<tr>
<td>Iris</td>
<td>.961 ±.002</td>
<td>.971 ±.003</td>
</tr>
<tr>
<td>Vehicle</td>
<td>.863 ±.003</td>
<td>.870 ±.031</td>
</tr>
<tr>
<td>Vowel</td>
<td>.897 ±.017</td>
<td>.858 ±.018</td>
</tr>
<tr>
<td>Wine</td>
<td>.901 ±.001</td>
<td>.931 ±.043</td>
</tr>
</tbody>
</table>

Table 3. Performance of RBLR+, BCLR in terms of Spearman’s rank (mean and standard deviation)

<table>
<thead>
<tr>
<th></th>
<th>RBLR+</th>
<th>BCLR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Glass</td>
<td>.928 ±.005</td>
<td>.890 ±.046</td>
</tr>
<tr>
<td>Iris</td>
<td>.971 ±.001</td>
<td>.977 ±.025</td>
</tr>
<tr>
<td>Vehicle</td>
<td>.891 ±.003</td>
<td>.896 ±.025</td>
</tr>
<tr>
<td>Vowel</td>
<td>.945 ±.012</td>
<td>.924 ±.013</td>
</tr>
<tr>
<td>Wine</td>
<td>.919 ±.011</td>
<td>.948 ±.032</td>
</tr>
</tbody>
</table>
Considerations

- Very **interpretable** and clear model
- **Global** Model
- **Modular** Conception (meta-learner)
- Treat preference information **at once**
- Can also deal with **criteria** instead of simple attributes (MCDA)
- Good performance with respect to existing methods (Kendall’s Tau and Spearman’s rank correlation)
WHAT ABOUT OUR TOY EXAMPLE?
Toy Example: How to solve it?

<table>
<thead>
<tr>
<th>Customers’ Profile</th>
<th>Fiction</th>
<th>History</th>
<th>Business</th>
<th>Hobbies</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5 0.3 -2.1 -0.5</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>-0.2 0.1 2.7 1.5</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>1.5 0.4 0.1 0.7</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>-1.5 1.3 2.4 -1.6</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>2.3 -0.3 0.1 0.1</td>
<td>?</td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
</tbody>
</table>
### Toy Example: Learning Reduction

