Towards the Personalization of Algorithms Evaluation in Data Mining

Gholamreza Nakhaeizadeh

Daimler-Benz AG,
Research and Technology 3
P. O. Box 23 60, D-89013 Ulm, Germany
nakhaeizadeh@dbag.ulm.DaimlerBenz.COM

Alexander Schnabl

Technical University Vienna, Department of
Econometrics, Operations Research and Systems Theory
Argentinierstr. 8/1, A-1040 Vienna, Austria
e9025473@fbma.tuwien.ac.at

Abstract

Like model selection in statistics, the choice of appropriate Data Mining Algorithms (DM-Algorithms) is a very important task in the process of Knowledge Discovery. Due to this fact it is necessary to have sophisticated metrics that can be used as comparators to evaluate alternative DM-algorithms. It has been shown in literature, that Data Envelopment Analysis (DEA) is an appropriate platform to develop multi-criteria evaluation metrics that can consider – in contrary to mono-criteria metrics – all positive and negative properties of DM-algorithms.
We discuss different extensions of DEA that enable consideration of qualitative properties of DM-algorithms and consideration of users preferences in development of evaluation metrics. The results open new discussions in the general debate on model selection in statistics and machine learning.

1. Introduction

Algorithm evaluation in Data Mining can be considered in conjunction with the general problem of model selection in statistics. Like model selection in statistics, the choice of appropriate Data Mining Algorithms (DM-Algorithms) is a very important task in the process of Knowledge Discovery in Databases (KDD). Due to this fact, it is necessary to have sophisticated metrics that can be used as comparators to evaluate alternative DM-algorithms. Specially, in two phases of the KDD-process, such metrics are necessary:

- In the phase of formulation, calibration and determining the level of detail of DM-Algorithms (models).
- To evaluate alternative DM-algorithms (models) based on different modelling techniques. e. g. neural networks, decision trees, etc.

To perform the evaluation tasks in both above-mentioned cases, a lot of metrics are suggested in literature as comparators between alternative algorithms. Most of these metrics are, however, based on only one property (often prediction accuracy rate) and can not consider all positive and negative properties of the DM-algorithms like computation time, complexity, understandability, novelty and usability of the discovered patterns (Fayyad et all. 1996). To overcome this shortcoming, Nakhaeizadeh and Schnabl (1997) suggest the application of Data Envelopment Analysis (DEA) which leads to multi-criteria metrics for evaluation of DM-algorithms (See also Jammernegg et al. 1998). The present paper extends the results of Nakhaeizadeh and Schnabl (1997) in two main aspects:

1. How can the qualitative properties of DM-algorithms (e.g. understandability) be considered, explicitly, to develop evaluation metrics and what is the effect of consideration of such properties on ranking of the DM-algorithms?
2. How can the user's preferences be considered in development of multicriteria-based evaluation metrics? In other words, how can the evaluation metrics be ***personalized?***

2. DEA-Concept and its Extensions

2.1 The main DEA-concept

Data Envelopment Analysis (DEA) has been developed by Charnes et al. (1978) for comparison of *Decision Making Units* (DMUs) using their efficiency. Efficiency can be used also to rank the DMUs. In our terminology, each DMU is a DM-algorithm and has positive and negative properties which are called *output* and *input components*, respectively. Generally, output components are those where higher values are better and input components are those where lower values are better. In our terminology, a typical output component is the accuracy rate produced by a supervised DM-algorithm. A typical input component is computation time that the DM-algorithm needs for training. Using these components, we can now define the efficiency of a DM-algorithm as follows:

$$\text{efficiency} = \frac{\sum \text{weighted output components}}{\sum \text{weighted input components}} \qquad (1)$$

Efficiency defined above can be used as an evaluation metric and considers all positive and negative properties of a DM-algorithm and thus it is a multi-criteria based metric. But this definition arises one major problem: how do the weights should be determined. Generally, it is not possible for the user to determine the weights objectively. For example, no user can say, what would be the benefit of 1% increasing of the accuracy rate in term of Dollar and Pence. DEA is an answer to this challenge. In the base DEA model the weights are chosen endogenous for each DM-algorithm individually by maximizing the efficiency. Model (1) can be transformed into a *Linear Program (LP)* that can be easily solved by the *Simplex Method*. The ranking method suggested by Andersen and Petersen (1993) can be used to rank the efficient algorithms (See Nakhaeizadeh and Schnabl (1997) for more detail).

