Estimating Attributed Central Orders — An Empirical Comparison —

T.Kamishima[1], H.Kazawa[2], S.Akaho[1]
1) National Institute of AIST, Japan
2) NTT Communication Science Labs, Japan

15th ECML, Pisa, Italy, 20-24/9/2004

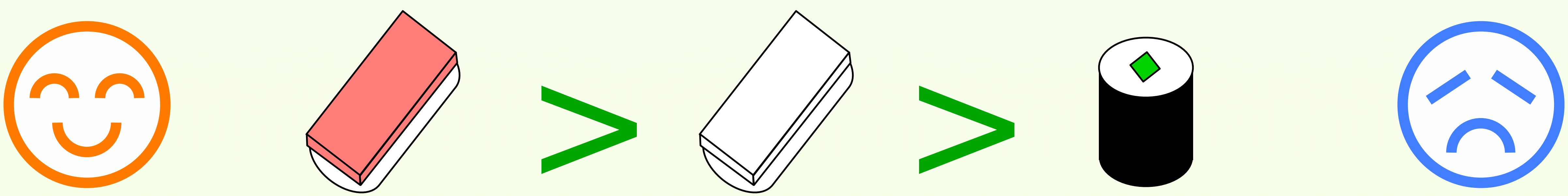
Overview

Attributed Central Order (ACO): an order as concordant with given samples as possible

Supervised Ordering: a learning function to sort objects so as to be concordant with ACO

Order: object sequence sorted according to a particular property

ex. a sequence of sushi sorted according to my preference



prefer Fatty Tuna > Squid > Cucumber Roll not prefer

“I prefer *Fatty Tuna* to *Squid*”

but “How much prefer is unknown”

comparing 5 supervised ordering methods

Why Orders?

- **Widely Used**
  - Common representation form
    ex. search result list, top-seller list
  - Fit for measuring subjective quantities
    subjective quantities, s.g., preference or sensation, can more easily measured by ordinal relation than by numerical scale
- **Important Level of Measurement**
  - An order is intrinsic regarding decision/selection
    no matter how small the errors of scores, it can not be guaranteed that candidates are correctly ordered

Attributed Central Orders & Supervised Ordering

given sample orders

A > B > C
D > E > B > C
A > D > C

concordant with both orders on average

unseen sample orders

A > B > C
E > D > A > B
D > C > B

Attributed Central Order (ACO)

E > D > A > B > C

concordant

B C E

unordered objects

sorting function

E > B > C

concordant with ACO

Supervised Ordering: a task for acquiring a sorting function form given sample orders; specifically, objects are represented by attributed vectors

Cohen's Method [Cohen 99]

Learning: Probability for estimating orders of object pairs based on attribute values

sample orders		decompose into ordered pairs		learn probability
$A \succ B \succ C$ $D \succ E \succ B \succ C$ $A \succ D \succ C$	$\Rightarrow$	$A \succ B, A \succ B, B \succ C$ $D \succ E, D \succ B, D \succ C, \cdots$ $A \succ D, A \succ C, D \succ C$	$\Rightarrow$	$\Pr[x \succ y \mid x, y]$

Sorting: sort unordered objects based on the learned probability function

candidate order		decompose into ordered pairs
$A \succ B \succ C$	$\Rightarrow$	$\mathrm{Pair} = \{A \succ B, A \succ B, B \succ C\}$

objective function: $$\sum_{x \succ y \in \mathrm{Pair}} \Pr[x \succ y \mid x, y]$$

