Learning user preferences from user ratings in a web environment

Alan Eckhardt

Outline

- Motivation
- Input from the user
- Preference model learning
- Specific settings for preference learning
- Experiments
- Conclusion

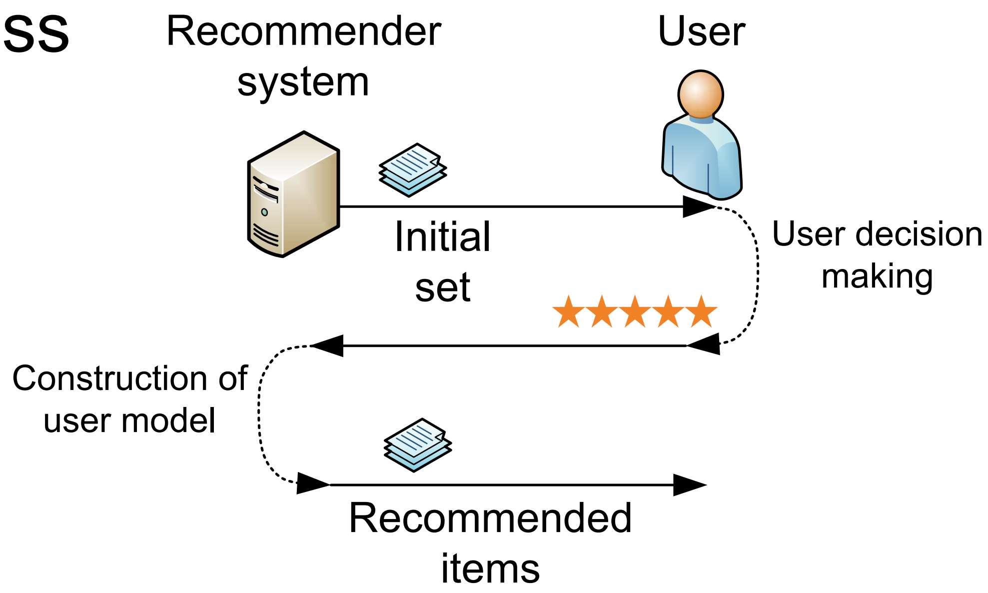
Motivation

- Helping the user to find what she looks for
  - E.g. notebooks
- A small amount of information required from the user
  - Ratings of notebooks,...

- Construction of a general user preference model
  - Each user has his/her own preference model
- Recommendation of the top *k* notebooks to the user
  - Which the preference model has chosen as the most preferred for the user

Motivation

- Recommendation process
- Initial set
  - Centers of clusters of objects
- Construction of user model
- Recommendation
- More iterations possible
  - In each iteration the user model is refined

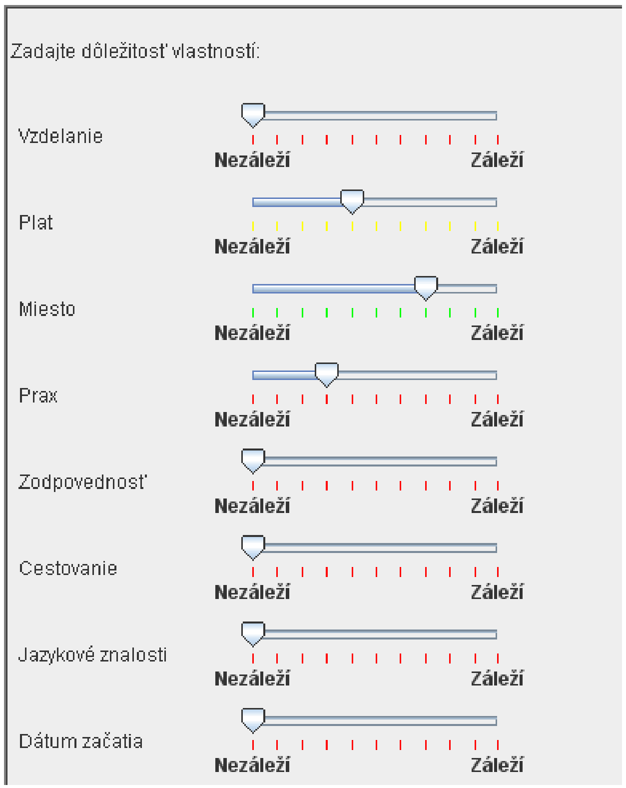


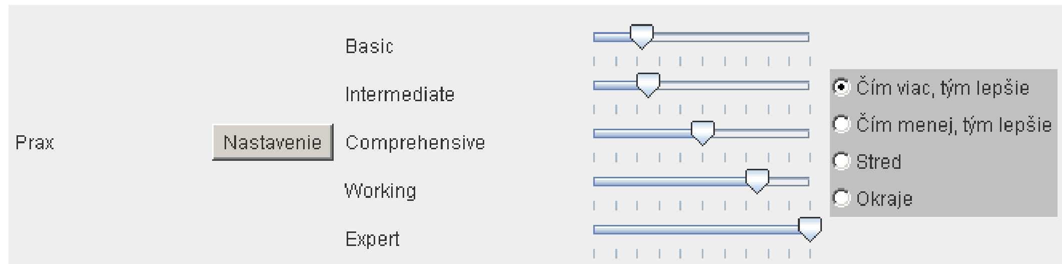
Input from the user

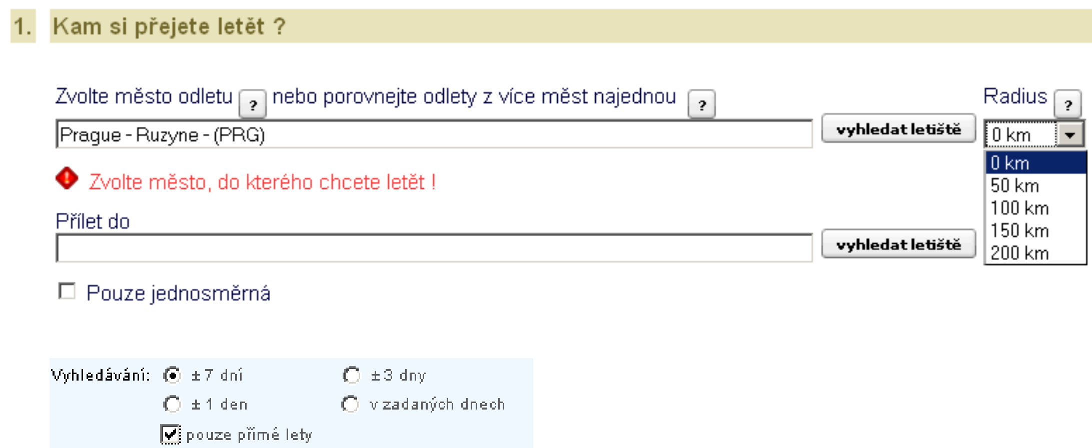
- Direct specification of preferences
  - User directly describes her preferences
    - Using a preference model
  - User has to know the preference model
  - Most difficult for the user

