The Pareto Principle in Datamining: an Above-Average Fencing Algorithm

K. Macek

This paper formulates a new datamining problem: which subset of input space has the relatively highest output where the minimal size of this subset is given. This can be useful where usual datamining methods fail because of error distribution asymmetry. The paper provides a novel algorithm for this datamining problem, and compares it with clustering of above-average individuals.

Keywords: ART, Pareto principle, insurance risk.

1 Introduction

In some cases, usual methods of supervised learning are not able to provide satisfactory results. This may occur in data with asymmetric distributed error, which is typical in insurance. In order to manage the asymmetry in the sense of the law of large numbers, this paper offers a new algorithm, which constructs a predictor not for points, but for sets. We will show an algorithm for finding sets of units with above-average outputs.

Let X be a set, and let μ, ν be measures over it. The Pareto principle arises if there is a set \( P \subseteq X \) where

\[
p(P, X) = \frac{\nu(P)}{\mu(P)} \mu(X) = 1
\]

and \( r(P) = \frac{\nu(P)}{\mu(X)} \geq 0 \). Let μ stand for volume, ν for productivity, p for proportion. Typically, the Pareto principle is considered as a rule that 20% of elements “produces” 80% of output. This principle was discovered by Vilfredo Pareto while assessing the welfare distribution in the UK at the end of the 19th century. His ideas were systematically described, applied and extended by Max Lorenz[7]. In this case, \( r(P) = 0.2 \) and \( p(P) \geq 4 \). Managerial science often works with the Pareto diagram [1]. X is discrete \( \{x_1, x_2, \ldots, x_n\} \). The elements are ordered by their production \( \nu(x_i) \), and the production is drawn in a chart. In statistics, the Pareto principle is represented by the continuous Pareto distribution:

\[
Pr(X > x) = \left( \frac{x}{x_m} \right)^{-k}
\]

for all \( x \geq x_m \), where \( x_m \) is the (necessarily positive) minimum possible value of X, and \( k \) is a positive parameter. Pareto distribution has positive skewness

\[
\frac{2(1 + k)}{k - 3} \sqrt{\frac{k + 2}{k}},
\]

which means that the below-average subset is bigger than the above-average complement. This occurs in many real life situations: the median citizen has a below-average salary, the median driver causes below-average claims, etc.

Let \( \xi : X \rightarrow \mathbb{R}^n \) be attributes of X. (Attributes can be considered as columns in a data table, i.e. the n-tuple from the i-th row. Mapping \( \xi \) can also involve preprocessing. If \( X \subseteq \mathbb{R}^k \) for some \( k \in \mathbb{N} \), the \( \xi \) mapping may also be an identity.) The problem of prediction consists in constructing the mapping \( \hat{y} \) so that

\[
\int_{\mathbb{R}^k} \| \hat{y}(\xi) - y \| d\mu \to 0.
\]

However, the construction of mapping \( \hat{y} \) may be difficult if the Pareto principle arises. A small subset of high productivity (called outliers) corrupts usual assumptions. Usual datamining techniques propose removing the set and working only with the rest. However, in case of the Pareto principle, the small set is very interesting. It is not adequate to speak about outliers, because such data is relevant and obvious. Therefore, we are dealing with more humble result, i.e. with finding the set \( P \) defined in (1).

The formulated problem, i.e. finding

\[
\arg \max_{P \subseteq X} \ p(X, P) \quad \text{under condition} \quad r(P) > r_0
\]

is new, and has not been found in the current literature. However, many other topics are related to it. First, clustering methods [12] can be employed. Creating clusters of above-average individuals, the set \( P \) will be defined as these clusters. It is important to define the border of the clusters somehow. If these clusters are well found, they can be employed for a more precise approximation [6]. Another approach is to attempt to find a prediction mapping \( \hat{y} \) where the set \( P \) is afterwards defined at some level of this mapping. RBF neural networks [5] provide an example where the approach of rough sets is employed. Finally, effective \( P \) can also be detected also Data Envelopment Analysis [2]. However, none of these methods – as the research in the bibliography shows – has been applied explicitly to the problem of above-average subsets.

2 The Fencing Algorithm

The following algorithm is the first attempt to solve this problem (3). It offers the construction of \( \hat{P} \subseteq X \) with above-average production, i.e. with high \( \hat{p} \). The space \( \mathbb{R}^n \) will be
considered as $X$. The set $P$ is represented by union intervals (from points to hyperboxes) represented by means of complement coding. (Complement coding is a concept applied in ART and ARTMAP neural networks, e.g. [3]. However, the objective of the Fencing algorithm is different. While ART and ARTMAP work iteratively, the Fencing algorithm must often go through the entire training set.) This algorithm works only with finite data sets $D$, namely with $(x_i, y_i)$ pairs. Therefore, for all subsets of $D$, the volume $\mu$ is defined as count and the production as the sum of the production of particular items $\nu(M) = \sum_{i \in \xi^{-1}(M) \cap D} y_i$.

