A Method for Analyzing High-dimensional Datasets

[A first approach]

Edwin Aldana-Bobadilla[1], Ivan Lopez-Arevalo[1], and J.L. Gonzalez[1] and Ana B. Rios-Alvarado[2]

[1] CINVESTAV Tamaulipas, Victoria, México
{ealdana,ilopez,jgonzalez}@tamps.cinvestav.mx
[2] Autonomous University of Tamaulipas, Victoria, México
arios@uat.edu.mx

Abstract. Due to technological advances to massively collect data, the last decades have raised several challenges for data analysis. Data reduction is one of the most important. Typically such reduction can be made in two ways: decrease the number of instances (elements) of the dataset and decrease the number of required attributes (columns) to describe each of these instances. In the last case, methods as Principal Component Analysis (PCA) and Low Variance Filter are usually applied. They are based on statistical measures that allow us to obtain the set of features or attributes with the minimal correlation between them. Since the dataset may have uncorrelated variables that cannot be eliminated, the number of obtained features might not always be appropriate. A dataset with this characteristic may represent a performance problem when there are constraints of time or space. To avoid this, we propose a method that allows us to represent a n-dimensional instance as a numerical data in one-dimensional space. We show that this transformation preserves the properties of the original dataset and thus, it can be suitable for many applications where a high reduction is required.

Keywords: Feature Selection, Dimensionality Reduction, Density Estimation

1 Introduction

Multivariate data analysis represents challenges in both theoretical and empirical levels. Until now, several methods for dimensionality reduction like Principal Component Analysis [6], Low Variance Filter [1] and High Correlated Columns [7] has been proposed. In this regard, we propose a method that allows us to represent a n-dimensional instance as a numerical data in one-dimensional space. It is compulsory that such representation preserves the information conveyed by the original data. Our proposal is based on a "discretization" of the data space. We resort to the idea of quantiles which are cutpoints dividing a set of observations or instances into equal sized intervals [4]. Usually the quantiles are defined over one-dimensional space. A set of instances in such space may be grouped by the quantile to which them belong. In this sense, a quantile represents all those instances that are close to each other. Many operations with an instance may be approximated by the range defined by its quantile. For example, assumming many instances $x_i \in \mathbb{R}$ which belongs to the quantile q defined by the interval $[0.25, 0.30)$. The operation $f(x_i) = sin(x_i) \forall x_i \in q$ may be approximated by $f(\underline{q}) = sin(0.25) = 0.004$, $f(\overline{q}) = sin(0.29) = 0.005$ or even $f(q) = f(\underline{q}) + f(\overline{q})/2 \approx 0.004$. We can see that the effectiveness of this approximation depends on the size of the interval that defines to q. Thus an appropriate size value must be determined.

Based on the above, we assume that the instances of a dataset may be represented by a set of quantiles, which preserves the properties of such instances when several operations are applied. As mentioned, typically the quantiles are defined over one-dimensional space. We propose a methodology that extends such definition to n-dimensional space. We want to project every instance of the dataset to its corresponding quantile. Every quantile is identified by an unique numerical code. The projection of an instance to its corresponding quantile allows us to obtain the numerical representation of such instance (encoding instances). The set of encoded instances will be the new dataset, we show experimentally that this set preserves the information and properties of the original dataset when several operations are applied.

The rest of this work is organized as follows: In Section 2 we present the main idea about dataset discretization based on quantiles. In Section 3 we explain how to map a data instance into one-dimensional space through its corresponding quantile. Then, in Section 4, we show the experimental methodology to measure the effectiveness of our method and present the results. Finally, in Section 5 we conclude and mention some future work.