Find the order maximize the objective function among candidate orders

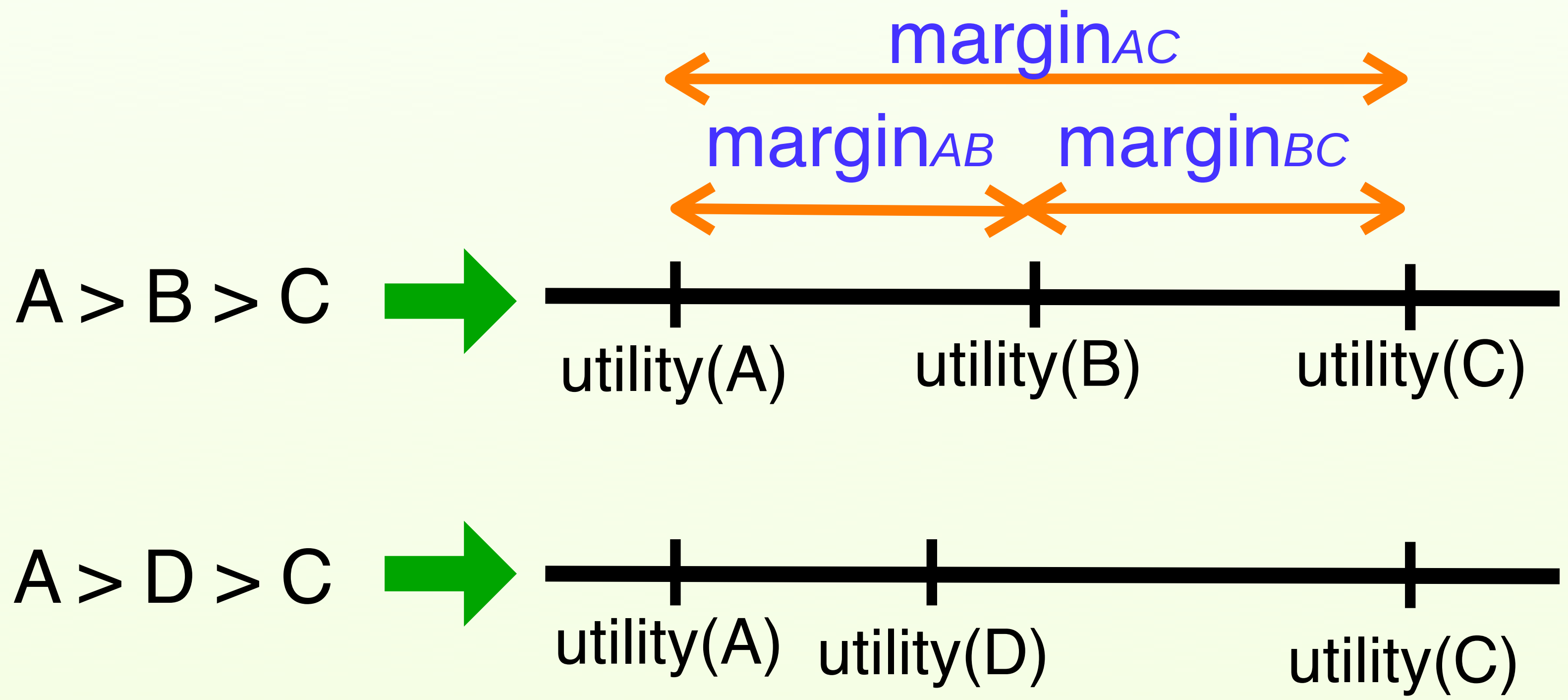
SVM-Based Method (SVOR)

Learning: Find a utility function that maximally separates preferred from non-preferred

SVOR **(Support Vector Ordinal Regression)** [Herbrich 98]

- SVM-like formulation & Kernel ready

sample orders | utility & margin | Objective



maximize:

$$\sum_{X,Y} \mathrm{margin}_{XY}$$

Sorting: sort unordered objects according to their utilities

SVM-Based Method (Order SVM)