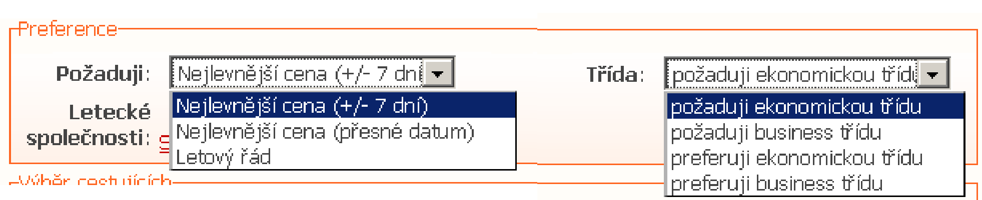
Direct specification of preferences

Preferencie používateľa









Input from user

- User behaviour
  - No user effort
  - Very noisy data, no explicit information, may contain very complex structure, different actions
  - Filtering the objects set by some attribute value, staying on a page with product details, clicking on a link to product details, clickstream, recommending product to someone, ...

Input from user

- User ratings
  - A widely use approach for expressing preferences

User preference model

- Content based
  - What user prefers
- Used to evaluate all objects
  - In order to get the top *k*
- Learned from the user evaluation of a small set of objects
- Notation
  - $A_1,\ldots,A_N$ - attributes
  - $r(o)$ – rating of object o
  - o_i – i-th attribute value of o

Two step user model

- User model is divided into two steps

1. **Local preferences** - normalisation of the attribute values of objects to their preference degrees

   $f_i : D_{A_i} \rightarrow [0,1]$

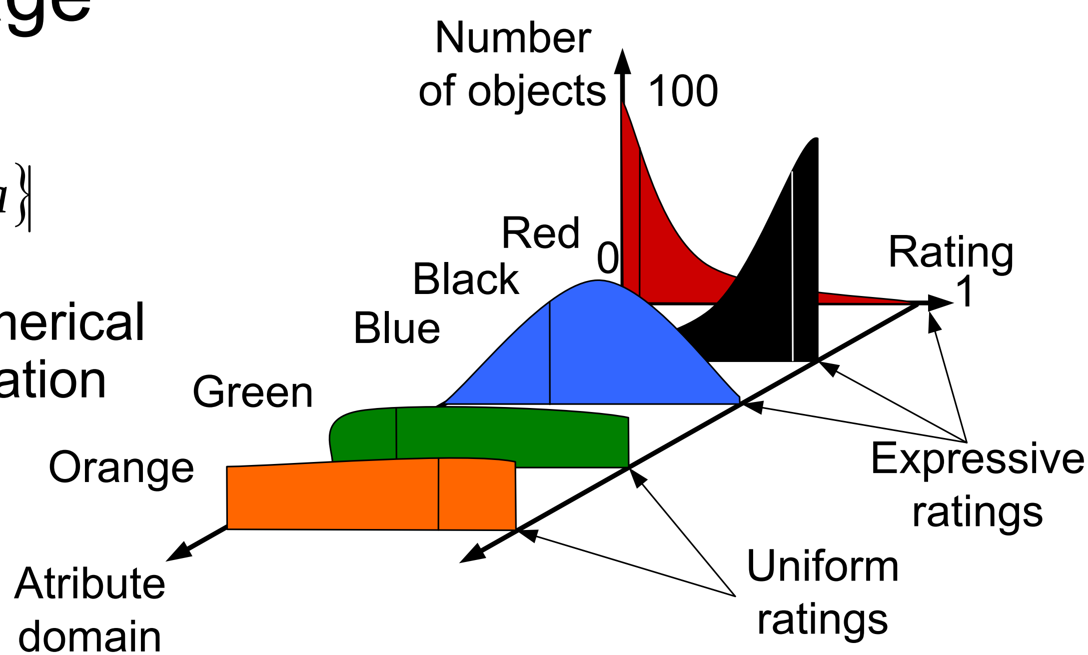
   Transforms the space ΠD_{A_i} into $[0,1]^N$

2. **Global preferences** - aggregation of preference degrees of attribute values into the predicted rating

   $@ : [0,1]^N \rightarrow [0,1]$

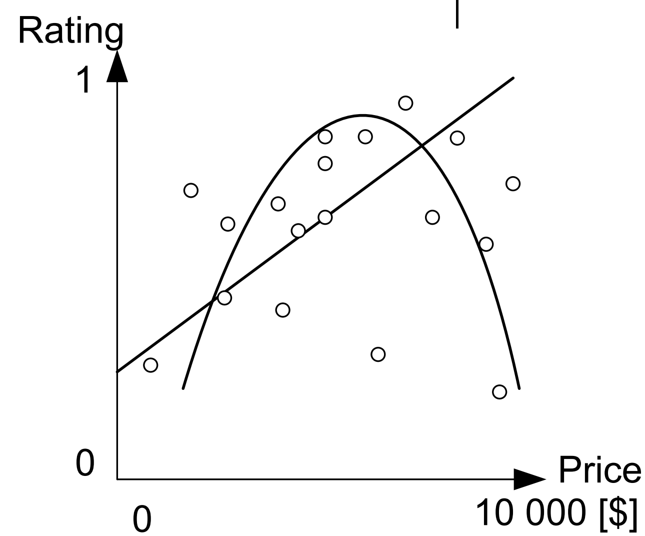
Preference model learning

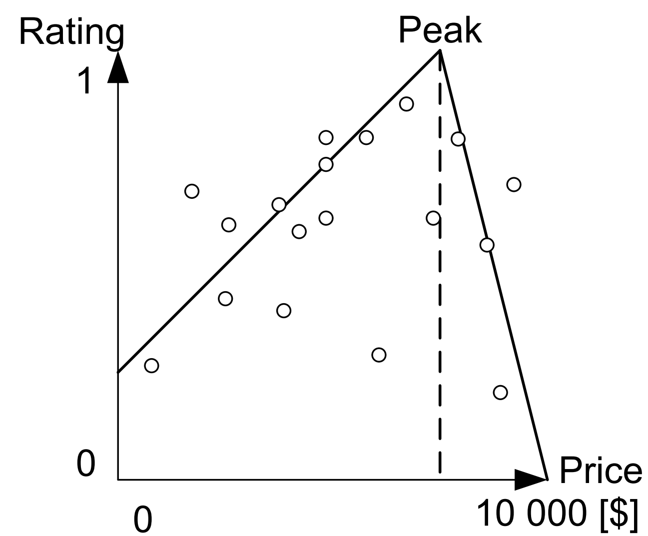
- Nominal attributes
  - Examine distribution of the ratings of the objects with a particular attribute value
    - “ASUS notebooks”
  - Take the average
    - Or median,...
  - $f_i(a) = \sum_{o|o_i=a} r(o) / \left|\{o \mid o_i = a\}\right|$
  - May be applied to numerical attributes via discretisation



