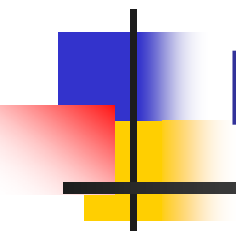


Decision Tree and Instance-Based Learning for Label Ranking

Weiwei Cheng, Jens Huhn, Eyke Hullermeier, ICML,2009

Presented by WANG Wei

Label Ranking (An example)

- Learning the customers' preference on cars

	label ranking
customer 1	MINI _ Toyota _ BMW
customer 2	BMW _ MINI _ Toyota
customer 3	BMW _ Toyota _ MINI
customer 4	Toyota _ MINI _ BMW
new customer	???

Where the customers could be represented by features vectors. Eg(gender, age, place of birth,...)

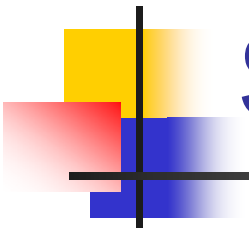
Label ranking (An example)

- Learning the customers' preference on cars

	MINI	Toyota	BMW
customer 1	1	2	3
customer 2	2	3	1
customer 3	3	2	1
customer 4	2	1	3
new customer	?	?	?

$\pi(i)$ = position of the i-th label in the ranking

1: MINI 2: Toyota 3: BMW



Some Challenges

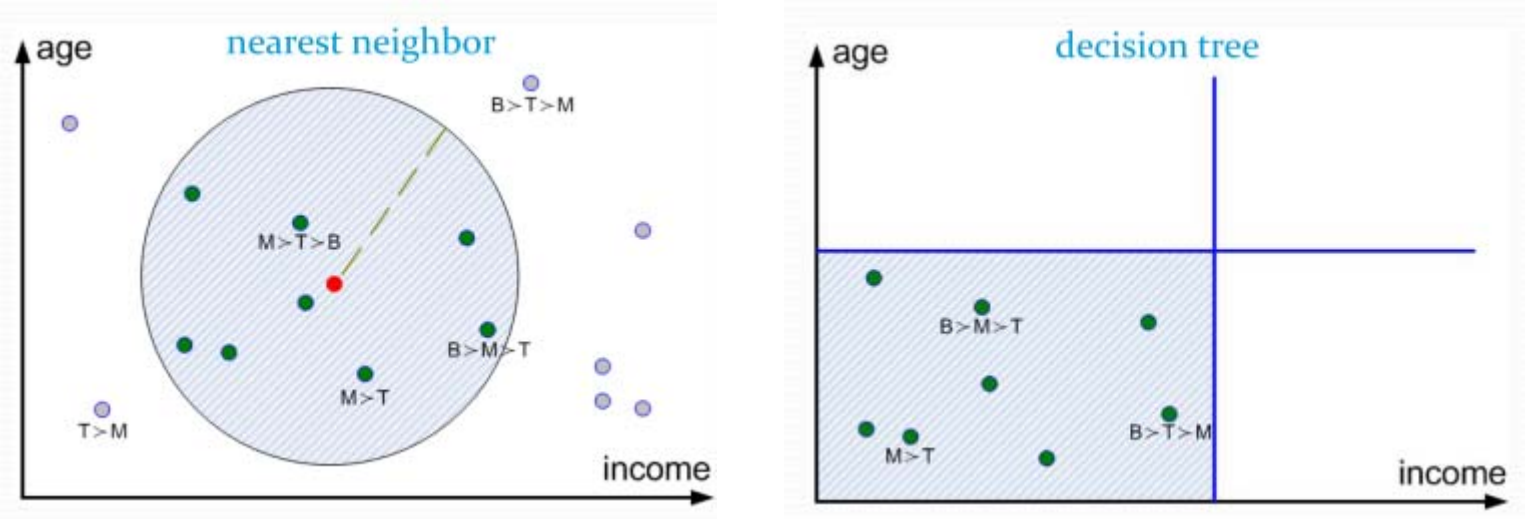
- Training data: Naive Bayes?
- Distance measures for ranking.
- Incomplete ranking.