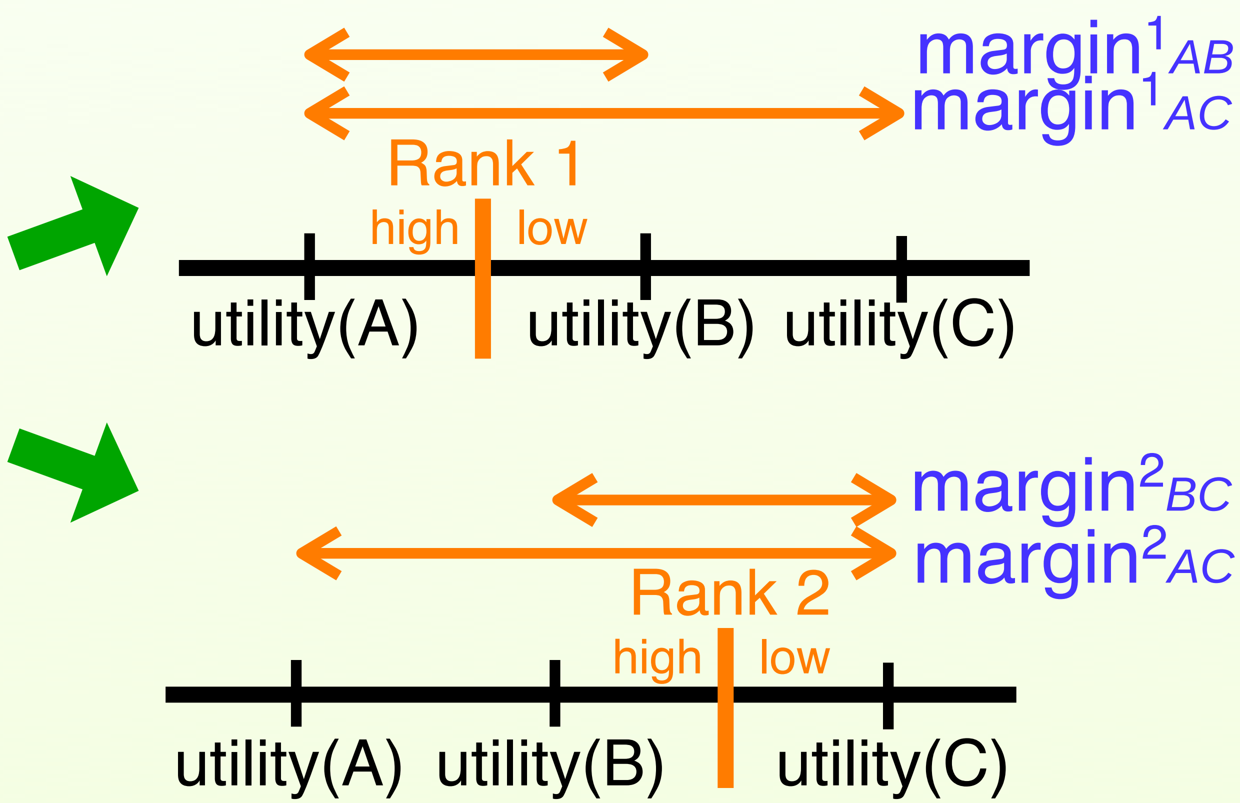
Learning: Find a utility function which maximally separate higher-ranked from lower-ranked on average

Order SVM [Kazawa 03]

- Rank as category & SVM-like formulation & Kernel ready

sample orders | utility & margin | Objective

A > B > C



maximize:

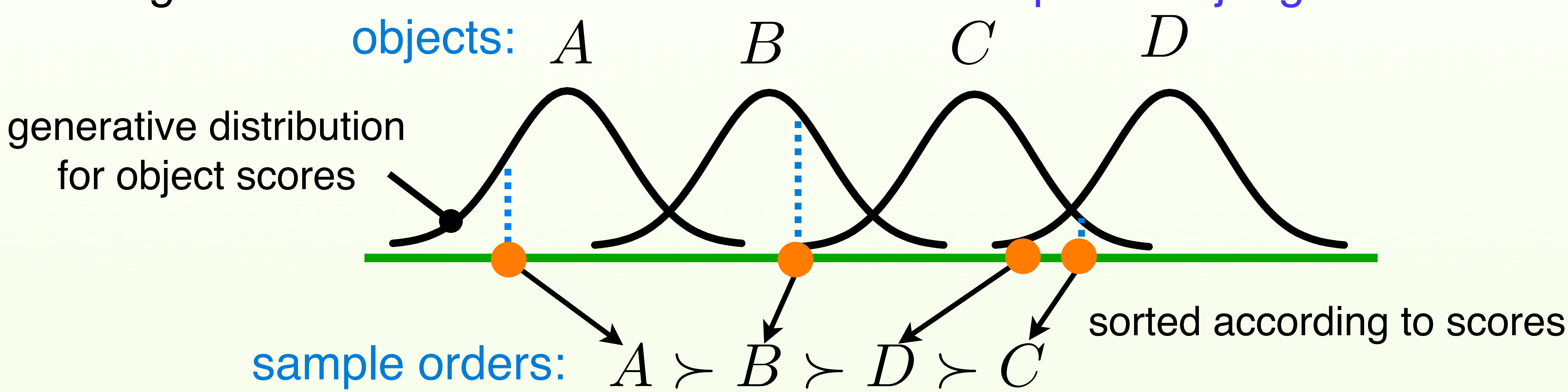
$$\sum_j \sum_{X,Y} \mathrm{margin}^j_{XY}$$