<table>
<thead>
<tr>
<th>$\Delta G_{1}G_{1}$</th>
<th>$A_{1}C_{1}$</th>
<th>$A_{2}G_{2}$</th>
<th>$A_{2}C_{2}$</th>
<th>$A_{3}G_{3}$</th>
<th>$A_{3}C_{3}$</th>
<th>$A_{4}G_{4}$</th>
<th>$A_{4}C_{4}$</th>
<th>$D_{1}$</th>
<th>$D_{2}$</th>
<th>$D_{1}$</th>
<th>$D_{2}$</th>
<th>$D_{1}$</th>
<th>$D_{2}$</th>
<th>$D_{1}$</th>
<th>$D_{2}$</th>
<th>$D_{1}$</th>
<th>$D_{2}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.500</td>
<td>0.500</td>
<td>0.300</td>
<td>0.300</td>
<td>-2.100</td>
<td>-2.100</td>
<td>-0.500</td>
<td>-0.500</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500</td>
<td>0.500</td>
<td>0.300</td>
<td>0.300</td>
<td>-2.100</td>
<td>-2.100</td>
<td>-0.500</td>
<td>-0.500</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500</td>
<td>0.500</td>
<td>0.300</td>
<td>0.300</td>
<td>-2.100</td>
<td>-2.100</td>
<td>-0.500</td>
<td>-0.500</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500</td>
<td>0.500</td>
<td>0.300</td>
<td>0.300</td>
<td>-2.100</td>
<td>-2.100</td>
<td>-0.500</td>
<td>-0.500</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-0.200</td>
<td>-0.200</td>
<td>0.100</td>
<td>0.100</td>
<td>2.700</td>
<td>2.700</td>
<td>1.500</td>
<td>1.500</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-0.200</td>
<td>-0.200</td>
<td>0.100</td>
<td>0.100</td>
<td>2.700</td>
<td>2.700</td>
<td>1.500</td>
<td>1.500</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-0.200</td>
<td>-0.200</td>
<td>0.100</td>
<td>0.100</td>
<td>2.700</td>
<td>2.700</td>
<td>1.500</td>
<td>1.500</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-0.200</td>
<td>-0.200</td>
<td>0.100</td>
<td>0.100</td>
<td>2.700</td>
<td>2.700</td>
<td>1.500</td>
<td>1.500</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.500</td>
<td>1.500</td>
<td>0.400</td>
<td>0.400</td>
<td>0.100</td>
<td>0.100</td>
<td>0.700</td>
<td>0.700</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.500</td>
<td>1.500</td>
<td>0.400</td>
<td>0.400</td>
<td>0.100</td>
<td>0.100</td>
<td>0.700</td>
<td>0.700</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.500</td>
<td>1.500</td>
<td>0.400</td>
<td>0.400</td>
<td>0.100</td>
<td>0.100</td>
<td>0.700</td>
<td>0.700</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.500</td>
<td>1.500</td>
<td>0.400</td>
<td>0.400</td>
<td>0.100</td>
<td>0.100</td>
<td>0.700</td>
<td>0.700</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-1.500</td>
<td>-1.500</td>
<td>1.300</td>
<td>1.300</td>
<td>2.400</td>
<td>2.400</td>
<td>-1.600</td>
<td>-1.600</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-1.500</td>
<td>-1.500</td>
<td>1.300</td>
<td>1.300</td>
<td>2.400</td>
<td>2.400</td>
<td>-1.600</td>
<td>-1.600</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-1.500</td>
<td>-1.500</td>
<td>1.300</td>
<td>1.300</td>
<td>2.400</td>
<td>2.400</td>
<td>-1.600</td>
<td>-1.600</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-1.500</td>
<td>-1.500</td>
<td>1.300</td>
<td>1.300</td>
<td>2.400</td>
<td>2.400</td>
<td>-1.600</td>
<td>-1.600</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-1.500</td>
<td>-1.500</td>
<td>1.300</td>
<td>1.300</td>
<td>2.400</td>
<td>2.400</td>
<td>-1.600</td>
<td>-1.600</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-1.500</td>
<td>-1.500</td>
<td>1.300</td>
<td>1.300</td>
<td>2.400</td>
<td>2.400</td>
<td>-1.600</td>
<td>-1.600</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-1.500</td>
<td>-1.500</td>
<td>1.300</td>
<td>1.300</td>
<td>2.400</td>
<td>2.400</td>
<td>-1.600</td>
<td>-1.600</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-1.500</td>
<td>-1.500</td>
<td>1.300</td>
<td>1.300</td>
<td>2.400</td>
<td>2.400</td>
<td>-1.600</td>
<td>-1.600</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-1.500</td>
<td>-1.500</td>
<td>1.300</td>
<td>1.300</td>
<td>2.400</td>
<td>2.400</td>
<td>-1.600</td>
<td>-1.600</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **$\Delta G_{1}$**: Logarithmic change in group 1
- **$A_{1}$**: Cost for attribute 1
- **$A_{2}$**: Gain for attribute 2
- **$A_{3}$**: Cost for attribute 3
- **$A_{4}$**: Gain for attribute 4
- **$D_{1}$**: Decision for group 1
- **$D_{2}$**: Decision for group 2
- **GT**: Ground Truth
### Toy Example: Rules Generation