2.2. Consideration of qualitative properties

The main DEA model described above can handle only the continuous components e.g. the accuracy rate. As mentioned before, to the properties of DM-algorithms belong, however, qualitative properties as well. Understandability, usability and novelty of the patterns covered by a DM-algorithm are examples for such properties. In this section, we describe extensions of DEA that can overcome this shortcoming and allow development of multi-criteria evaluation metrics based on all quantitative and qualitative properties of DM-algorithms. The practicability of these extensions will be examined later in section 3.

Before we describe the DEA extensions, it should be mentioned that in DEA, it is possible to consider only the properties that are at least ordinal. Consideration of nominal properties doesn't make sense. Fortunately almost all qualitative properties of DM-algorithms we are interested in like understandability, usability, etc. are ordinal. So that this shortcoming of DEA has no significant effect.

The first extension that we consider is based on the work Cook et al. (1996). In this extension, the ordinal properties are transformed by using binary representation to new components with the values 1 and 0. Such representation is actually usual in statistics and neural networks to handle nominal and ordinal attributes. For example if we have tree ordinal level for understandability of the results of the DM-algorithm k as high, middle and low, then we would have three new output components O_{k1}, O_{k2} and O_{k3} with following representation:

$O_{k1}=0$ $O_{k2}=0$ $O_{k3}=1$ for low understandability

$O_{k1}=0$ $O_{k2}=1$ $O_{k3}=1$ for middle understandability

$O_{k1}=1$ $O_{k2}=1$ $O_{k3}=1$ for high understandability

Regarding the fact that the third column has values all equal to one, it would be enough to use only O_{k1} and O_{k2}. After this transformation we have again a classical DEA described by model (1) including two additional output components that can be handled like before. We have used this modified version in our empirical results.

One shortcoming of the above approach is that if we have properties with a lot of values, then we need a lot of additional input and output components that lead to model augmentation and high dimension. An alternative to this approach is representing each ordinal property by using only one component. In the above example, we would have an additional component with ordinal values e.g. 0, 1 and 2. This alternative approach is used in our empirical study as well that will be described in section 3. This representation has, however, an arbitrary character and it is not robust, if one uses alternative values keeping the same order. For example one could use the order 1, 2 and 3 instead of 0, 1 and 2.

2.3. Personalization Aspects

As mentioned in the section one, the basic DEA model of Charnes et al. (1978) chooses the weights endogenous. It means that the unknown weights are determined by the model automatically. This approach is very appropriate specially for the cases in which the user has no a priori knowledge about the importance of different positive and negative properties of the algorithms. In practice, there are a lot of situations in which the user has a priori knowledge or preferences and he would like to consider such knowledge and preferences in evaluation of the DM-algorithms. For e.g. the user might prefer the accuracy rate to understandability of the results or vice versa. Now, the question is, how can one consider such ***personal desires*** in developing evaluation metrics for DM-algorithms? To perform this task, we suggest the following methods.

The first method is based on Allen (1997). The main idea is hereby to maximize the efficiency defined in model (1) under consideration of different restrictions that represent the preferences of users. Using the weights for output and input components, such restrictions can have different functional form like:

$$\lambda_y v_{ky} + \lambda_{y+1} v_{k,y+1} \le v_{k,y+2} \text{ or } \alpha_y \le \frac{v_{ky}}{v_{k,y+1}} \le \beta_y \qquad (2)$$

$$\gamma_y v_{ky} \ge u_{kx}, \qquad (3)$$

$$\delta_y \le v_{ky} \le \tau_y \quad \text{or} \quad \rho_x \le u_{kx} \le \eta_x \qquad (4)$$

Relation (2) represents two preferences on the output components of the algorithm k, u_{kx} and v_{ky} are the weights of the x-th input and y-th output of the algorithm k, resp. Relation (3) is an example representing the relation between output and input components and (4) represents user preferences as bounded intervals for output or input. Parameters $\alpha, \beta, \gamma, \ldots\ldots$ are determined by the user according to their a priori knowledge or preferences. It is possible also in some cases to estimate these parameters using various approaches. Detail can be found in Roll et al. (1991), Dyson and Thanassoulis (1988) and Roll and

Golany (1993). Put together, in this approach, the efficiency of each DM-algorithm is determined by model (1) and additional restrictions representing user preferences.