Sorting: sort unordered objects according to their utilities

Thurstone Based Method

Learning: acquire utility function to describe the degree to be ranked higher

generation model "Thurstone's law of comparative judgment"



Given sample orders are combined based on this model

by min MSE **Thurstone Regression (TR)** [Kamishima 02]

by ML **Attributed Thurstone Model (ATM)** [Akaho 02]

Utility is a linear function to estimate ranks of combined order

Sorting: sort unordered objects according to their utilities

Experiments

Data Generation Procedure

1. for each object, randomly generate attribute vector
2. generate true ACO according to the utility function:

$$\mathrm{utility}(x_j) = (1 + \sum_l^k w_l x_{jl})^{\mathrm{dim}}$$

3. as sample orders, generate sub-orders of the ACO

true central order

$A \succ B \succ C \succ D \succ E \succ F$

sample orders

$A \succ C \succ D \quad B \succ C \succ E \succ F$

Test Procedure: Cross Validation

1. divide sample orders into a training set and a test set
2. learn sorting function from training set
3. compare the estimated order derived by sorting function and the true ACO in the test set

Attribute Noise & Order Noise

Order Noise

noiseless

true ACO

$A \succ B \succ C \succ D$

On sampling objects from true ACO
orders among objects are preserved

sample order

$A \succ B \succ D$

with noise

true ACO

$A \succ B \succ C \succ D$

objects are
sampled and permutated

sample order

$D \succ B \succ A$

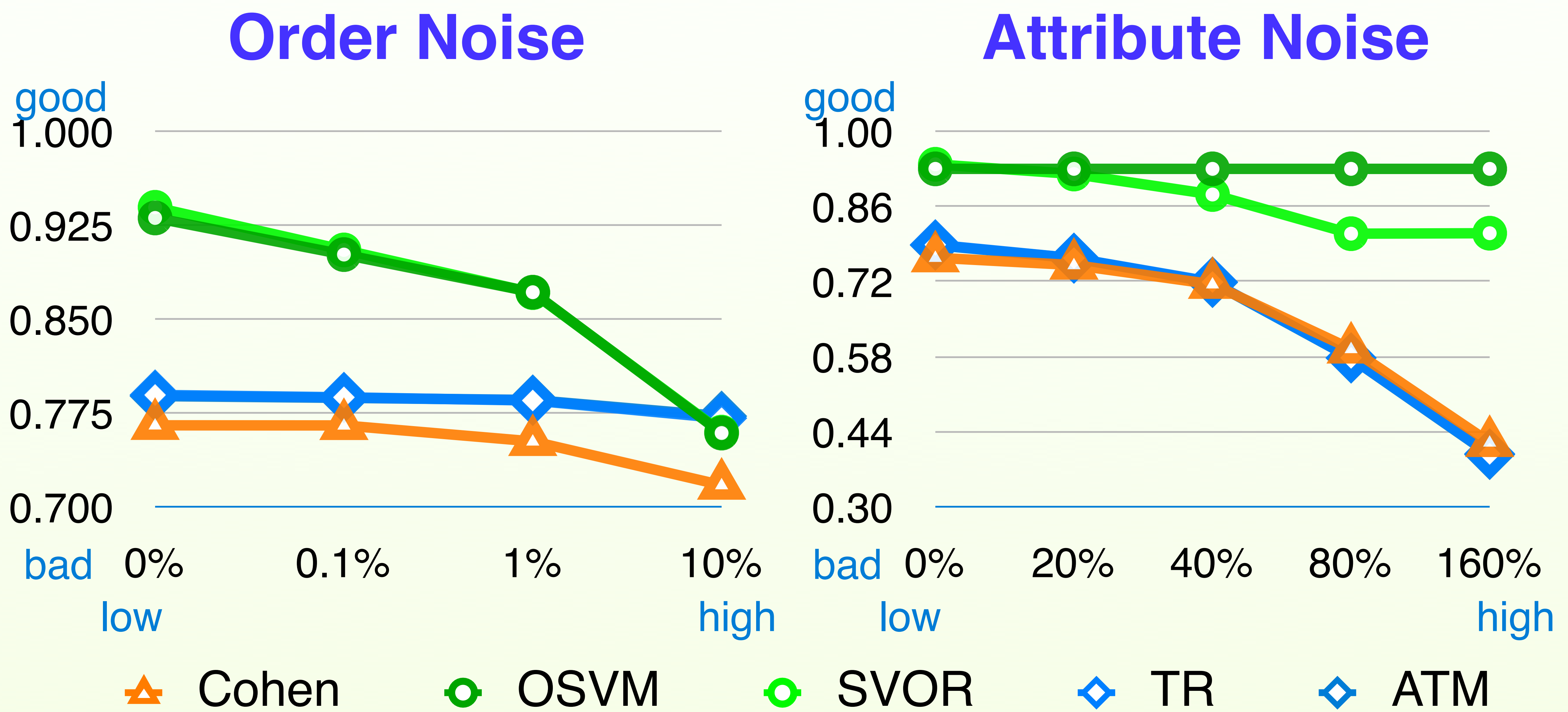
randomly selected adjacent objects
are permutated