<table>
<thead>
<tr>
<th>ID</th>
<th>DECISION PART 1</th>
<th>CONDITION 1</th>
<th>CONDITION 2</th>
<th>CONDITION 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(D &gt;= GT)</td>
<td>(A3COST &lt;= -2.1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>(D &gt;= GT)</td>
<td>(A1COST &lt;= -0.2)  &amp;</td>
<td>(Labels_Relation = L2VSL4)</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>(D &gt;= GT)</td>
<td>(A1COST &lt;= -0.2)  &amp;</td>
<td>(Labels_Relation = L3VSL4)</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>(D &gt;= GT)</td>
<td>(A3COST &lt;= 0.1)   &amp;</td>
<td>(Labels_Relation = L1VSL2)</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>(D &gt;= GT)</td>
<td>(A3COST &lt;= 0.1)   &amp;</td>
<td>(Labels_Relation = L1VSL3)</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>(D &gt;= GT)</td>
<td>(A1GAIN &gt;= -0.2)  &amp;</td>
<td>(A1COST &lt;= -0.2)  &amp;</td>
<td>(Labels_Relation = L1VSL4)</td>
</tr>
<tr>
<td>7</td>
<td>(D &gt;= GT)</td>
<td>(A1COST &lt;= -1.5)  &amp;</td>
<td>(Labels_Relation = L2VSL3)</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>(D &lt;= LT)</td>
<td>(A1COST &gt; 1.5)    &amp;</td>
<td>(Labels_Relation = L1VSL4)</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>(D &lt;= LT)</td>
<td>(A1COST &gt; 1.5)    &amp;</td>
<td>(Labels_Relation = L2VSL3)</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>(D &lt;= LT)</td>
<td>(A1COST &gt; 1.5)    &amp;</td>
<td>(Labels_Relation = L2VSL4)</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>(D &lt;= LT)</td>
<td>(A1COST &gt; 1.5)    &amp;</td>
<td>(Labels_Relation = L3VSL4)</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>(D &lt;= LT)</td>
<td>(A1GAIN &lt;= -0.2)  &amp;</td>
<td>(Labels_Relation = L1VSL2)</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>(D &lt;= LT)</td>
<td>(A1GAIN &lt;= -0.2)  &amp;</td>
<td>(Labels_Relation = L1VSL3)</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>(D &lt;= LT)</td>
<td>(A1GAIN &lt;= -1.5)  &amp;</td>
<td>(Labels_Relation = L1VSL4)</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>(D &lt;= LT)</td>
<td>(A1GAIN &lt;= -0.2)  &amp;</td>
<td>(A1COST &gt;= -0.2)  &amp;</td>
<td>(Labels_Relation = L2VSL3)</td>
</tr>
</tbody>
</table>

**Rule type:** CERTAIN  **Usage type:** AT LEAST  **Characteristic class:** GT

- **Support:** 1
- **SupportingExamples:** 22
- **Strength:** 0.042
- **Confidence:** 1
NEW Customer supports rules # 4, 5, 8, 9, 10, 11

Pair-wise preferences provided by rules are:
- \((L_1 \succ L_2) \Leftrightarrow \text{Fiction preferred to History} - [0, 1]\)
- \((L_1 \succ L_3) \Leftrightarrow \text{Fiction preferred to Business} - [0, 1]\)
- \((L_4 \succ L_1) \Leftrightarrow \text{Hobbies preferred to Fiction} - [0, 1]\)
- \((L_3 \succ L_2) \Leftrightarrow \text{Business preferred to History} - [0, 1]\)
- \((L_4 \succ L_2) \Leftrightarrow \text{Hobbies preferred to History} - [0, 1]\)
- \((L_4 \succ L_3) \Leftrightarrow \text{Hobbies preferred to Business} - [0, 1]\)

Labels Scores:
- \(S(L_1) = 1\)
- \(S(L_2) = -3\)
- \(S(L_3) = -1\)
- \(S(L_4) = 3\)

The final ranking is:

\(\text{Hobbies} \succ \text{Fiction} \succ \text{Business} \succ \text{History}\)
### Toy Example: Ranking Generation

<table>
<thead>
<tr>
<th>Customers</th>
<th>Label Ranking</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer 1</td>
<td>Fiction $\succ$ History $\succ$ Business $\succ$ Hobbies</td>
</tr>
<tr>
<td>Customer 2</td>
<td>Business $\succ$ History $\succ$ Fiction $\succ$ Hobbies</td>
</tr>
<tr>
<td>Customer 3</td>
<td>Hobbies $\succ$ Fiction $\succ$ Business $\succ$ History</td>
</tr>
<tr>
<td>Customer 4</td>
<td>History $\succ$ Business $\succ$ Hobbies $\succ$ Fiction</td>
</tr>
<tr>
<td>NEW Customer</td>
<td>Hobbies $\succ$ Fiction $\succ$ Business $\succ$ History</td>
</tr>
</tbody>
</table>
For which kind of profile (for example): \(Hobbies \succ Fiction\)?

