Expert Systems with Applications 39 (2012) 9435–9442

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

DEA based data preprocessing for maximum decisional efficiency linear case valuation models

Parag C. Pendharkar [a,*], Marvin D. Troutt [b,1]

[a] School of Business Adminsitration, Penn State Harrisburg, 777 West Harrisburg Pike, Middletown, PA 17057, USA
[b] Kent State University, College of Business Administration, Kent State University, Kent, OH 44242, USA

ARTICLE INFO

Keywords:
Data envelopment analysis
Interactive classification
Linear programming
Data mining
Decisional efficiency

ABSTRACT

In this paper, we use data envelopment analysis (DEA) to preprocess training data cases before the maximum decisional efficiency (MDE) principle is used to estimate discriminant function parameters. Using an example from the literature and simulated datasets, we compare the performance of DEA-MDE procedure for parameter estimation with traditional MDE procedure without data preprocessing. The results of our experiments indicate that the DEA-MDE procedure eliminates some inconsistencies caused by MDE principle, provides results that are consistent with an ensemble of expert decisions, reduces dimensionality of examples used in training datasets, and performs equal to or better than the MDE procedure for holdout sample tests. The DEA-MDE procedure appears to be sensitive to class data distribution and best results are obtained when a class data distribution is exponential.

1. Introduction

Data envelopment analysis (DEA) is a technique developed to measure efficiency of decision-making units (DMUs) in a variety of settings. Since its introduction, the technique has been used for several manufacturing, banking, health care and service industries. Recently, DEA has been used for data-mining applications. Among the applications of DEA in data mining are the uses of DEA for data preprocessing in forecasting applications (Pendharkar, 2005; Pendharkar & Rodger, 2003), outlier detection (Banker & Chang, 2006), classification (Pendharkar, 2011; Seiford & Zhu, 1998; Troutt, Rai, & Zhang, 1996), cluster analysis (Po, Guh, & Yang, 2009) and inverse classification problems (Pendharkar, 2002). We do not know of any studies that have used DEA for data preprocessing for classification applications. Pendharkar (2005) used the DEA based data preprocessing for forecasting applications where predicted variable was continuous. In classification applications, predicted variable is binary and the application of DEA for data preprocessing requires a different approach. An application of DEA for data preprocessing in classification applications would be desirable for at least two reasons. First, data preprocessing would result in fewer records and less computational effort. Second, data preprocessing would result in elimination of trivial classification examples and outliers resulting in a classification function that may have better generalizability due to lower training data over fitting[2].

There is substantial literature on data preprocessing (Kone & Karwan, 2011) for classification problems (Chen, Hsu, & Chang, 2010; Wang & Shi, 2008). Some of the reasons for data preprocessing are to improve scalability (Wang & Shi, 2008), reduce bias originating from class imbalance (Chen, Hsu, & Chang, 2010), and improve generalizability (Pendharkar, 2005). Most DEA data mining applications consider single output multiple input settings (Pendharkar, 2011; Seiford & Zhu, 1998; Troutt et al., 1996). Banker (1993) provides a statistical foundation for DEA under the single output multiple input setting where DEA estimators are shown to be maximum likelihood estimators (MLE) of non-parametric probability density functions. Banker (1993) argues that the primary difference between statistical MLE and DEA estimators is the assumption that the production frontier in DEA is non-parametric monotone increasing and concave function. When all the assumptions of single output, multiple inputs and concave monotone increasing production function are satisfied, Banker (1993) showed that DEA estimators maximize the likelihood for a broad class of density functions including exponential and half-normal distributions.

* Corresponding author. Tel.: +1 717 948 6028; fax: +1 717 948 6456.
E-mail addresses: pxp19@psu.edu (P.C. Pendharkar), mtroutt@kent.edu (M.D. Troutt).
[1] Tel.: +1 330 672 2750x335; fax: +1 330 672 2953.

[2] Training data over fitting is a concern in machine learning literature (see Bhattacharyya and Pendharkar (1998)). Training data over fitting implies learning "noise" in the dataset. Noisy examples are examples that may be considered as outliers and learning their characteristics may adversely impact generalizability of learned classification rules.

0957-4174/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2012.02.118

When DEA is used for classification, DMUs with efficiency (ξ) score of 1 are considered to lie on the classification boundary or envelopment that separates one class from another (Pendharkar, 2011). If we define one-sided deviation term $v = (1 - \xi)$ then maximizing likelihood of the probability density function for v, $\alpha(v)$, is equivalent to minimizing the sum of deviations if $\alpha(v)$ is exponential or equivalent to minimizing the sum of squared deviations if $\alpha(v)$ is half-normal (Banker, 1993). Thus, computing efficiencies of DMUs and removing the DMUs with low efficiency scores is equivalent to removing DMUs that lie in the tail of $\alpha(v)$. As low efficiency DMUs are removed from the original dataset, the MLE estimate for the remaining dataset with fewer DMUs will be lower than the original dataset. The lowering of MLE estimate may achieves better generalization due to removal of outliers/trivial classification cases, however, care must be exercised to not eliminate too many DMUs where the new model may lose generalizability compared to the model built from the original dataset.