Attribute Noise

noiseless keep original attribute values $x_i = (x_{i1}, \ldots, x_{ik})$

with noise add noise
following normal distribution $x'_{ij} = x_i j \times N(1, \sigma)$

Attribute Noise and Order Noise (Result)



Vertical: estimation precision Horizontal: noise level

Non-SVM-based ➡ robust against order noise

SVM-based ➡ robust against attribute noise

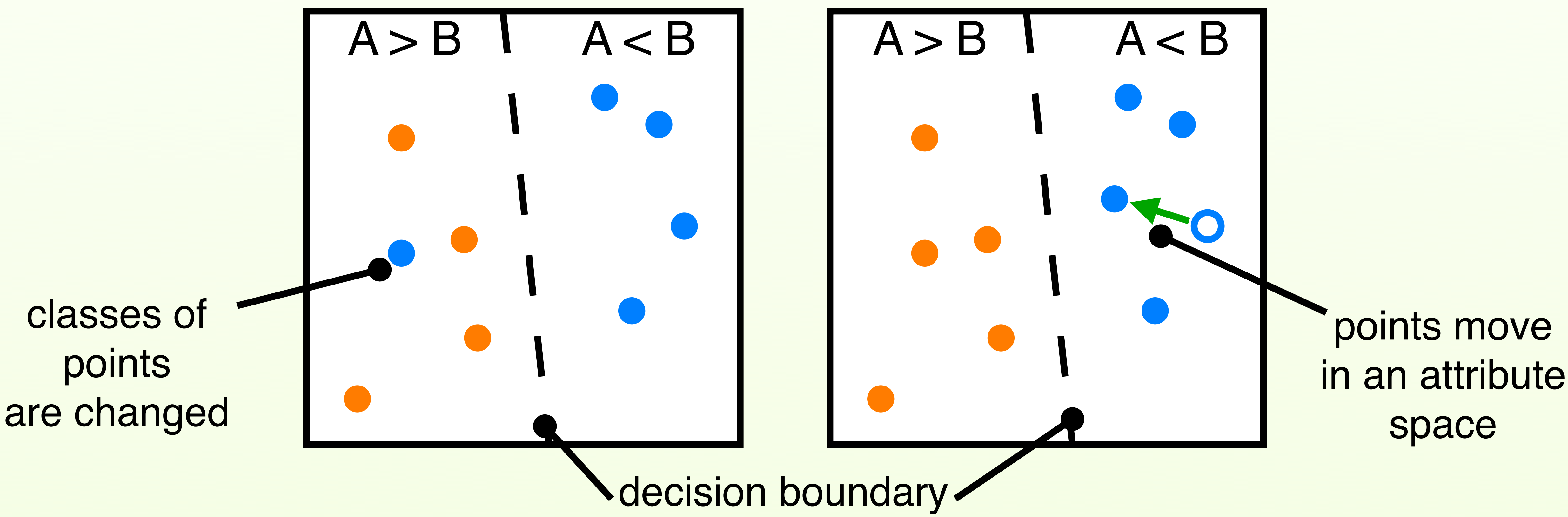
SVM-based Methods & Noise

Basically, analogous to the SVM for classification

OSVM: ranked higher than j-th or not

SVOR: which object in a pair is ranked higher

Order Noise **Attribute Noise**



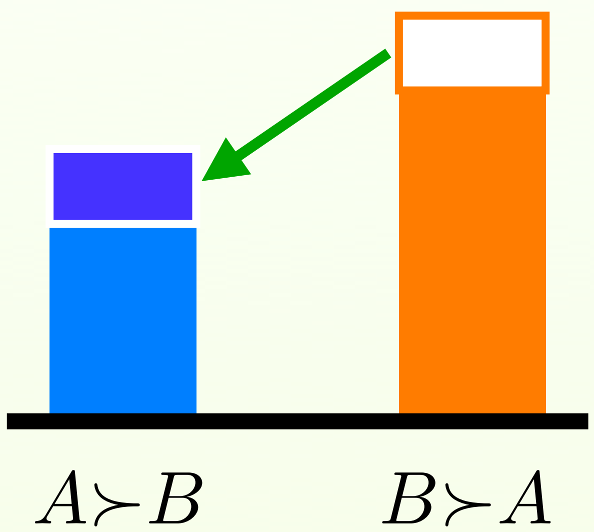
Changed points become support-vectors with high probability, and seriously affect