- This means rules with \((L_1, L_4)\) as relation attribute and \(d = LT\) as decision (if necessary fixing a support threshold):

\[
\begin{align*}
\text{8 (D <= LT)} & \leq (A1COST \geq 1.5) & \& (Labels_{Relation} = L1VSL4) \\
\text{14 (D <= LT)} & \leq (A1GAIN \leq -1.5) & \& (Labels_{Relation} = L1VSL4)
\end{align*}
\]

- This could be helpful when searching for a specific customer (marketing strategies, advertising, ...).
Thank you!
SCORES
Let be $R_{i,j}$ the set of rules concerning the pair $(\lambda_i, \lambda_j)$ and $R_{(i,j,GT)}, R_{(i,j,LT)} \subseteq R_{i,j}$ subsets having $d = GT$ and $d = LT$ as decisions.
For any rule $r \in R_{i,j}$, for a given unknown $x$, let define:

- $\omega_r = \frac{S_r}{S_{i,j}}$ with: $S_{i,j} = \sum_{r \in R_{i,j}} S_r$

- $T_{i,j} = \sum_{r \in R_{i,j}} \omega_r \cdot \delta^x_r$ with: $\delta^x_r = \begin{cases} 
1 & \text{if } x \text{ supports } r, \\
0 & \text{otherwise}.
\end{cases}$

- $W_{(i,j)}^+ = \sum_{r \in R_{(i,j),GT}} \omega_r$, $W_{(i,j)}^- = \sum_{r \in R_{(i,j),LT}} \omega_r$

- $\alpha_{(i,j)}^+ = \sum_{r \in R_{(i,j),GT}} \omega_r \cdot \delta^x_r$, $\alpha_{(i,j)}^- = \sum_{r \in R_{(i,j),LT}} \omega_r \cdot \delta^x_r$

- $\Gamma_{(i,j)}^+ = \frac{\alpha_{(i,j)}^+}{T_{i,j}}$, $\Gamma_{(i,j)}^- = \frac{\alpha_{(i,j)}^-}{T_{i,j}}$
COMPLEXITY
Theorem

The total number of training instances (after the learning reduction) is \( n \frac{k(k-1)}{2} \) where \( n \) is the number of original training instances and \( k \) is the number of labels.

Proof.

Each original training instance is split into the set \( \{X_{1,2}, X_{1,3}, \ldots, X_{i,j}, \ldots\} \) which contains exactly \( \frac{k(k-1)}{2} \) new instances, one for each possible pair.
Corollary

The time complexity of the inferring rule algorithm (after the learning reduction) is \( O\left(\frac{n^{k(k-1)}}{2}(2s+1)(\frac{n^{k(k-1)}}{2}+1)(2s+2)\right) \) where \( n \) is the number of original training instances, \( k \) is the number of labels and \( s \) is the original number of attributes.

Proof.

Since the time complexity of VC-DomLEM is given by \( O\left(\frac{ns(n+1)(s+1)}{4}\right) \), after the learning reduction the total number of training instance is \( n\frac{k(k-1)}{2} \) and the total number of attributes is \( 2s+1 \). \( \square \)
The Net Flow Score Procedure as a minimizer for the expected loss
By considering $\Gamma^+_{(i,j)} = P(\lambda_i \succ_x \lambda_j)$ as the probability to observe $\lambda_i \succ_x \lambda_j$ and $\Gamma^-_{(i,j)} = P(\lambda_j \succ_x \lambda_i)$ as the probability to observe $\lambda_j \succ_x \lambda_i$, for a given $x$, the net flow score is given by:

$$S(i) = \sum_{j \neq i} (P(\lambda_i \succ_x \lambda_j) - P(\lambda_j \succ_x \lambda_i)),$$

The final ranking $\pi$ is obtained by ordering labels according to decreasing values of these scores (the higher the score, the higher the preference in the ranking): $S(i) > S(j) \iff \pi_i < \pi_j$. 