Other ways to construct such intervals may be considered. The following algorithm uses fencing. Fencing is a heuristic approach which anticipates that areas of higher average production are located between mutually close points with high production. The algorithm attempts to build a rectangular fence around the area of above-average production, as shown in Fig. 1.

### 2.1 Measuring and data preprocessing

$\xi$ mapping is necessary. This mapping involves measuring and data preprocessing. The simplest way is to transform binary attributes into real attributes by 0–1 coding. Categorical attributes are transformed into more binary attributes. It is very useful to reduce the input vector dimension, e.g. by Principal Components Analysis [11]. Let us define $x_i^\xi = \xi(i)$ and the vector $x_{\max} = (x_1^{\max}, x_2^{\max}, \ldots, x_n^{\max})$ so $x_i^{\max} > x_j^\xi \forall i \in D$. The vector $x_{\max}$ is used for complementary coding $x$.

### 2.2 Data splitting

The data set $D$ is divided randomly into three subsets: base subset $B$, training subset $T$ and validation subset $V$. Sets $B$ and $T$ are used for constructing the predictor $y$, whereby their size will be represented by $|B| \ll |T|$, say $10 \cdot |B| = |T|$. The size of $V$ is chosen with respect to cross validation [8].

#### 2.3 Starting set of intervals

The starting set of intervals is defined as follows $R_O = \xi(x_i) | i \in B_O$, whereby $B_O = \left\{ ni \in B | y_i > k \cdot \mu(B) \right\}$, $\xi$ is the complement coding $\xi : \mathbb{R}^n \rightarrow \mathbb{R}^{2n}$ and $k$ is a parameter. The definition of $B_O$ ensures that $p(B_O, B) > k$.

#### 2.4 Interval expansion

Two intervals $r_1$, $r_2$ can be expanded as follows $r_{1,2}^{\text{new}} = \min(r_1, r_2)$, $i = 1, 2, \ldots, 2n$. The construction of $\hat{P}$ consists in iterative expansion of intervals. The process starts with $R_O$. Two intervals are expanded only if the new interval covers $p(I(r^{\text{new}}), T) > q_i$, where $q_i$ is a parameter that sinks linearly during the process from $p_0$ to $p_1$. In order to expand the compared intervals, a heuristic is used. Two intervals $I(r^{a})$, $I(r^{b})$ are suitable for expansion if $p(I(r^{a}))$ and $p(I(r^{b}))$ are high, and if $r^a$ and $r^b$ are close. Let us define suitability as:

$$v_{a,b} = \frac{p(I(r^{a}), T) \cdot p(I(r^{b}), T)}{\delta(r^{a}, r^{b})},$$

where $\delta$ is a metrics. The first version of the algorithm worked with Hamming distance [6], but other metrics can be also applied. In each step, a pair of intervals is tested for expansion. The pair is selected partly randomly as follows: the pair with highest suitability (probability 0.8), the pair with lowest suitability (0.1), or a random pair (0.1). If a pair is expanded, its suitability is recalculated for all other intervals.

![Fig. 1: Fencing: Expanding the square, a new fence (dashed) is recommended to the rectangle that is close and has high productivity $\rho$](image)

2.5 Termination

The algorithm terminates after all pairs of intervals have been tested and none can be expanded. Afterwards, unexpanded intervals (i.e. points) are deleted. Because intervals may overlap, their conjunction may have lower \( p \) than average \( p \) of all intervals. Therefore, only intervals with highest \( p \) are considered as results so their conjunction has \( p \) high enough (e.g. higher than a given threshold).

2.6 Validation

Finally, the results are validated with respect to the validation set \( r(P \cup V, V) \), and \( p(P \cup V, V) \) are calculated. Such values can be considered as the quality of the algorithm.

3 Results

The Fencing algorithm has been applied successfully on data on 18 177 insurance claims related to traffic accidents in the Czech Republic in 2003–2005. Categorical and numerical attributes were transformed into binary attributes. There was a total of 135 binary attributes. The considered attributes and their transformation is summarized in Table 1. The Fencing algorithm has been implemented in Matlab as a set of simple scripts. It should be mentioned that this particular data set inspired the author to invent the Fencing Algorithm, after attempts to build some regression model failed. Generalized Linear Models [4], which are typical in insurance mathematics, and multilayer perceptrons [8] did not provide sufficient results, as shown in Table 2. The data set was split into 10 subsets and one subset was always tested. The logarithm of total costs was taken as an output variable. However, the mean absolute error remains very high (the prediction and reality differ over twentyfold on average!).

First experiments showed that the 135 dimensional space is too sparse and there are many further unexpandable intervals. Therefore, the dimension was reduced by selecting 36 attributes describing the region of the claimant, road type, and cause of the accident. After next unsuccessful experiments with \( p = 4 \) and \( p = 2 \), it was necessary to set \( p = 15 \). Then 9 intervals were found. However, the conjunction of them had \( p = 1.19 \) only. Therefore only 3 best intervals were selected, with \( p = 1.48 \) and \( r = 0.32 \).