Preference model learning

- Numerical attributes
  - Linear regression
    - $f_i(x)=ax+b$
  - Quadratic regression
    - $f_i(x)=ax^2+bx+c$
  - Peak
    - $f_1(x)=a_1x+b_1$
    - $f_2(x)=a_2x+b_2$
    - Peak is found by trying all values and taking the best option
      - Costly approach





Preference model learning

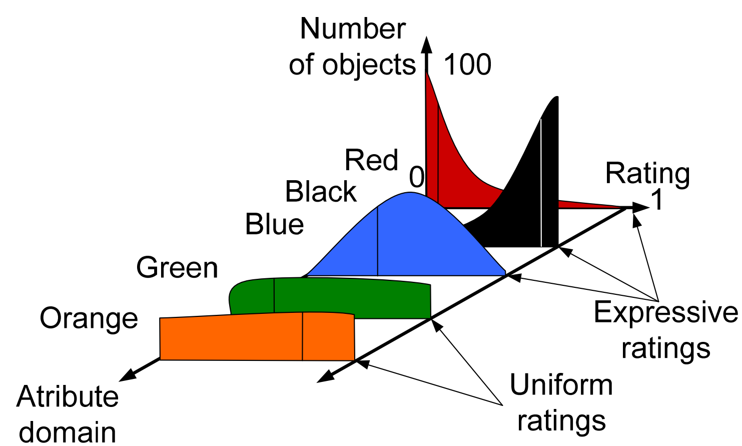
- Statistical
  - Weighted average aggregation

$$@\left(\mathrm{MPix\_U}_1, \mathrm{Fast\_U}_1, \mathrm{Cheap\_U}_1\right) = \frac{5 * \mathrm{MPix\_U}_1 + 1 * \mathrm{Fast\_U}_1 + 3 * \mathrm{Cheap\_U}_1}{9}$$

$$@(o) = \sum_{i=1,\dots,N} w_i f_i(o_i) \Big/ \sum_{i=1,\dots,N} w_i$$

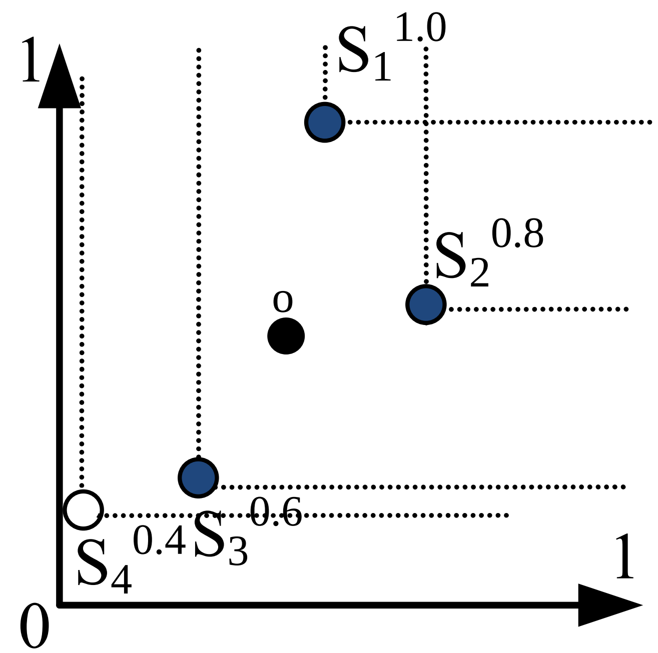
$$w_i = \sum_{a \in A_i} 1/\mathrm{var}\left(r(a)\right) / \left|A_i\right|$$

  - *r(a) = {r(o)|o$_i$=a}*
    - ratings of objects having attribute value *a*



Preference model learning

- Instances
  - Based on k-NN approach
  - Use monotone space $[0,1]^N$
    - Notion of dominance
  - Skyline
    - A set of incomparable objects, dominating *o*
    - or dominated by *o*
  - (The lowest rating from the upper skyline + the highest rating from the lower skyline) / 2



Specific settings for preference learning

- Small training set
  - Users are not willing to rate too many objects
- Order is important
- Ratings on a discrete scale 1,...,5
- Only the best objects are of interest
  - Bad prediction for bad objects in not tragic
  - Bad prediction for good objects is much worse

A	B	C
1	2	1
2	1	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	10
10	10	9

Specific settings for preference learning - Error measures

- RMSE
  - $RMSE(\hat{r}) = \sqrt{\sum_{o \in Test} \left(\hat{r}(o) - r(o)^2 \right) / |Test|}$
- Tau coefficient
  - similarity of two ordered lists $\tau = \dfrac{n_c - n_d}{1/2n(n-1)}$
  - count of non-reversed pairs
- Pearson correlation
  - captures linear dependence of two variables
- Build time
  - Time to build the classifier
- Test time
  - Time to evaluate objects from the test set

Specific settings for preference learning

- F1 Score
  - Percentage of common top 20 object in method’s and user’s lists
- Weighted RMSE/Tau
  - Weighting the error with the user ratings
  - More attention to better objects
- Monotonicity violation
  - Reversed order of two objects – 3 points, two equal objects made non-equal – 1 point
- Unpredicted objects count
  - Count of objects, which the method failed to evaluate

Experiments

- Statistical for Collaborative filtering
- Local preferences as preprocessing for UTA method and for ILP

Statistical for Collaborative filtering

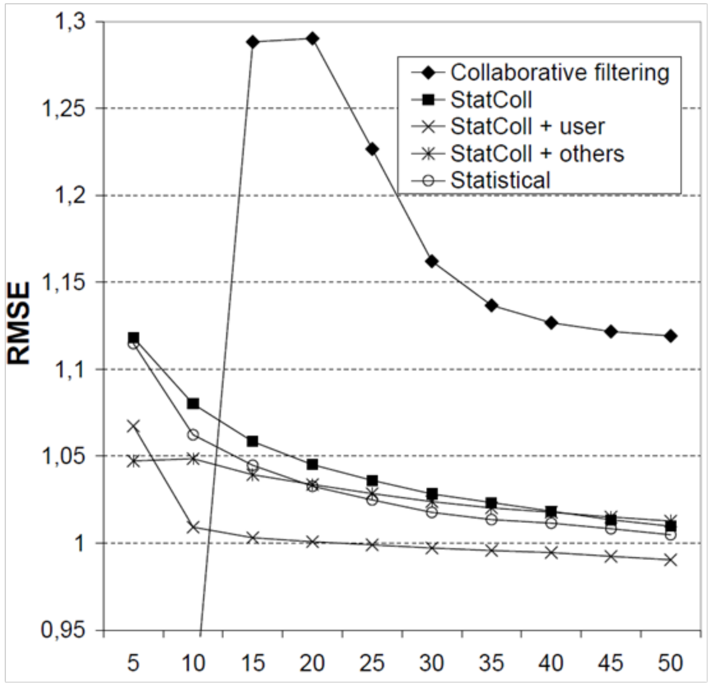
- Collaborative filtering
- Determining how user u evaluates object *o*
- Using how *o* was rated by other users
  - Similarity of users
  - Take into account only similar users’ ratings
  - Users are similar, if they rated objects similarly
- Similarity of items
  - Take into account also the ratings of similar items
  - Objects are similar, if they were rated similarly by user *u*