The second approach we have considered is our own suggestion in which the user preferences are represented by linear equalities showing the relation between outputs or inputs as linear functions like:

$$\kappa_y u_{ky} + \kappa_{y+1} u_{k,y+1} = u_{k,y+2} \qquad (5)$$

Such restrictions lead to reduction of the input (out) dimension. For example if we solve equation (5) by u_{ky} and put the result in the definition of the efficiency given in (1) then instead of p dimension we will have p-1 It means that consideration of restriction like (5) leads to solving an optimization problem with a lower dimension.

The above approaches are sensitive to the scaling of the input and output components. This problem can be solved, however, by normalizing. In our empirical study described in section 3, we have normalized the input and output components by the dividing them to the corresponding maximum value. Other normalization approaches are possible as well.

3. Empirical results

3.1. Impact of additional qualitative criteria on evaluation of DM-algorithms

In the last section, we have seen that DEA can handle ordinal qualitative properties like understandability, usability etc. as well. The first point we would like to analyze in this section is the effect of consideration of such the ordinal qualitative properties of DM-algorithms on their evaluation. The base for our empirical study is again the project StatLog dealing with evaluation of 23 supervised classification algorithms using 22 databases reported in Michie, Spiegelhalter and Taylor (MST) (1994).

MST report totally five measured properties for each evaluated algorithm namely accuracy rates for testing and training data, needed storage and computation time for training and testing data. In some cases they report instead of the accuracy rate the average cost of misclassification. They don't measure the qualitative properties like understandability etc. Thus to perform our study, we need measuring of such additional properties.

It is clear that it is not possible for us to determine e.g. the level of the understandability of the results of 23 algorithms applied to 22 domains. It would be a task for the end users of such results. To find a solution we used the *explanation power* of the DM-algorithms as a *proxy* for the understandability of their results and divided the algorithms reported in MST into following three groups:

- Group one includes all Machine Learning algorithms evaluated in StatLog. To this group belong CART IndCART, NewID, AC2, Baytree, CN2, C4.5, Itrule and Cal5. We think that it is a general agreement that the explanation power of the machine learning algorithm is high. Thus we have assigned to these algorithms the highest explanation power degree, namely 2
- Group two includes all statistical algorithms like Discrim, Quadisc, Logdisc, SMART, ALLOC80, k-NN, CASTLE and Naivebayes. We assigned to these algorithms a middle explanation power degree, 1
- Group three includes all Neural Networks evaluated in StatLog namely Kohonen, DIPOL92, Backprop, RBF, LVQ and Cascade. To this group, we have assigned the lowest explanation power degree, 0

Explanation power of various DM-algorithms defined above, serve in our study now as values of an additional output component, namely, the *understandability* of the results. The other input and output components remain the same as they are reported in Nakhaeizadeh and Schnabl (1997).

To evaluate the DM-algorithms, we have used two approaches described in section 2.2. The first approach is due to Cook et al. (1996) that was modified by us for two binary output components. The second is our own suggested approach. It is not possible to report here the results for all datasets. As examples, we discuss the evaluation results of the algorithms for the datasets Satellite Images and Diabetes. The ranks of the algorithms are reported in the forth and fifth columns (Cook and NS1) of Tables 1 and 2, respectively. The second column of these tables report the original mono-criteria ranking of MST based only on accuracy rate or the cost of misclassification. The third column (Jam) comprises the results reported in Jammernegg et al. (1998) achieved without including the *understandability* of the algorithms as an additional output component. The last two columns in the tables report the results dealing with the user preferences. We will refer to these results later in this section.

To remember again, the results of the columns "Cook" and "NS1" in Tables 1 and 2 are achieved under consideration of understandability and the results of the column "Jam" without it. We can see that for most of the algorithms, two alternative approaches that we have used for consideration of understandability have no significant influence on ranking results. For statistical algorithms (e.g. Quadisc, Logdisc) are, however, the ranks different. Another interesting result is that consideration of understandability as an additional positive property leads for all neural networks in no cases to a better rank. This result is consistent with the fact that we assign to neural networks the lowest degree of understandability. The new ranks are worth in most cases and in a few cases remain the same. Ranks decreasing is more significant, if we compare the new results with MST-results in which the evaluation of

DM-algorithms is performed by using only one positive criterion namely the accuracy rate. This conclusion is not valid only for the three datasets reported here, but for all others. But, there is no such stable and homogenous conclusion for other algorithms (statistical and machine learning) throughout our study.