---

Net Flow Score
Distance Metrics

Given two permutations \( \pi, \pi' \in \Omega \):

- The *Spearman’s footrule* distance:
  \[
  F(\pi, \pi') = \sum_{i=1}^{k} |\pi_i - \pi'_i|
  \]

- The *Kendall’s tau* distance:
  \[
  K(\pi, \pi') = \# \{(i, j) \in [k] \times [k] : i < j | \pi_i > \pi_j \land \pi'_i < \pi'_j \}
  \]

- The sum of squared rank distances:
  \[
  D(\pi, \pi') = \sum_{i=1}^{k} (\pi_i - \pi'_i)^2
  \]
Distance Metrics

**Lemma**

For all \( \pi, \pi' \in \Omega \), \( F(\pi, \pi') \leq D(\pi, \pi') \).

**Proof.**

Since \( |\pi_i - \pi'_i| \in \{0, 1, 2, ..., k - 1\} \), \( \forall i \), it follows that
\[
|\pi_i - \pi'_i| \leq (\pi_i - \pi'_i)^2, \forall i.
\]

By considering the sum over \( i \), it is easy to prove the lemma.

**Theorem**

*(DIACONIS-GRAHAM)*

For all \( \pi, \pi' \in \Omega \), \( K(\pi, \pi') \leq F(\pi, \pi') \leq 2K(\pi, \pi') \).
Corollary

For all \( \pi, \pi' \in \Omega \), \( K(\pi, \pi') \leq D(\pi, \pi') \).

Proof.

This directly follows from lemma 1 and theorem 1.
Lemma

(Hüllemeier et al.) Let \( s_i, i = 1, 2, \ldots k \) be real numbers such that \( 0 \leq s_1 \leq s_2 \leq \ldots \leq s_k \). Then, for all permutations \( \pi \in \Omega \),
\[
\sum_{i=1}^{k} (i - s_i)^2 \leq \sum_{i=1}^{k} (i - s_{\pi_i})^2
\]

Lemma

Let be \( s_i = \frac{1}{2}(k + 1 - S(i)) \quad (1) \)

Then \( s_i > 0, \forall i \) and \( s_i \leq s_j \) whenever \( S(i) \geq S(j) \).

Proof.

Since \( -k \leq S(i) \leq k < k + 1 \), then \( \forall i, s_i > 0 \). It is obvious that \( s_i \leq s_j \) is equivalent to \( S(i) \geq S(j) \).
Lemma

Let $\mathbb{P}(\cdot|x)$ a probability distribution over $\Omega$ and let be

$$s_i = \frac{1}{2}(k + 1 - S(i)) \quad \text{and} \quad \mathbb{P}(\lambda_i \succsim_x \lambda_j) = \sum_{\pi: \pi_i < \pi_j} \mathbb{P}(\pi|x).$$

Then $s_i = k + 1 - \sum_{\pi} \mathbb{P}(\pi|x) \pi_i$. 

Net Flow Score as a minimizer for the expected loss
Proof.

\[ s_i = \frac{1}{2} (k + 1 - S(i)) = \frac{1}{2} (k + 1 - \sum_{j \neq i} (P(\lambda_i \succ x \lambda_j) - P(\lambda_j \succ x \lambda_i))) = \]
\[ \frac{1}{2} (k + 1 - \sum_{j \neq i} (2P(\lambda_i \succ x \lambda_j) - 1)) = \frac{1}{2} (k + 1 - \sum_{j \neq i} 2P(\lambda_i \succ x \lambda_j) + \sum_{j \neq i} 1) = \]
\[ \frac{1}{2} (k + 1 - \sum_{j \neq i} 2P(\lambda_i \succ x \lambda_j) + k - 1) = \frac{1}{2} (2k - 2 \sum_{j \neq i} P(\lambda_i \succ x \lambda_j)) = \]
\[ k - \sum_{j \neq i} \sum_{\pi : \pi_i < \pi_j} P(\pi | x) = k - \sum_{\pi} P(\pi | x) \sum_{j \neq i} \delta(i,j) = k - \sum_{\pi} P(\pi | x)(\pi_i - 1) = \]
\[ k - \sum_{\pi} P(\pi | x)\pi_i + \sum_{\pi} P(\pi | x)) = k - \sum_{\pi} P(\pi | x)\pi_i + 1. \]