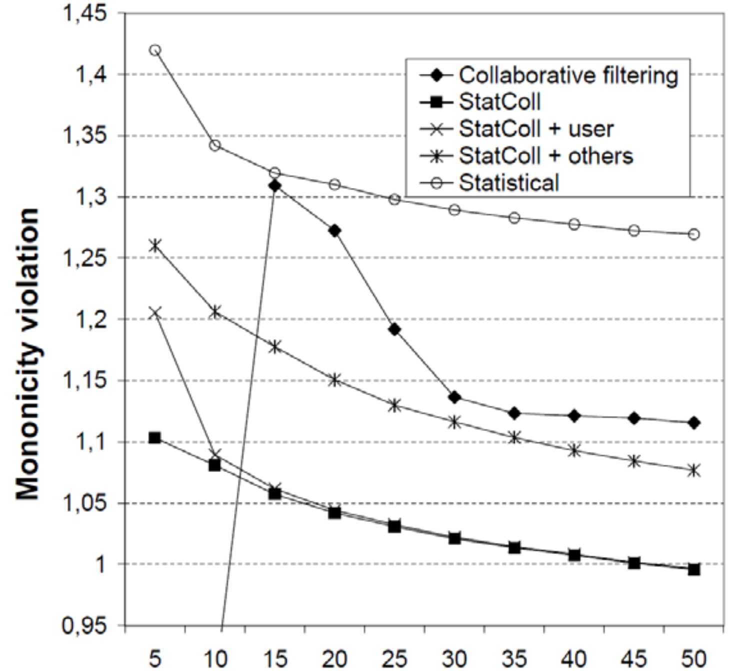
Statistical for Collaborative filtering

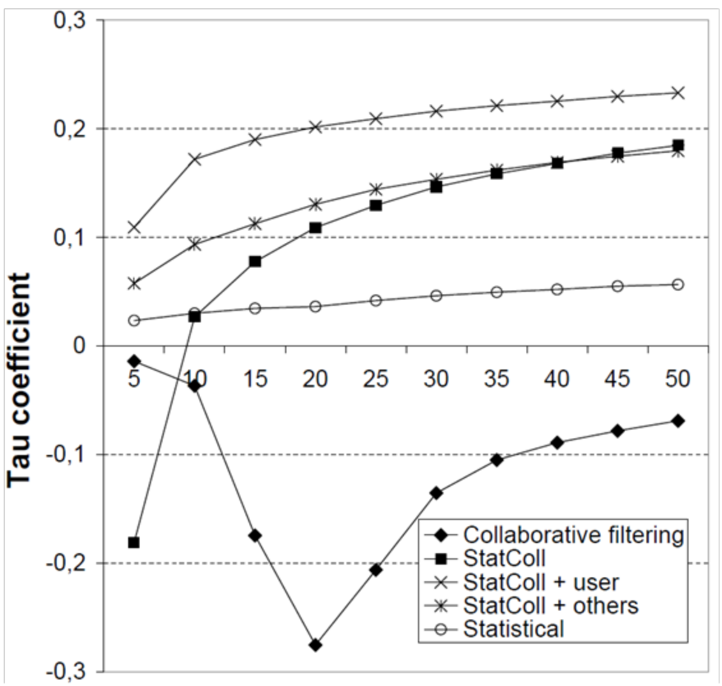
- Using similarity of preference models as distance for collaborative filtering
  - Linear = a^1x+b^1, a^2x+b^2
    - $(3*|a^1-a^2|+|b^1-b^2|)/4$ – more weight to the slope
  - Nominal
    - $r^1(a)-r^2(a)$
  - Aggregation – weighted average
    - $|w_i^1-w_i^2|$

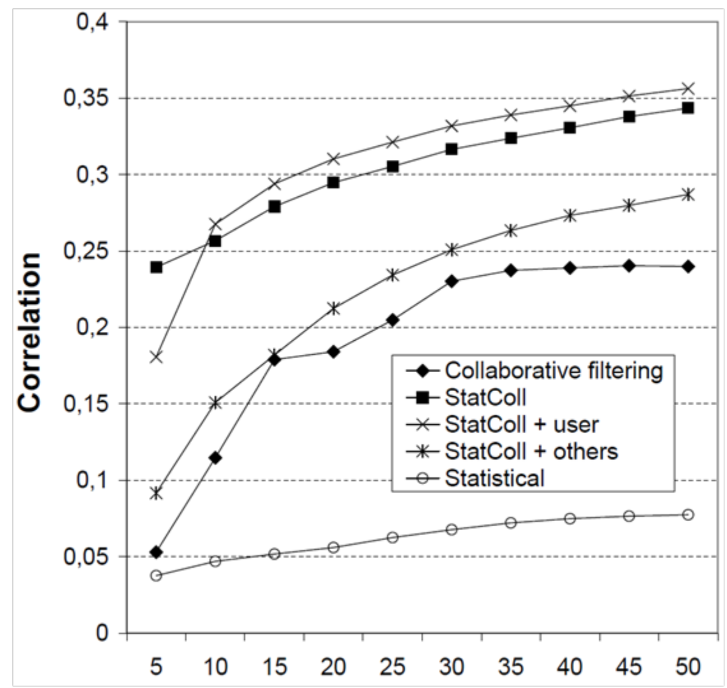
Statistical for Collaborative filtering

- Experiment settings
  - Netflix dataset, only 1000 users used
  - Each user had 200 ratings
    - 1-5 rating scale
  - Training set sizes 5-50
    - Standard collaborative filtering requires much more data – both users and ratings