3.2. Impact of user preferences on evaluation of DM-algorithms

The main contribution of this paper is, however, to study the effect of consideration of user preferences on the evaluation of the DM-algorithms. We have discussed the relevant theoretical issues in section 2.3. To examine the practicability of the extensions of DEA to this issue, we have defined following preferences. The first category of the user preferences is examined by using our own suggested model based on dimension reduction and includes three preferences:

- Testing accuracy is 100 times more important than the training accuracy
- Explanation power is 50 times more important than the training accuracy
- Testing time is 100 times more important than the training time

The corresponding results for this approach are reported in Tables 1 and 2 in the column "NS2". The second category of the user preferences includes three preferences as well. They are described below and are examined by using the approach due to Allen et al. (1997) reported in section 2.3 :

- Testing accuracy is at least two times more important than explanation power
- Explanation power is at least two times more important than training accuracy
- Testing time is at least two times more important than training time

The corresponding results for this approach are reported in Tables 1 and 2 in the last column "Allen".

The first general conclusion that we can get from the results of this section is that considering the preferences of users defined above, change significantly the evaluation of DM-algorithms. This is an overall valid result and is independent from the approach that we have used to consider the user preferences.

The results show also that putting a high importance degree on the understandability has left an significant on evaluation of neural networks. We can see this effect, if we compare the results of the column NS1 with those of NS2. The results of NS2 are achieved by considering the preference that "explanation power is 50 times more important than the training accuracy" and ". This has caused overall decreasing of the rank of neural network. These results are again consistent with the fact that we have given the lowest degree to the understandability of the results of neural networks. An interesting exception is the algorithm DIPOL92 applied to Diabetes dataset (Table 2). In this case though the above mentioned preference, we can see that its rank increases. The reason might be very high accuracy rate of this algorithms for Diabetes dataset

Algorithms	MST	Jam	Cook	NS1	NS2	Allen
Discrim	19	11	14	12	14	15
Quadisc	14	8	9	9	11	3
Logdisc	17	19	17	19	13	13
SMART	16	17	16	17	12	7
ALLOC80	5	15	10	15	10	6
k-NN	1	1	2	2	9	4
CASTLE	21	*	*	*	*	*
CART	6	7	7	7	1	1
IndCART	6	6	8	8	2	12
NewID	10	14	1	1	8	2
AC^2	15	*	*	*	*	*
Baytree	9	10	13	11	4	16
NaiveBay	22	12	12	13	5	17
CN2	10	5	6	6	7	11
C4.5	10	9	11	10	3	19
Itrule	FD	FD	FD	FD	FD	FD
Cal5	13	13	15	14	6	18
Kohonen	20	*	*	*	*	*
DIPOL92	3	4	5	5	16	9
Backprop	8	18	19	18	18	10
RBF	4	3	4	4	17	5
LVQ	2	2	3	3	15	8
Cascade	17	16	18	16	19	14

Table 1: Algorithms ranking for Satellite Image dataset for MST and different multi-criteria metrics. FD: Algorithm failed on this dataset. "*" is used for missing values (or not applicable).

Which has led to second best algorithm in MST ranking. On the other hand, we can see that the empirical results achieved by consideration of the user preference that "explanation power is at least two times more important than training accuracy" and " testing accuracy is at least two times more important than explanation power" are totally else. Apparently this time, the neural networks could compensate their low explanation degree in many cases by better results of other criteria.

In the both groups of above defined preferences the testing time is weighted higher than training time. We can see the effect of such preferences very good for algorithms with relatively high training and low testing time. Statistical algorithm SMART e.g. has the 17th rank (Table 1, column NS1). In NS1 the weights are determined automatically by using the same a priori importance degree for all evaluation components. By consideration the user preference giving more importance degree to accuracy rate, we can see from Table 1 (column NS2 and Allen) that it gets the better ranks 12 and 7, respectively. This is due the fact that Satellite Image dataset, SMART needs 27376.2 seconds for training that is relatively high. Testing time is 10.8 seconds (MST,

1994 p. 145). We can see that its MST ranking 16 which is achieved by using only the accuracy rate is improved by increasing the importance of testing time.