Label Ranking (formally)

- Given:
  - a set of training instances: $\{\mathbf{x}_k \mid k = 1 \ldots m\} \subseteq \mathbf{X}$
  - a set of labels $L = \{l_i \mid i = 1 \ldots n\}$
  - for each training instances $\mathbf{x}_k$: a set of pairwise preferences of the form $l_i \succ_{\mathbf{x}_k} l_j$ (for some of the labels)
- Find:

a ranking function ($\mathcal{X} \to \Omega$ mapping) that maps each $\mathbf{x} \in \mathbf{X}$ to a ranking $\succ_{\mathbf{x}}$ of L (Permutation $\pi_{\mathbf{x}}$) and generalizes well in terms of a loss function on rankings (e.g. , kendall's tau coefficient)

Local approach (this work)



- Target function $\mathcal{X} \to \Omega$ is estimated (on demand) in a local way.
- Distribution of ranking is (approx.) constant in a local region.
- Core part is to estimate the locally constant model.

Local approach

- Output (ranking) of an instance x is generated according to a distribution $\mathcal{P}(\cdot \mid \mathbf{x})$ on Ω.
- This distribution is (approximately) constant within the local region under consideration.
- Nearby preferences are considered as a sample generated by $\mathcal{P}$, which is estimated on the basis of this sample via ML.

Probabilistic Model for Ranking

Mallows model (Mallows, Biometrika, 1957)

$$\mathcal{P}(\sigma|\theta,\pi) = \frac{\exp(-\theta d(\pi,\sigma))}{\phi(\theta,\pi)}$$

with

center ranking $\pi \in \Omega$

spread parameter $\theta > 0$

and d(.) is a right invariant metric on permutations

$$\forall \pi,\sigma,\nu \in \Omega,\ d(\pi,\sigma) = d(\pi\nu,\sigma\nu).$$

Inference (complete rankings)

Rankings $\boldsymbol{\sigma} = \{\sigma_1, \ldots, \sigma_k\}$ observed locally.

$$\mathcal{P}(\boldsymbol{\sigma}|\theta,\pi) = \prod_{i=1}^{k} \mathcal{P}(\sigma_i|\theta,\pi)$$
$$= \prod_{i=1}^{k} \frac{\exp\left(-\theta d(\sigma_i,\pi)\right)}{\phi(\theta)}$$
$$= \frac{\exp\left(-\theta(d(\sigma_1,\pi)+\ldots+d(\sigma_k,\pi))\right)}{\phi^k(\theta)}$$
$$= \frac{\exp\left(-\theta\sum_{i=1}^{k} d(\sigma_i,\pi)\right)}{\left(\prod_{j=1}^{n}\frac{1-\exp(-j\theta)}{1-\exp(-\theta)}\right)^k}.$$

ML

$$\hat{\pi} = \arg\min_{\pi} \sum_{i=1}^{k} d(\sigma_i,\pi)$$

monotone in θ

$$\frac{1}{k}\sum_{i=1}^{k} d(\sigma_i,\hat{\pi}) = \frac{n\exp(-\theta)}{1-\exp(-\theta)} - \sum_{j=1}^{n}\frac{j\exp(-j\theta)}{1-\exp(-j\theta)}$$

Inference (incomplete rankings)

Probability of an incomplete ranking:

$$\mathcal{P}(E(\sigma_i) \,|\, \theta, \pi) = \sum_{\sigma \in E(\sigma_i)} \mathcal{P}(\sigma \,|\, \theta, \pi)$$

where $E(\sigma_i)$ denotes the set of consistent extensions of σ_i.

Example for label set $\{a,b,c\}$:

Observation σ	Extensions $E(\sigma)$
$a \_ b$	$a \_ b \_ c$ $a \_ c \_ b$ $c \_ a \_ b$

Inference (incomplete rankings) cont.

The corresponding likelihood:

$$\begin{aligned}\mathcal{P}(\boldsymbol{\sigma}|\theta,\pi) &= \prod_{i=1}^{k}\mathcal{P}(E(\sigma_i)|\theta,\pi)\\ &= \prod_{i=1}^{k}\sum_{\sigma\in E(\sigma_i)}\mathcal{P}(\sigma|\theta,\pi)\\ &= \frac{\prod_{i=1}^{k}\sum_{\sigma\in E(\sigma_i)}\exp\left(-\theta d(\sigma,\pi)\right)}{\left(\prod_{j=1}^{n}\frac{1-\exp(-j\theta)}{1-\exp(-\theta)}\right)^{k}}.\end{aligned}$$

Exact MLE $(\hat{\pi},\hat{\theta}) = \arg\max_{\pi,\theta}\mathcal{P}(\boldsymbol{\sigma}|\theta,\pi)$ becomes infeasible when n is large. Approximation is needed.