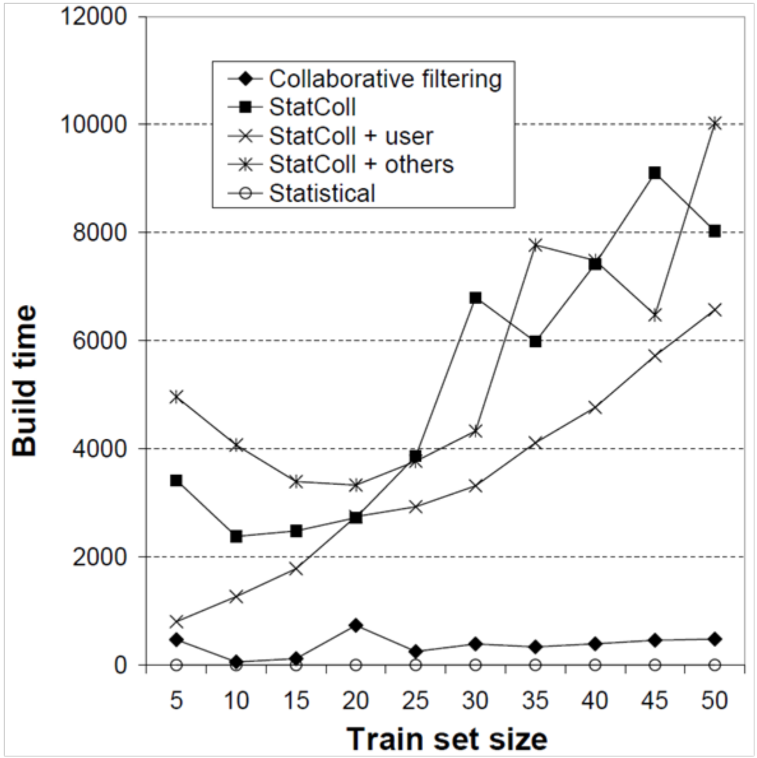

RMSE
1,3
1,25
1,2
1,15
1,1
1,05
1
0,95
5 10 15 20 25 30 35 40 45 50
Collaborative filtering
StatColl
StatColl + user
StatColl + others
Statistical

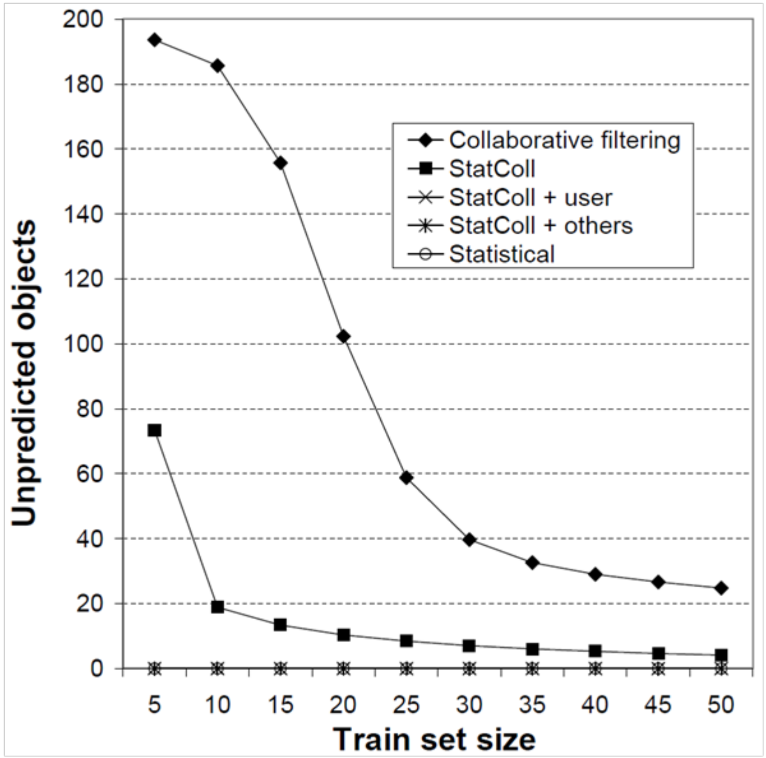

Mononicity violation
1,45
1,4
1,35
1,3
1,25
1,2
1,15
1,1
1,05
1
0,95
5 10 15 20 25 30 35 40 45 50
Collaborative filtering
StatColl
StatColl + user
StatColl + others
Statistical


Tau coefficient
0,3
0,2
0,1
0
-0,1
-0,2
-0,3
5 10 15 20 25 30 35 40 45 50
Collaborative filtering
StatColl
StatColl + user
StatColl + others
Statistical


Correlation
0,4
0,35
0,3
0,25
0,2
0,15
0,1
0,05
0
5 10 15 20 25 30 35 40 45 50
Collaborative filtering
StatColl
StatColl + user
StatColl + others
Statistical

Statistical for Collaborative filtering





Preprocessing for UTA

- UTA works with criteria
  - Attributes already sorted according to preference
- But real data are often not ordered
  - User would have to specify his preferences for attributes explicitly
    - Too much effort
    - Display size – no perfect ordering for every user
- Using local preferences eases the effort of the user

