Program NOO
Measuring the approximation quality for sorting rules

Version 1.14

Günther Gediga*
Institut für Evaluation und Marktanalysen
Brinkstr. 19
49143 Jeggen, Germany
Gediga@Eval-Institut.de

Ivo Düntsch*
School of Information and Software Engineering
University of Ulster
Newtownabbey, BT 37 0QB, N.Ireland
I.Duentsch@ulst.ac.uk

June 22, 2001

Summary

The paper presents a description of the program NOO, which can be used for computing an approximation quality for sorting rules of the types "nominal $\rightarrow$ nominal" (NN), "nominal $\rightarrow$ ordinal" (NO), "ordinal $\rightarrow$ ordinal" (OO), and its generalisation to "(nominal, ordinal) $\rightarrow$ ordinal" rules (NOO). We provide a significance test for the overall approximation quality, and a test for partial influence of attributes based on the bootstrap technology.
A competing model - the approach of Greco et al. [3] – will be also computed.

Key words: Sorting rules, approximation quality, ordinal prediction

1 Introduction

In multi-criteria sorting problems we are often faced with statements of the form

> If someone is male and at least 30 years of age, then he will spend at least £10 a month on magazines.

Even though such a rule need not be globally true, it is worthwhile to approximate the prediction quality of the set of condition criteria taking into account all rules of the above form, and aggregate the values into a single measurement. As an example, consider the information system given in Table 1 on the following page. There, U is a set of objects, q is a condition criterion, and d a decision criterion. With respect to the orderings $\leq$, we can observe, among others, the following rules:

(1.1) $$(\forall x) f_q(x) \geq 3 \Rightarrow f_d(x) \geq 3,$$

(1.2) $$(\forall x) f_q(x) \geq 5 \Rightarrow f_d(x) \geq 5,$$

*Equal authorship is implied.

Table 1: A simple bivariate order information table

U	x_1	x_2	x_3	x_4	x_5	x_6
q	2	1	4	3	6	5
d	1	2	3	4	5	6

Here, $f_q(x)$, resp. $f_d(x)$ is the value of object x under attribute q, resp. d. The rules (1.1), (1.2) are called *ordinal-ordinal* (OO)-rules [2], because each side of the rule addresses an order relation.

The question arises, how we can measure the overall prediction success based on the instances of the observable rules in such a way that each rule contributes to the measure. This problem is, of course, not new, and solutions have been offered e.g. by [3] within the framework of rough sets. However, we shall see in Section **??** that the approximation quality given there is dissociated from the theory, and that a more refined measurement is needed.

We shall investigate not only pure sortings, but also "nominal" sortings, which results in rules such as

(1.3) $$(\forall x) f_{q_1}(x) = M \Rightarrow f_d(x) = 1000,$$

(1.4) $$(\forall x) f_{q_1}(x) = M \Rightarrow f_d(x) \geq 1000,$$

(1.5) $$(\forall x) f_{q_1}(x) = F \wedge f_{q_2}(x) \geq 3 \Rightarrow f_d(x) \geq 500,$$

Rule (1.3) is a *nominal - nominal* (NN) - rule, (1.4) is a *nominal-ordinal* (NO) - rule, and (1.5) is a mixed case (NOO) - rule.

The example of a published data set [1] shows the applicability of our method.

2 Program parameters

```
Program NOO(2.12):Computes (nominalxordinal)->ordinal-prediction (GG/ID,2001)

call noo <in> [-A -C -D.. -E.. -F[..] -L.. -N.. -Oout -P.. -Q -R.. -S. -U.. -X]

Options (lower or upper case):
 -A: gamma for ALL possible direction combinations within the attr. (def. no)
 -C: Perform CLASSICAL Rough Set Analysis (= assume every attr. to be nominal)
 -D: Decision attribute (default: last var.) -D=3 -> Attr. 3 is decision attribute
 -E: Exclude the variables in the list from analysis (default: no vars.)
     -E1,2,3,L will exclude 1st, 2nd, 3rd and last variable
 -F: Find reducts (def.: no). Without par it will find relative reducts.
     -F0.9 finds reducts for gamma=0.9
 -L: Number of lines per case in infile (default: 1)
 -N: Change attribute from ordinal to nominal values (def.: no nominal attr.)
 -O: A filename for the results (default: noo.out)
 -P: Bootstrap N for partial gamma testing (default: 0)
     -P0 disables testing; -P100 enables test with 100 trials
 -Q: Shows the Q product relation (def.: no)
 -R: Change directions of attribute coding (reversed coding).
     This option causes no effect for nominal attributes! (Ex.: -R1,2,3,L)
```

```
-S: Number of simulations for significance testing (default: 0)
    -S0 disables significance testing; -S1000 enables test with 1000 trials
-U: Use only the variables in the list for analysis (default: all vars.)
    -U1,2,3,L will use only the 1st, 2nd, 3rd and last variable
-X: eXtended output including rules and rule evaluation (def.: no)
```

$$\begin{aligned}\gamma_{NO}(= \mid \leq) &= \sum_{t \in V_d} w_t \cdot \gamma_{NO}(S|t) \\ &= \sum_{t \gneqq t_{\min}} \frac{|f_d^{-1}(T(t))|}{\sum_{t \gneqq t_{\min}} |f_d^{-1}(T(t))|} \cdot \frac{1}{|f_d^{-1}(T(t))|} \cdot \sum_{s \in det(t)} |f_Q^{-1}(s)| \\ &= \frac{1}{\sum_{t \gneqq t_{\min}} |f_d^{-1}(T(t))|} \cdot \sum_{t \gneqq t_{\min}} \sum_{s \in det(t)} |f_Q^{-1}(s)|.\end{aligned}$$