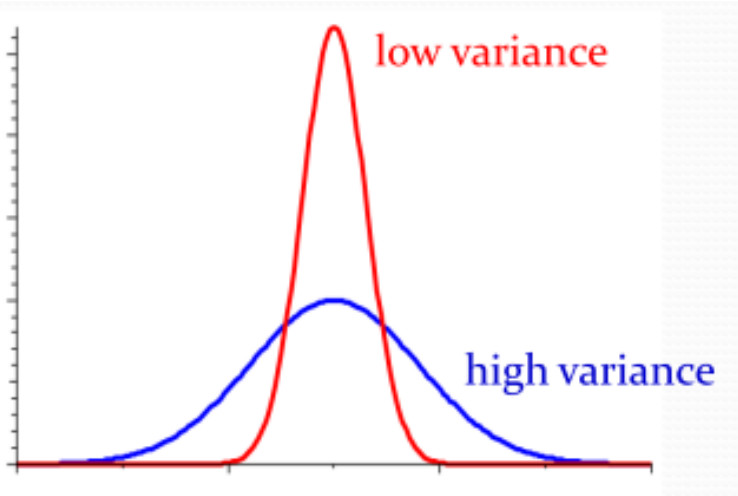
Inference (incomplete rankings) cont.

Approximation via a variant of EM, viewing the non-observed labels as hidden variables.

- replace the E-step of EM algorithm with a maximization step

1. Start with an initial center ranking (via generalized Borda count)
2. Replace an incomplete observation with its most probable extension (first M-step, can be done efficiently)
3. Obtain MLE as in the complete ranking case (second M-step)
4. Replace the initial center ranking with current estimation
5. Repeat until convergence

Inference

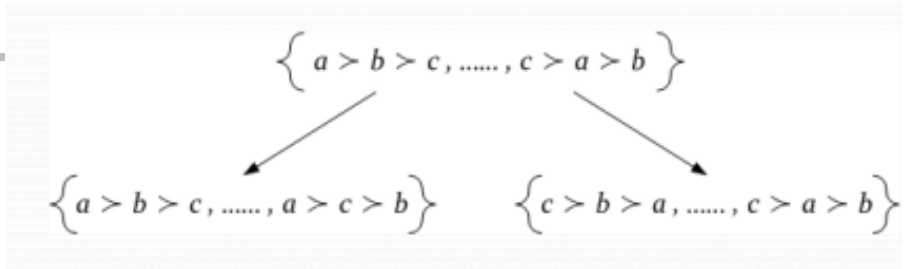


Not only the estimated ranking $\hat{\pi}$ is of interest ...

... but also the spread parameter $\hat{\theta}$, which is a measure of precision and, therefore, reflects the confidence/reliability of the prediction (just like the variance of an estimated mean).

The bigger $\hat{\theta}$, the more peaked the distribution around the center ranking.

Label Ranking Trees



Major modifications:

- split criterion

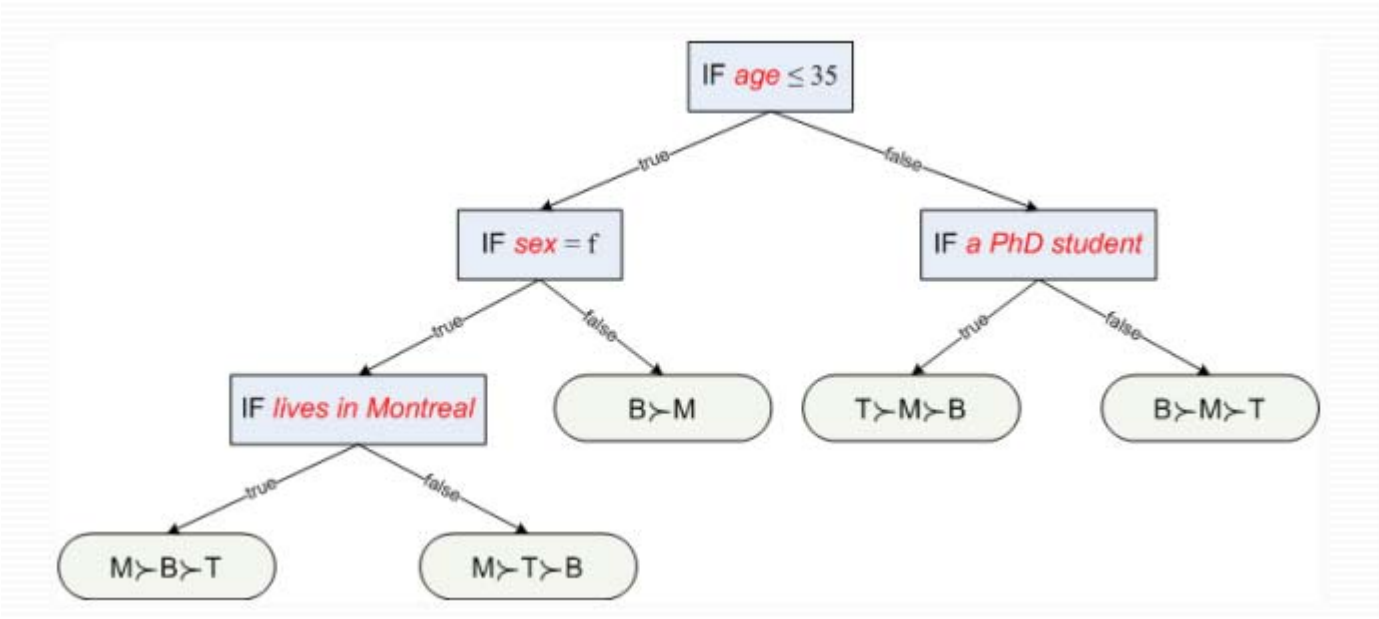
  split ranking set T into T^+ and T^-, maximizing goodness-of-fit