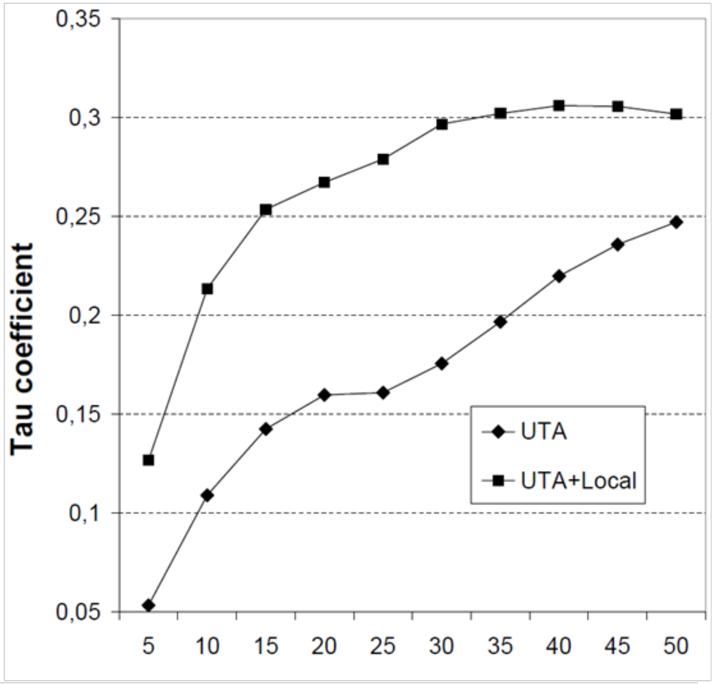
Preprocessing for UTA

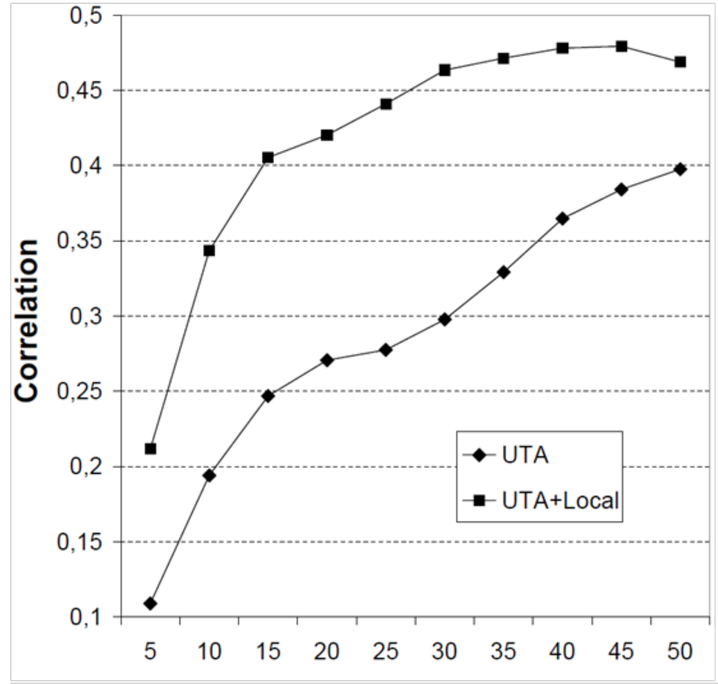
- Implementation of Tomas Kliegr used
  - http://kliegreen.cz/uta/vw.html
  - Non monotonicity allowed in criteria
  - Possible changes of slope for a criterion set to 2
- First variant (UTA) trained on the data
- Second variant (UTA + local) trained on $[0,1]^N$ monotone space

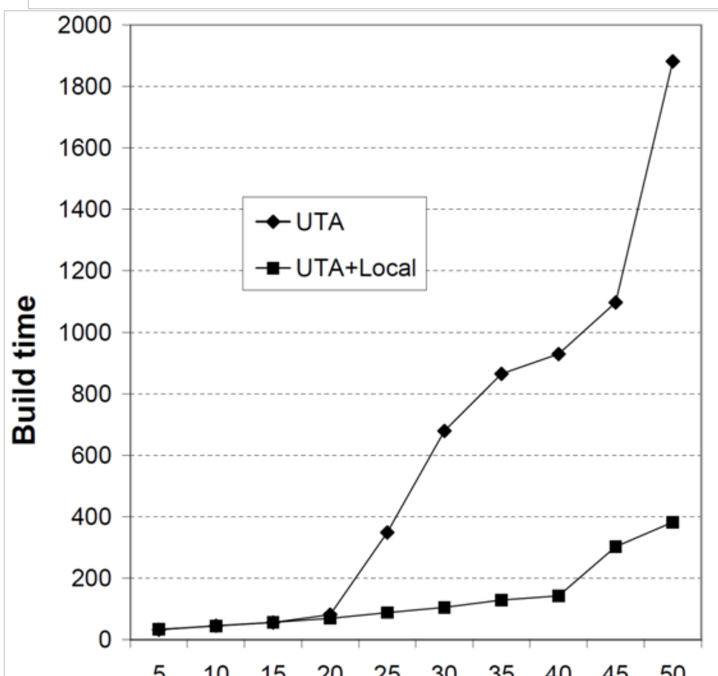
Preprocessing for UTA

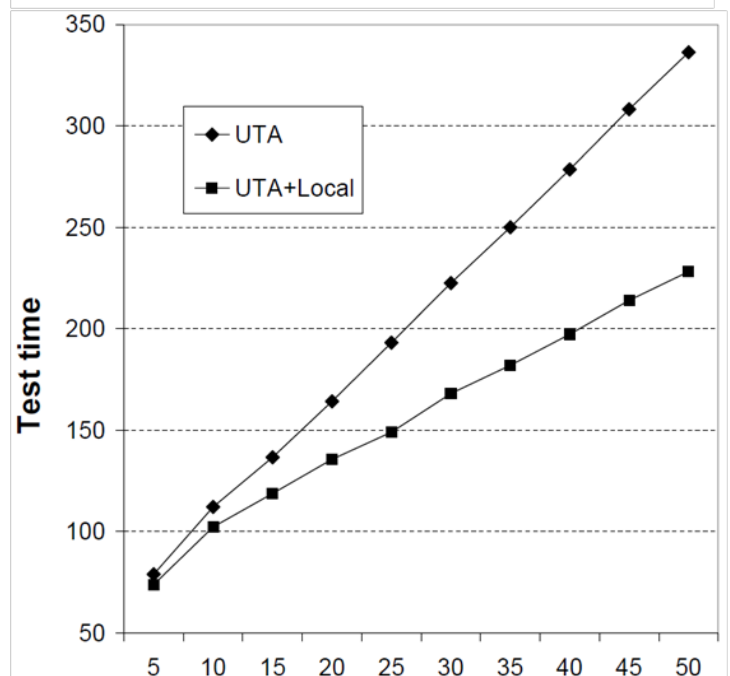
- Experiment settings
  - UCI datasets for classification
    - with monotone class variable
      - transformed to 1-5 ratings
    - reduced to 200 objects
  - Artificial preferences on real data
    - Notebooks and autobazar

  - 5-50 training sets


Tau coefficient
0,35
0,3
0,25
0,2
0,15
0,1
0,05
5
10
15
20
25
30
35
40
45
50
UTA
UTA+Local


Correlation
0,5
0,45
0,4
0,35
0,3
0,25
0,2
0,15
0,1
5
10
15
20
25
30
35
40
45
50
UTA
UTA+Local


Build time
2000
1800
1600
1400
1200
1000
800
600
400
200
0
5
10
15
20
25
30
35
40
45
50
UTA
UTA+Local