<table>
<thead>
<tr>
<th>Observed values</th>
<th>Data type</th>
<th>Used as</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Accident-related information</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hour</td>
<td>numerical</td>
<td>24 input binary variables</td>
</tr>
<tr>
<td>Day</td>
<td>numerical</td>
<td>7 input binary variables</td>
</tr>
<tr>
<td>Month</td>
<td>numerical</td>
<td>12 input binary variables</td>
</tr>
<tr>
<td>Year</td>
<td>numerical</td>
<td>6 input binary variables</td>
</tr>
<tr>
<td>District</td>
<td>categorical</td>
<td>not used</td>
</tr>
<tr>
<td>Municipality size</td>
<td>numerical</td>
<td>6 input binary variables</td>
</tr>
<tr>
<td>Cause</td>
<td>categorical</td>
<td>11 input binary variables</td>
</tr>
<tr>
<td>Road type</td>
<td>categorical</td>
<td>10 input binary variables</td>
</tr>
<tr>
<td>Tariff group</td>
<td>categorical</td>
<td>25 input binary variables</td>
</tr>
<tr>
<td>Car make</td>
<td>categorical</td>
<td>not used</td>
</tr>
<tr>
<td><strong>Information about causing person</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Age</td>
<td>numerical</td>
<td>8 input binary variables</td>
</tr>
<tr>
<td>Sex</td>
<td>categorical</td>
<td>3 input binary variables</td>
</tr>
<tr>
<td>District</td>
<td>categorical</td>
<td>not used</td>
</tr>
<tr>
<td>Municipality size</td>
<td>numerical</td>
<td>6 input binary variables</td>
</tr>
<tr>
<td>Region</td>
<td>categorical</td>
<td>15 input binary variables</td>
</tr>
<tr>
<td>Accident at place of abode</td>
<td>binary</td>
<td>1 input binary variables</td>
</tr>
<tr>
<td><strong>Claim costs</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Paid</td>
<td>numerical</td>
<td></td>
</tr>
<tr>
<td>Additional expected</td>
<td>numerical</td>
<td></td>
</tr>
</tbody>
</table>
3.1 Comparison

The problem (3) formulated here is novel and the Fencing algorithm is the only solution so far. However, for a simple comparison a clustering based method was involved that can be described briefly as follows:

1. **Building above-average clusters from training data:** Best 20% records were extracted and clustered via the k-means algorithm. For each cluster, the diameter was calculated as the maximum of distance between the center and the record belonging to it.

2. **Finding above-average records in the testing data:** For each record, we test whether there is a cluster whose center is closer to the record than the c multiplied diameter of the cluster. Parameter c is set up so that the level of r is satisfied. So P is defined and r ensured.

3. **Calculation of p from provided data:** p is calculated.

Table 3 shows the results achieved by this method, and compares them with the Fencing Algorithm: the alternative method based on known algorithm provides less narrow results. However, the goal of this paper was not test proposed Fencing Algorithm, but to show that this algorithm is able to solve problem formulated above (3). More experiments with the k-means based approach might provide better results.

### Table 2: Mean absolute error of machine learning for different validation sets

<table>
<thead>
<tr>
<th>Validation sets #</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLM</td>
<td>3.43</td>
<td>3.37</td>
<td>3.27</td>
<td>3.33</td>
<td>3.30</td>
<td>3.22</td>
<td>3.35</td>
<td>3.27</td>
<td>3.43</td>
<td>3.41</td>
</tr>
<tr>
<td>MLP</td>
<td>3.35</td>
<td>3.29</td>
<td>3.24</td>
<td>3.30</td>
<td>3.23</td>
<td>3.10</td>
<td>3.27</td>
<td>3.28</td>
<td>3.44</td>
<td>3.34</td>
</tr>
</tbody>
</table>

### Table 3: Comparison of the Fencing algorithm with the k-means based approach

<table>
<thead>
<tr>
<th>Method</th>
<th>r (fixed)</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>means based approach (6 means)</td>
<td>0.32</td>
<td>0.96</td>
</tr>
<tr>
<td>means based approach (20 means)</td>
<td>0.32</td>
<td>0.94</td>
</tr>
<tr>
<td>Fencing algorithm (raw results)</td>
<td>0.48</td>
<td>1.19</td>
</tr>
<tr>
<td>Fencing algorithm (results after best interval selection)</td>
<td>0.32</td>
<td>1.48</td>
</tr>
</tbody>
</table>

5 Conclusion

The Fencing algorithm is a novel heuristic method for finding a subset of with above-average production. The main idea of the algorithm is to join intervals with high production and small mutual distance. The Fencing Algorithm has been successfully applied to insurance data. Further work has been discussed above.

### References


Mgr. Karel Macek
e-mail: karel.macek@fjfi.cvut.cz
Department of Mathematics
Czech Technical University in Prague
Faculty of Nuclear Sciences and Physical Engineering
Trojanova 13
120 00 Praha2, Czech Republic