$$\frac{|T^+| \cdot \theta^+ + |T^-| \cdot \theta^-}{|T|}$$

- stopping criterion for partition

1. tree is pure
   any two labels in two different rankings have the same order
2. number of labels in a node is too small
   prevent an excessive fragmentation

Labels: BMW, Mini, Toyota



Experimental Results

	complete rankings				30% missing labels				60% missing labels			
	CC	IBLR	LRT		CC	IBLR	LRT		CC	IBLR	LRT	
authorship	.920(2)	.936(1)	.882(3)	1.1	.891(2)	.932(1)	.871(3)	0.9	.835(2)	.920(1)	.828(3)	0.7
bodyfat	.281(1)	.248(2)	.117(3)	1.6	.260(1)	.223(2)	.097(3)	1.7	.224(1)	.180(2)	.070(3)	1.0
calhousing	.250(3)	.351(1)	.324(2)	0.7	.249(3)	.327(1)	.307(2)	0.5	.247(3)	.289(1)	.273(2)	0.3
cpu-small	.475(2)	.506(1)	.447(3)	2.3	.474(2)	.498(1)	.405(3)	2.3	.470(2)	.480(1)	.367(3)	1.5
elevators	.768(1)	.733(3)	.760(2)	0.2	.767(1)	.719(3)	.756(2)	0.2	.765(1)	.690(3)	.742(2)	0.3
fried	.999(1)	.935(2)	.890(3)	5.5	.998(1)	.928(2)	.863(3)	5.3	.997(1)	.895(2)	.809(3)	3.0
glass	.846(3)	.865(2)	.883(1)	2.5	.835(2)	.824(3)	.850(1)	2.0	.789(2)	.771(3)	.799(1)	2.0
housing	.660(3)	.745(2)	.797(1)	2.3	.655(3)	.697(2)	.734(1)	2.4	.638(1)	.630(3)	.634(2)	1.5
iris	.836(3)	.966(1)	.947(2)	1.5	.807(3)	.945(1)	.909(2)	1.2	.743(3)	.882(1)	.794(2)	1.5
pendigits	.903(3)	.944(1)	.935(2)	6.2	.902(3)	.924(1)	.914(2)	3.2	.900(1)	.899(2)	.871(3)	2.2
segment	.914(3)	.959(1)	.949(2)	3.8	.911(3)	.934(1)	.933(2)	3.8	.902(2)	.902(3)	.903(1)	2.3
stock	.737(3)	.927(1)	.895(2)	1.5	.735(3)	.904(1)	.877(2)	1.6	.724(3)	.858(1)	.827(2)	1.1
vehicle	.855(2)	.862(1)	.827(3)	0.8	.839(2)	.842(1)	.819(3)	0.9	.810(1)	.791(2)	.764(3)	0.5
vowel	.623(3)	.900(1)	.794(2)	4.6	.615(3)	.824(1)	.718(2)	3.6	.598(3)	.722(1)	.615(2)	3.2
wine	.933(2)	.949(1)	.882(3)	0.8	.911(2)	.941(1)	.862(3)	1.1	.853(1)	.789(2)	.752(3)	0.8
wisconsin	.629(1)	.506(2)	.343(3)	1.6	.617(1)	.484(2)	.284(3)	1.5	.566(1)	.438(2)	.251(3)	1.6
average rank	2.25	1.44	2.31		2.19	1.50	2.31		1.75	1.88	2.38	

Accuracy (Kendall's tau)

- Typical "learning curves":