Test time
350
300
250
200
150
100
50
5
10
15
20
25
30
35
40
45
50
UTA
UTA+Local

Preprocessing for ILP

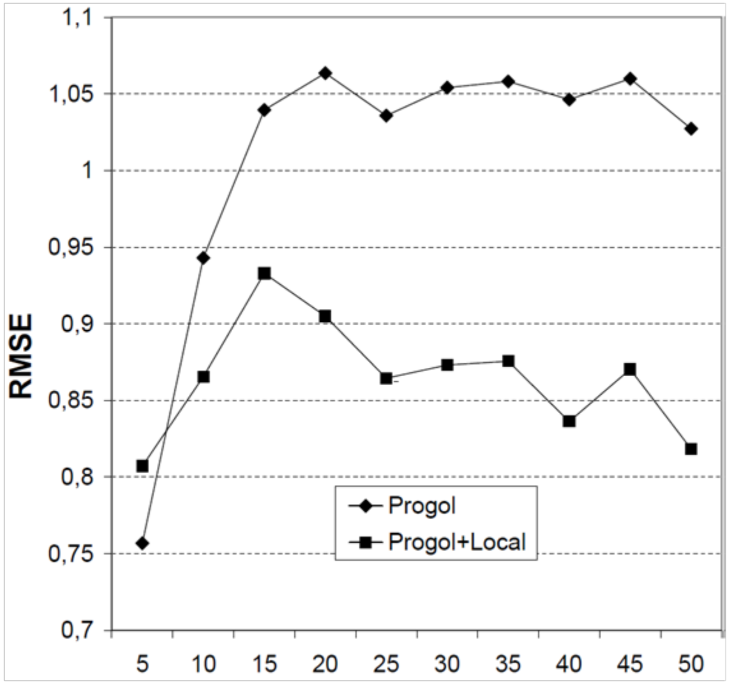
- Preprocessing for ILP
  - Inductive logic programming
  - Method for finding rules in complex data
    - Prolog programs used
  - Progol implementation was used
    - http://www.doc.ic.ac.uk/~shm/progol.html

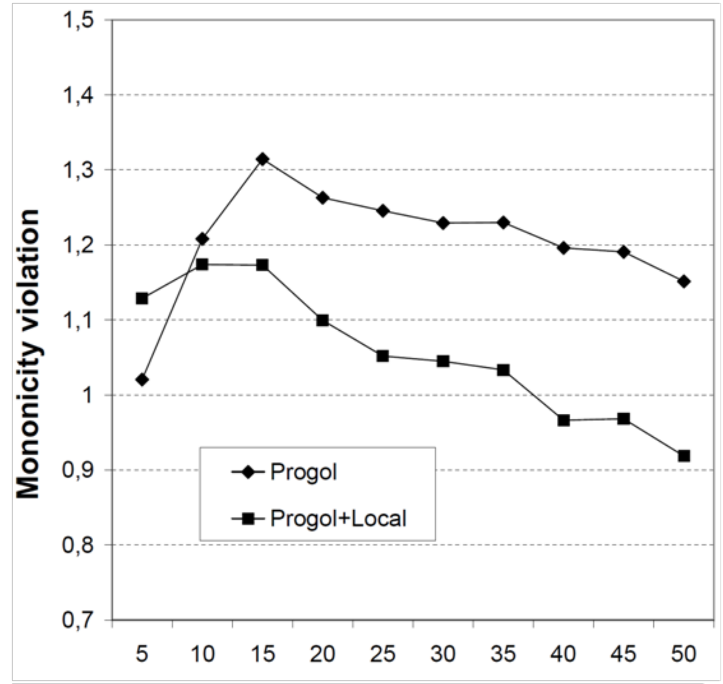
```
rating(30, '2').
make(30, 'renault').
bodywork(30, 'combi').
originplace(30,
'francuzsko').
originyear(30, '2003').
price(30, '264000.0').
crashed(30,
'nehavarovane').
runnedkm(30,
'118000.0').
owner(30, '1.0').
fuel(30, 'diesel').
...
```

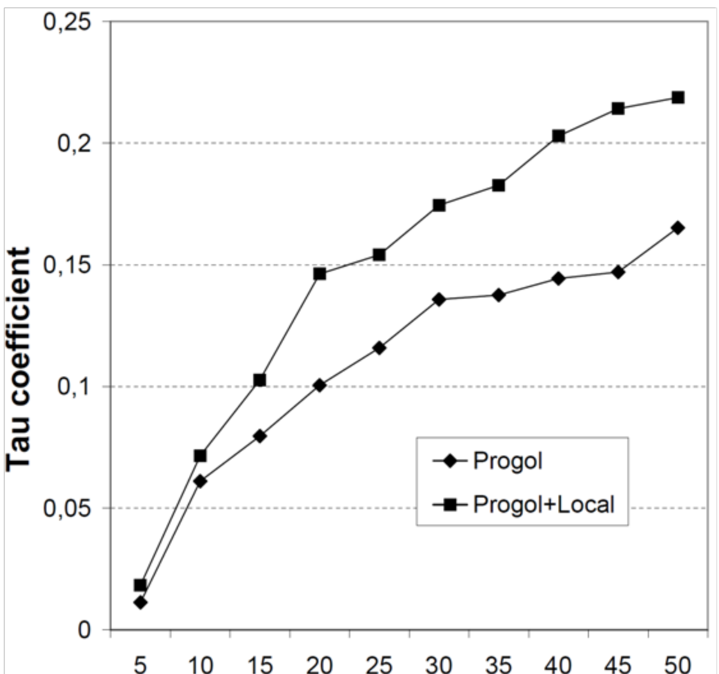
```
rating(A,2) :- owner(A,2.0).
rating(A,3) :- originplace(A,korea).
rating(A,3) :- horsepowerkw(A,96.0).
rating(A,3) :- horsepowerkw(A,103.0).
rating(A,3) :- airbags(A,10.0).
rating(A,4) :- bodywork(A,limuzina).
rating(A,4) :- doors(A,4), fuel(A,benzin), safetydrive(A,esp).
```