Where \( \delta_{i,j}^{(\pi)} = 1 \) if \( \pi_i < \pi_j \) and \( \delta_{i,j}^{(\pi)} = 0 \) otherwise.
The expected distance

\[ \mathbb{E}(D(\pi_x, \pi'_x)) = \sum_{\pi} \mathbb{P}(\pi|x) D(\pi_x, \pi'_x) = \sum_{\pi} \mathbb{P}(\pi|x) \sum_{i=1}^{k} (\pi_i - \pi'_i)^2, \]

is minimal by choosing \( \pi' \) so that \( \pi'_i \leq \pi'_j \) whenever \( s_i \leq s_j \) with \( s_i \) given by (1).
Proof.

\[ E(D(\pi_x, \pi'_x)) = \sum_{\pi} P(\pi|x) \sum_{i=1}^{k} (\pi_i - \pi'_i)^2 = \]

\[ = \sum_{i=1}^{k} \sum_{\pi} P(\pi|x)(\pi_i - \pi'_i)^2 = \]

\[ = \sum_{i=1}^{k} \sum_{\pi} P(\pi|x)(\pi_i - s_i + s_i - \pi'_i)^2 = \]

\[ = \sum_{i=1}^{k} \sum_{\pi} P(\pi|x)[(\pi_i - s_i)^2 - 2(\pi_i - s_i)(s_i - \pi'_i) + (s_i - \pi'_i)^2] = \]

\[ = \sum_{i=1}^{k} \sum_{\pi} P(\pi|x)(\pi_i - s_i)^2 - 2(s_i - \pi'_i) \sum_{\pi} P(\pi|x)(\pi_i - s_i) + \sum_{\pi} P(\pi|x)(s_i - \pi'_i)^2. \]

In the last equation, the terms \( \sum_{\pi} P(\pi|x)(\pi_i - s_i)^2 \) and \( \sum_{\pi} P(\pi|x)(s_i - \pi'_i)^2 \) are constants \( c \) and \( h \) since they do not depend on \( \pi' \).

So \[ E(D(\pi_x, \pi'_x)) = \sum_{i=1}^{k} c - 2h \sum_{i=1}^{k} (s_i - \pi'_i) + \sum_{i=1}^{k} (s_i - \pi'_i)^2 \sum_{\pi} P(\pi|x). \]
\[ E(D(\pi_x, \pi'_x)) = \sum_{i=1}^{k} c - 2h \sum_{i=1}^{k} (s_i - \pi'_i) + \sum_{i=1}^{k} (s_i - \pi'_i)^2 \sum_{\pi} \mathbb{P}(\pi|x) \]

The term \( \sum_{i=1}^{k} c \) is a constant \( C \), the term \(-2h \sum_{i=1}^{k} (s_i - \pi'_i) \) is also a constant \( H \) since \( \forall \pi' \in \Omega : \sum_{i=1}^{k} (s_i - \pi'_i) = \sum_{i=1}^{k} (s_i - i) \) and the term \( \sum_{\pi} \mathbb{P}(\pi|x) \) is obviously 1.

So the expected loss can be written as:
\[ E(D(\pi_x, \pi'_x)) = \sum_{i=1}^{k} (s_i - \pi'_i)^2 + C + H \]

and it is straightforward to see that the first term of this equation is minimal when choosing \( \pi' \) so that \( \pi'_i \leq \pi'_j \) whenever \( s_i \leq s_j \) (as stated in lemma 2). This proves the theorem.