Never affect, if moving within decision boundary

Thurstone-based Methods & Noise

frequency of ordered pairs A>B and B<A at a specific position in attribute space

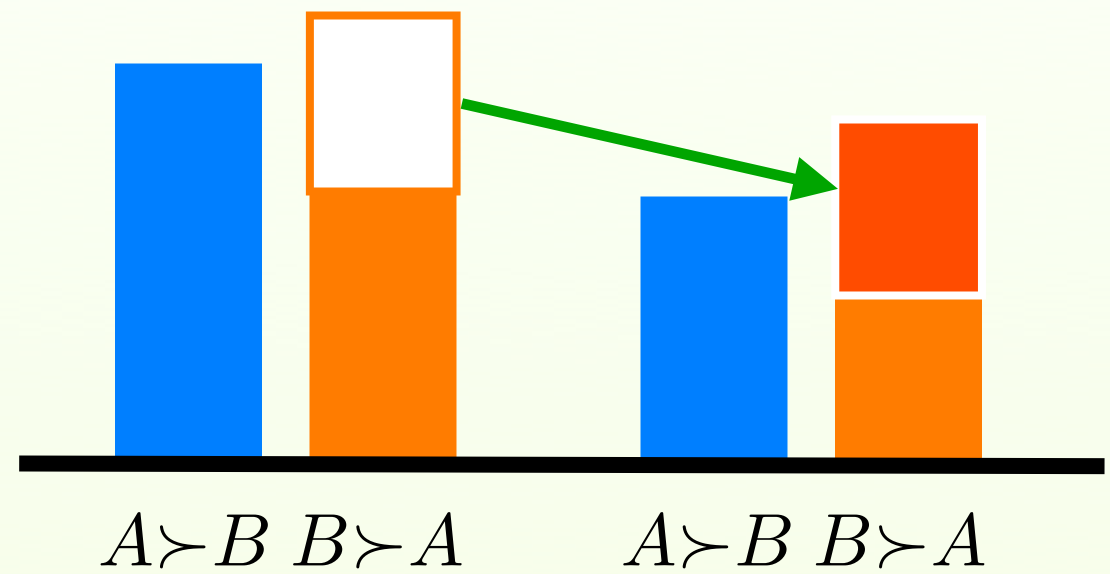
Order Noise



Instances are moved from B>A to A<B at the same position in attribute space

Results are not affected, if majority between A>B & B<A do not change

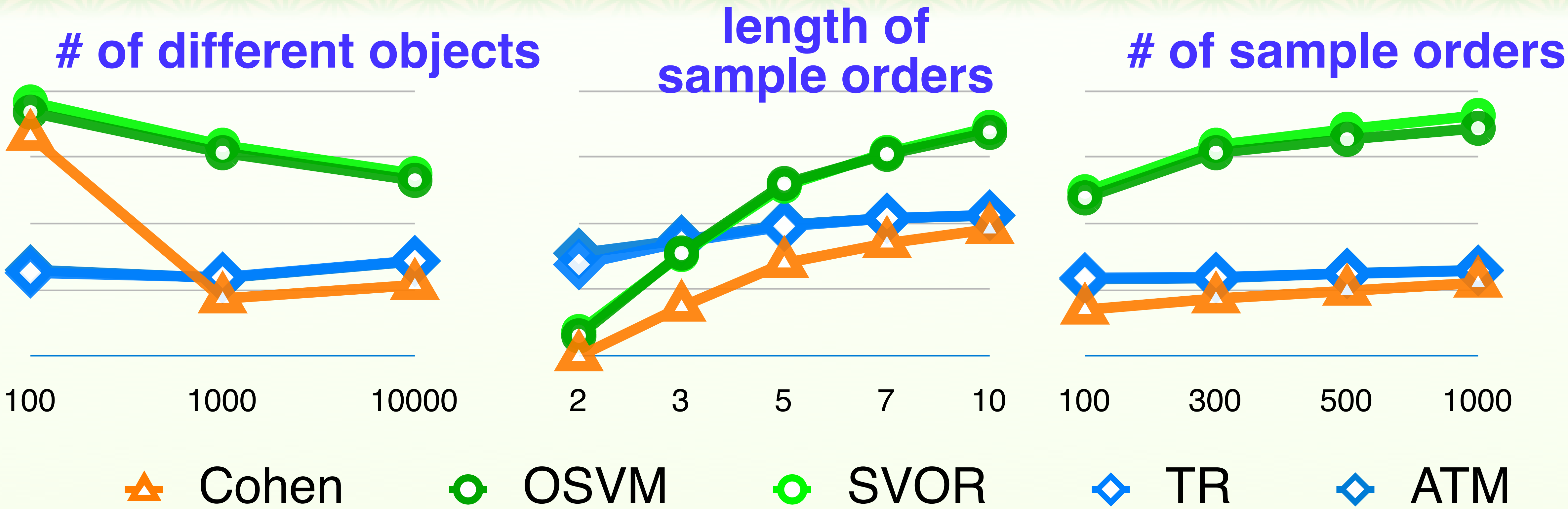
Attribute Noise



Instance of B>A moves to another position in attribute space

Results are affected, unless the majorities don't change at the source and the destination position

Other Learning Parameters



summary of weak points	SVM-Based low model bias	Non-SVM-Based high model bias
many kinds of objects need for high generalization ability	If: few # of objects ×: overfitting	If: many # of objects ×: insufficient fit
short/few vs long/many sample orders	If: short/few samples ×: insufficient information	If: long/many samples ×: insufficient use of information

Summary (1)

- **Cohen's Method**
  - Low estimation precision, but the fastest
  - Suit for on-line learning
  - Low bias models can be applied for learning posterior probabilities, but such a model may increase computational complexity
- **SVM-Based Methods (OSVM & SVOR)**
  - High estimation precision, but the slowest
  - (# of orders)×(order length)^2 are limited to 10^5 - 10^6
  - Robust against attribute noise, not against order noise
  - It is able to use high-bias models by changing kernel functions, but computational complexity cannot be reduced

Summary (2)

- **Thurstone-Based Methods (TR & ATM)**