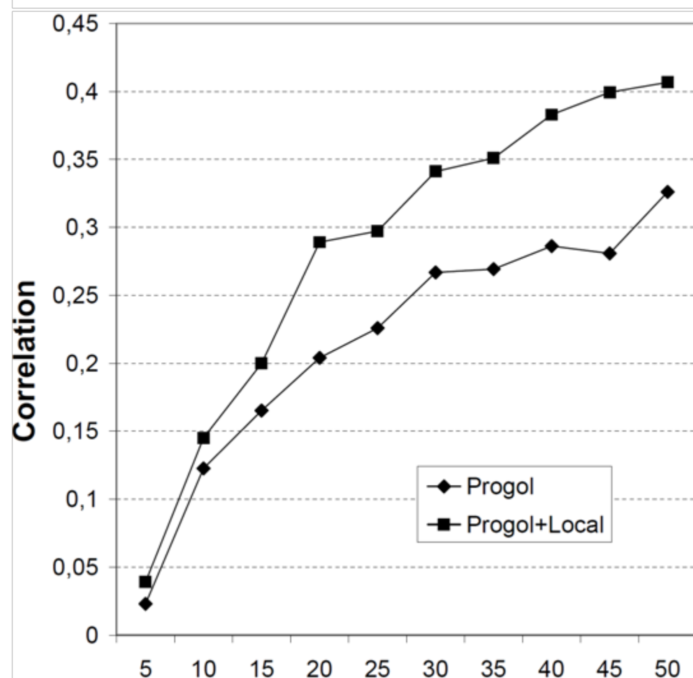
Preprocessing for ILP

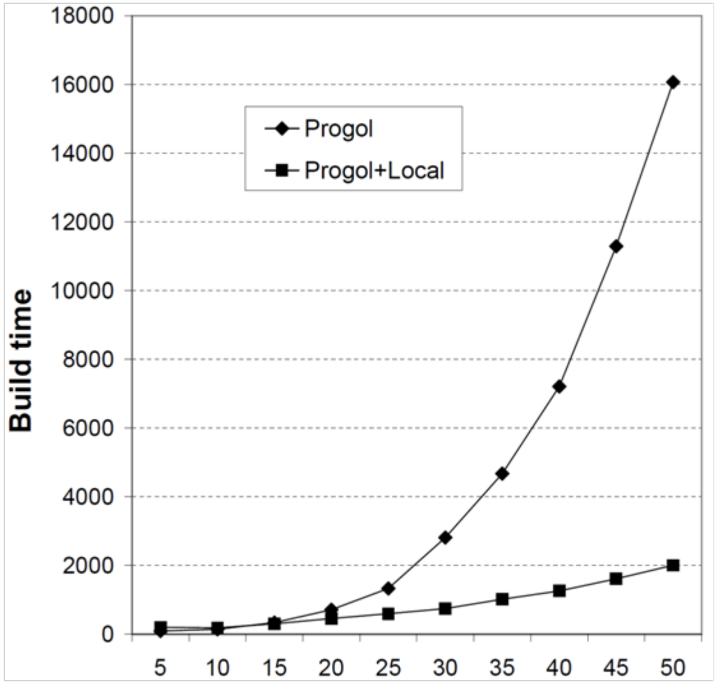
- Same settings as for UTA
  - UCI datasets for classification
    - with monotone class variable
      - transformed to 1-5 ratings
    - reduced to 200 objects
  - Artificial preferences on real data
    - Notebooks and autobazar
  - 5-50 training sets

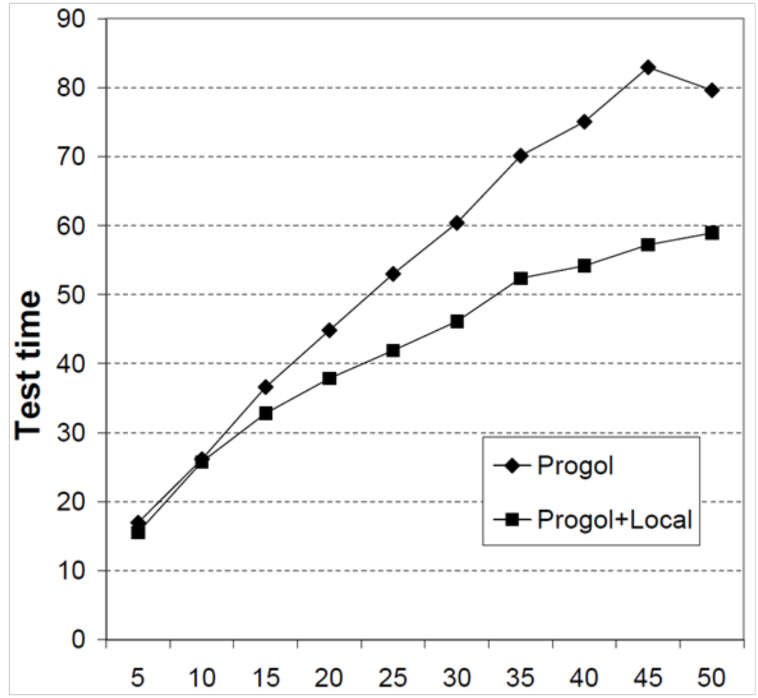

RMSE
1,1
1,05
1
0,95
0,9
0,85
0,8
0,75
0,7
5 10 15 20 25 30 35 40 45 50
Progol
Progol+Local

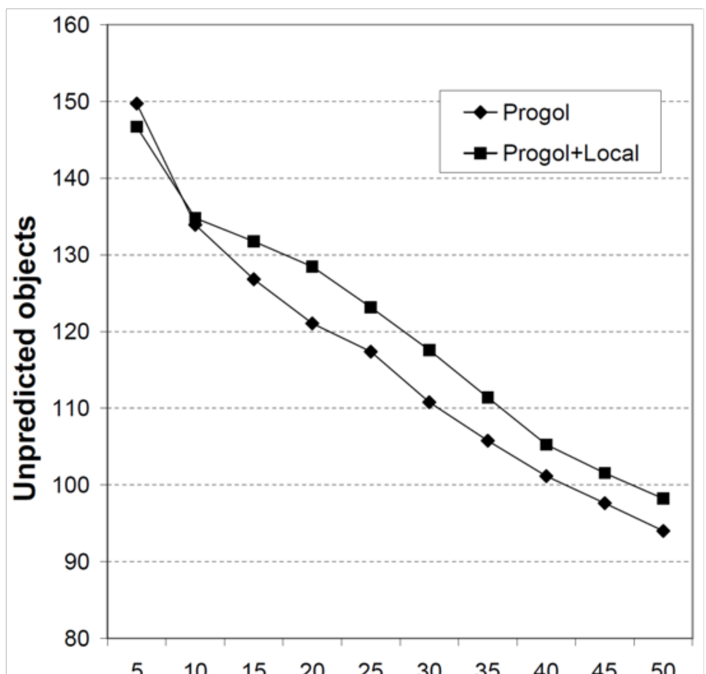

Mononicity violation
1,5
1,4
1,3
1,2
1,1
1
0,9
0,8
0,7
5 10 15 20 25 30 35 40 45 50
Progol
Progol+Local


Tau coefficient
0,25
0,2
0,15
0,1
0,05
0
5 10 15 20 25 30 35 40 45 50
Progol
Progol+Local


Correlation
0,45
0,4
0,35
0,3
0,25
0,2
0,15
0,1
0,05
0
5 10 15 20 25 30 35 40 45 50
Progol
Progol+Local


Build time
18000
16000
14000
12000
10000
8000
6000
4000
2000
0
5
10
15
20
25
30
35
40
45
50
Progol
Progol+Local


Test time
90
80
70
60
50
40
30
20
10
0
5
10
15
20
25
30
35
40
45
50
Progol
Progol+Local


Unpredicted objects
160
150
140
130
120
110
100
90
80
5
10
15
20
25
30
35
40
45
50
Progol
Progol+Local

Conclusion

- Description of a preference model
  - Local preferences form monotone space $[0,1]^N$
  - Global preferences aggregates preference degrees of attribute values into the overall rating of the object
- Description of learning the model
  - Various procedures for local and global preferences
- Experiments
  - Using similarity of preference models for collaborative filtering
  - Using local preferences as preprocessing for UTA and ILP