$$\gamma_{NO}(= \mid \leq,\, \geq) = \frac{\sum_{t \gneqq t_{min}} |\bigcup_{s \in det(t)} \varphi^{1'|\leq}(s,t)| + \sum_{t \lneqq t_{max}} |\bigcup_{s \in det(t)} \varphi^{1'|\geq}(s,t)|}{\sum_{t \gneqq t_{min}} |\varphi^{\leq}(t)| + \sum_{t \lneqq t_{max}} |\varphi^{\geq}(t)|} \tag{2.1}$$

$$= \frac{\sum_{t \gneqq t_{min}} |\bigcup_{s \in det^{\leq}(t)} f_Q^{-1}(s)| + \sum_{t \lneqq t_{max}} |\bigcup_{s \in det^{\geq}(t)} f_Q^{-1}(s)|}{\sum_{t \gneqq t_{min}} |f_d^{-1}(T(t))| + \sum_{t \gneqq t_{min}} |f_d^{-1}(T^{\smile}(t))|} \tag{2.2}$$

$$\gamma_{OO}(\leq \mid \leq) = \frac{\sum_{t \gneqq t_{\min}} |\bigcup_{s \in det(t)} f_Q^{-1}(S(s))|}{\sum_{t \gneqq t_{\min}} |f_d^{-1}(T(t))|}. \tag{2.3}$$

3 Example

The data are presented in Table 2.

The values of the decision attribute d (”Profit”(1) is better than ”Loss” (0)) should be predicted by the attributes:

q_1: capacity of the sales staff (”high” (2) is better than ”medium” (1) is better than ”low” (0))

q_2: perceived quality of goods (”good” (1) is better than ”medium” (0))

q_3: high traffic location (”yes” (1) is better than ”no” (0))

q_4: the geographical region (”A” (0) and ”B” (1) code different regions)

Call:

```
noo wareh.dat -n4 -e2,3 -s1000 -p1000 -x
```

Results:

Table 2: A NOO example (from GMS,98b, pp. 64)

Warehouse	q_1	q_2	q_3	q_4	d
1	High	Good	no	A	Profit
2	Medium	Good	no	A	Loss
3	Medium	Good	no	A	Profit
4	Low	Medium	no	A	Loss
5	Medium	Medium	yes	A	Loss
6	High	Medium	yes	A	Profit
7	Medium	Medium	no	A	Profit
8	High	Good	no	B	Profit
9	Medium	Good	no	B	Profit
10	Low	Medium	no	B	Loss
11	Medium	Medium	yes	B	Profit
12	High	Medium	yes	B	Profit

```
Infile ................... : wareh.dat
Resultfile ............... : noo.out
Number of lines per record : 1
Decision Attribute ....... : last
Exclude Attributes ....... : 2,3
Nominal scaled Attributes  : 4
Number of Simulations .... : 1000
Bootstrap N for partial test 1000
Extended Output

Number of ordinal attributes         : 1
Number of nominal attributes         : 1
Low category in d                    : 0
High category in d                   : 1
The q assignments to d
  2  0 (  0) (  2) ->   1
  1  0 (  1) (  4) ->   0
  1  0 (  1) (  4) ->   1
  0  0 (  2) (  1) ->   0
  1  0 (  1) (  4) ->   0
  2  0 (  0) (  2) ->   1
  1  0 (  1) (  4) ->   1
  2  1 (  3) (  2) ->   1
  1  1 (  4) (  2) ->   1
  0  1 (  5) (  1) ->   0
  1  1 (  4) (  2) ->   1
  2  1 (  3) (  2) ->   1

Number of Equivalence classes in Q: 6
Number of different d values         : 2
Informative pairs in <= Relation     : 4
Informative pairs in >= Relation     : 8
d>=1, when: c>=0 or c>=3 or c>=4  (6)
d<=0, when: c<=2 or c<=5        (2)
```

```
gamma   = 0.6667    g(leq)= 0.5000   g(geq)= 0.7500
      p = 0.0360      ERW = 0.2384   kappa = 0.5623

gamma(GMS)= 0.6667
      p = 0.0170      ERW = 0.1562   kappa = 0.6050

Analysis of partial gamma
Infl. of attr. 1: gammadiff=0.667 BIAS=0.102 s(boot)=0.263 z(boot)=2.534 low=0.250
Infl. of attr. 4: gammadiff=0.167 BIAS=0.008 s(boot)=0.137 z(boot)=1.219 low=0.000
```

References

[1] Cliff, N. (1994). Predicting ordinal relations. *British J. Math. Statist. Psych.*, **47**, 127–150.

[2] Düntsch, I. & Gediga, G. (1997). Relation restricted prediction analysis. In A. Sydow (Ed.), *Proc. 15th IMACS World Congress, Berlin*, vol. 4, 619–624, Berlin. Wissenschaft und Technik Verlag.

[3] Greco, S., Matarazzo, B. & Słowinski (1998). Rough set handling of ambiguity. In H.-J. Zimmermann (Ed.), *Proceedings of the 6th European Congress on intelligent techniques and soft computing*, 3–14, Aachen. Mainz Wissenschaftsverlag.