Algorithms	MST	Jam	Cook	NS1	NS2	Allen
Discrim	3	5	4	6	13	9
Quadisc	11	17	16	10	15	10
Logdisc	1	4	6	7	12	7
SMART	4	*	*	*	*	*
ALLOC80	21	*	*	*	*	*
k-NN	22	6	8	8	17	19
CASTLE	10	19	14	18	14	2
CART	9	16	5	4	7	4
IndCART	14	20	18	15	8	12
NewID	19	2	2	2	10	1
AC^2	18	3	7	5	9	8
Baytree	14	8	11	11	4	15
NaiveBay	11	18	17	20	16	11
CN2	19	12	15	14	11	5
C4.5	13	11	13	13	2	16
Itrule	6	13	1	1	3	20
Cal5	8	10	10	10	1	18
Kohonen	17	15	20	17	20	13
DIPOL92	2	9	12	12	6	17
Backprop	7	1	3	3	18	6
RBF	5	7	9	9	5	14
LVQ	16	14	19	16	19	3
Cascade	*	*	*	*	*	*

Table 2: Algorithms ranking for Diabetes dataset for MST and different multi-criteria metrics.

4. Conclusions

Different aspects of model selection have been studied in the statistical literature. But no comprehensive attention is paid to development of the methods which can explicitly consider the user preferences in model selection and algorithm evaluation. This statement is valid also for KDD-Community. On one hand we need multi-criteria metrics for evaluation of the DM-algorithms which are based on all positive and negative properties of DM-algorithms. Using only mono-criteria metric e.g. accuracy rate would lead to no fair evaluation of DM-algorithms and consequently to no appropriate model selection. Nakhaeizadeh and Schnabl (1997) and Jammernegg et al. (1998) have been the first attempts contributing to this debate. On the other hand, we need metrics that are multi-criteria-based and can explicitly consider the user preferences in evaluation of DM-algorithm. The present paper is an attempt to develop such metrics. The main contribution of this paper is that it suggests quantitative methods which can handle the a priori preferences of users and considering them in an explicitly way in the process of evaluation of DM-algorithms. We think that such approaches bring new idea and open new perspectives to the general debate on model evaluation and selection.

References

Allen, R; Athanassopoulos, A.; Dyson, R. G. and Thanassoulis, E. (1997). Weights restrictions and value judgements in Data Envelopment Analysis: Evolution, development and future directions, Annals of Operations Research 73, pp. 13-34

Andersen, P. and Petersen, N. C. (1993) A Procedure for Ranking Efficient Units in Data Envelopment Analysis, Management Science, Vol. 39, No. 10, pp. 1261-1264

Charnes, A.; Cooper, W. and Rhodes, E. (1978) Measuring the efficiency of decision making units, European Journal of Operational Research 2, pp. 429-444

Cook, W.D.; Kress, M. and Seiford, L.M. (1996) Date Envelopment Analysis in the Presence of Both Quantitative and Qualitative Factors, Journal of Operational Research Society 47, pp. 945-953

Dyson, R.G. and Thanassoulis, E. (1988) Reducing weight flexibility in DEA, Journal of the Operations Research Society 39, pp. 563-576

Fayyad, U.M.; Piatesky-Shapiro, G. and Smyth, P. (1996) From data mining to knowledge discovery: An overview, in: Fayyad, U.M.; Piatesky-Shapiro, G.; Smyth, P. and Uthurusamy, R.: Advances in Knowledge Discovery and Data Mining, AAAI/MIT Press, pp. 1-30

Jammernegg, W; Luptàcik, M; Nakhaeizadeh, G. and Schnabl, A. (1998). Ist ein fairer Vergleich von Data Mining Algorithmen möglich ? In Nakhaeizadeh, G. (Ed.) Data Mining . Theoretische Aspekte und Anwendungen, 225-247, Physica Verlag, Heidelberg.

Michie, D.; Spiegelhalter, D.J. and Taylor, C.C. (1994) eds., Machine Learning, Neural and Statistical Classification, Ellis Horwood, Chicester

Nakhaeizadeh, G. and Schnabl, A. (1997) Development of Multi-Criteria Metrics for Evaluation of Data Mining Algorithms, Third International Conference on Knowledge Discovery and Data Mining, Proceedings, Newport Beach, California, August 14-17, pp. 37-42

Roll, Y.; Cook, W.D and Golany, B. (1991) Controlling factor weights in DEA, IIE Transactions 23, pp. 2-9

Roll, Y. and Golany, B. (1993) Alternative methods of treating factor weights in DEA, Omega, International Journal of Management Science 21, pp. 99-109