  - estimation precision and computational complexity are medium
  - This method can be applied even if # of samples are many, # of different objects are limited to 10^5 - 10^6
  - Low bias models can be applied for regression, but such a model may increase computational complexity
  - TR and ATM methods are comparable in estimation precision while the ATM requires additional computation, so the TR is preferred
- **Future Woks**
  - Explore the effects of the tuing options of model bias
  - Test on another real data set in which # of objects is large to evaluate the methods' generalization abilities

Bibliography

[Cohen 99] W.W.Cohen, R.E.Schapire, Y.Singer : Learning to order things. Journal of Artificial Intelligence Research 10 (1999) 243–270

[Kazawa 03] H.Kazawa, T.Hirao, E.Maeda: Order SVM: A kernel method for order learning based on generalized order statistic. The IEICE Trans. on Information and Systems, pt. 2 J86-D-II (2003) 926–933 (in Japanese)

[An English version will appear in “Systems and Computers in Japan” Wiley Periodicals, Inc.]

[Herbrich 98] R.Herbrich, T.Graepel, P.Bollmann-Sdorra, K.Obermayer: Learning preference relations for information retrieval. In: ICML-98 Workshop: Text Categorization and Machine Learning. (1998) 80–84

[Kamishima 02] T.Kamishima, S.Akaho: Learning from order examples. In: Proc. of The IEEE Int’l Conf. on Data Mining. (2002) 645–648

[Akaho 02] S.Akaho, T.Kamishima: A statistical approach for learning from order examples linear models. In: Proc. of the 16th Annual Conference of JSAI. (2002) (in Japanese).

More Information

Toshihiro Kamishima Home Page

http://www.kamishima.net/