Troutt (1995) proposed a related decisional efficiency based procedure for parameter estimation of certain optimization models. Troutt (1995) showed that these parameter estimation models can be formulated using the maximum decisional efficiency (MDE) principle. The MDE model was shown to be a MLE for certain class of monotone increasing density functions. The primary difference between MDE and DEA is that in former case a production function is specified in finite number of parameters, whereas in case of DEA the number of parameters to be estimated increases with the sample size (Banker, 1993). For finite sample sizes, DEA estimators would be biased and provide MLE estimates below the theoretical frontier [2] suggested by MLE model. Given that Troutt (1995) proved that the MDE model maximizes likelihood of certain monotone increasing density functions, it can be assumed that DEA estimators would provide MLE estimates below the theoretical frontier suggested by the MDE model.

To illustrate the utility of DEA for data preprocessing for MDE linear case valuation models, we consider two different scenarios. In the first scenario, we assume a classification problem where classification data is generated by several decision-makers, which in the context of DEA may be considered as different decision-making processes. Given different decision-making processes, DEA is applied independently to screen examples that are fed into MDE data aggregation and parameter estimation process (Troutt, 1995; Troutt, Rai, & Tadisina, 1997; Troutt, Zhang, Tadisina, & Rai, 1997). The MDE principle aggregates data from different decision-making processes or decision-makers and generates a linear case valuation model that can be used for classification (Troutt, Rai, et al., 1997; Troutt, Zhang, et al., 1997). When the DEA is used to preprocess data for the MDE model, we call our procedure DEA-MDE, otherwise the procedure is called MDE. In the second scenario, we consider classic classification problem where training data comes from one source and represents only one decision-making process. In both scenarios, we compare DEA-MDE results with the MDE results.

The rest of the paper is organized as follows. In Section 2, we provide an overview of the DEA based preprocessing and the MDE principle based data aggregation for linear case valuation. In Section 3, we illustrate the application of DEA-MDE procedure using a pre-reported example of multiple decision-maker data; and compare it with the MDE procedure without data preprocessing and a multiple decision-maker ensemble of what we call overlapping cases. In Section 4, using simulated datasets, we compare the application of DEA-MDE and the MDE procedures for our second scenario. In Section 5, we conclude the paper with a summary and provide a few directions for future work.

2. DEA procedure for data preprocessing and the MDE principle

For over a decade, DEA models have been used for classification (Troutt et al., 1996; Pendharkar, 2002; Pendharkar, 2011; Seiford & Zhu, 1998). Pendharkar (2011) used input-oriented variable returns to scale models for developing classification frontiers for binary classification problems. To describe Pendharkar (2011) input-oriented models, we assume that the data matrix $D = \{<\boldsymbol{x_1}, c_1>, \ldots, <\boldsymbol{x_n}, c_n>\}$ for a classification problem with two classes accept (A) and reject (R) is available. The individual elements of the matrix D are represented as x_{ij}, where $i \in \{1,..,n\}$ represents the row index and $j \in \{1,..,m\}$ represents the column index. The last column in the data class matrix D represents the class label for the vector in a row, which is denoted by c_i, where c_i was the class assigned by a decision-maker. We make an assumption of conditional monotonicity, where higher values of decision-making attributes (x_{ij}) indicate higher probability of accept class classification decision. If we partition the dataset D into D^A and D^R, where $D^A = \{\boldsymbol{x}_i \mid c_i = A, \ \forall i \in \{1,\ldots,n\}\}$ and $D^R = D - D^A$. The accept class frontier is given by solving following set of linear programs for each vector $\boldsymbol{x}_i \in D^A$.

$$\text{Minimize } \xi^i, \ i = 1, \ldots, n, \text{ and } i \in D^A \tag{1}$$

subject to:

$$\sum_{i=1}^{n} \lambda_i x_{ij} - \xi^i x_{ij} \leqslant 0, \ j = 1, \ldots, m \tag{2}$$

$$\sum_{i=1}^{n} \lambda_i = 1 \tag{3}$$

$$\lambda_i \geqslant 0 \ \forall i = 1, \ldots, n \text{ and } i \in D^A \tag{4}$$

If for some $i = \{1,\ldots,n\}$, solution of (1)–(4) yields a value of $\xi^{i*} = 1$ then that case was considered to lie on the DEA accept class frontier, otherwise ($\xi^{i*} < 1$) it was considered to be above the accept class frontier indicating higher degree of acceptance under conditional-monotonicity assumption. Similarly, a reject class frontier is given by solving the following set of linear programs by considering each vector $\boldsymbol{x}_i \in D^R$.

$$\text{Minimize} \left(\xi^i = \frac{1}{\omega^i}\right), \ i = 1, \ldots, n, \ \textit{and } i \in D^R \tag{5}$$

subject to:

$$\sum_{i=1}^{n} \lambda_i x_{ij} - \omega^i x_{ij} \geqslant 0, \ j = 1, \ldots, m \tag{6}$$

$$\sum_{i=1}^{n} \lambda_i = 1 \tag{7}$$

$$\lambda_i \geqslant 0 \ \forall i = \{1, \ldots, n\}; \ \omega^i \text{unrestricted } \textit{and } i \in D^R. \tag{8}$$

For any i where the value of $\xi^{i*} = 1$, the case was considered to lie on the reject class frontier. Otherwise, it was considered to lie under the reject class frontier with value of $\xi^{i*} < 1$.

Figs. 1 and 2 illustrate typical accept class and reject class frontiers, where $m = 2$. The frontiers are defined by points taking values of $\xi^{i*} = 1$ (denoted by stars). The inefficient examples for accept class A lie above the accept class frontier, and inefficient examples for reject class fall below the reject class frontier. When ξ^{i*} values are less than 0.5, it means that these cases are very easy to classify into either accept class or reject class because these cases will typically lie farther from the classification decision boundary.