2 Data Discretization

Given a dataset Y in a one-dimensional space, we can divide its space into a set of quantiles as it is shown in Figure 1.
In this case, a quantile q_i is an interval of the form $q_i = [\underline{y_i}, \overline{y_i}]$ where $\underline{y_i}$ and $\overline{y_i}$ are the lower and upper limits of q_i respectively. To determine the values of $\underline{y_i}$ and $\overline{y_i}$ a quantile width δ must be defined. Such value is given by:

$$\delta = \frac{|max(Y) - min(Y)|}{N} \tag{1}$$

where N is a prior value of the desired number of quantiles. Based on the above, the first quantile is defined as a half-closed interval of the form:

$$q_1 = [min(Y), min(Y) + \delta) \tag{2}$$

and the subsequent quantiles are defined as:

$$q_i = \begin{cases} [\overline{y}_{i-1}, \overline{y}_{i-1} + \delta] \text{ if } i = N \\ [\overline{y}_{i-1}, \overline{y}_{i-1} + \delta) \text{ otherwise} \end{cases} \tag{3}$$

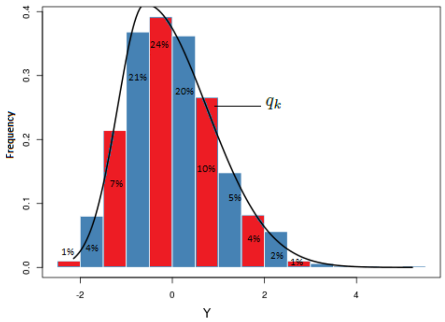

Fig. 1: A possible division of the space of an one-dimensional random variable Y. The quantile q_k contains a proportion of instances of Y.

The idea could be extended to higher dimensional data, in which case, a quantile will be a n-dimensional partition of the data space. In this case, given a dataset $Y \in \mathbb{R}^n$ with instances of the form $y = [y_1, y_2, ..., y_n]$, we can divide the data space into a set of n-dimensional quantiles.

Each n-dimensional quantile is composed by a set of intervals that determine the upper and lower limits for each dimension. Such definition is expressed as:

$$q_i = [[\underline{y}_{i1}, \overline{y}_{i1}], [\underline{y}_{i2}, \overline{y}_{i2}], \ldots, [\underline{y}_{in}, \overline{y}_{in}]] \tag{4}$$

where $\underline{y}_{i,k}$ and $\overline{y}_{i,k}$ are the lower and upper limit of q_i in the k^{th} dimension and the width of each interval is now given by:

$$\delta_k = \frac{|max(Y_k) - min(Y_k)|}{N} \tag{5}$$

The variable Y_k corresponds to the data in the k^{th} dimension. We can generalize the way to determine the limits of a quantile when $Y \in \mathbb{R}^n$ as:

$$q_1 = \begin{bmatrix} [min(Y_1), min(Y_1) + \delta_1) \\ [min(Y_2), min(Y_2) + \delta_2) \\ \vdots \\ [min(Y_n), min(Y_n) + \delta_n) \end{bmatrix}^T \tag{6}$$

for the first quantile, and:

$$q_i = \begin{cases} \begin{bmatrix} [\overline{y}_{(i-1),1}, \overline{y}_{(i-1),1} + \delta_1] \\ [\overline{y}_{(i-1),2}, \overline{y}_{(i-1),2} + \delta_2] \\ \vdots \\ [\overline{y}_{(i-1),n}, \overline{y}_{(i-1),1} + \delta_n] \end{bmatrix}^T & \text{if } i = N \\ \\ \begin{bmatrix} [\overline{y}_{(i-1),1}, \overline{y}_{(i-1),1} + \delta_1) \\ [\overline{y}_{(i-1),2}, \overline{y}_{(i-1),2} + \delta_2) \\ \vdots \\ [\overline{y}_{(i-1),n}, \overline{y}_{(i-1),1} + \delta_n) \end{bmatrix}^T & \text{otherwise} \end{cases} \tag{7}$$

for subsequent quantiles. Note that in general a partial order is formed corresponding to the left-to-right precedence relation where $q_i < q_j$ if $\exists k$ such that $\overline{y}_{i,k} < \underline{y}_{j,k}$, for $k \leq n$. Thanks to the precedence relation, we can assign to each quantile a numerical code that preserves the partial order. To illustrate the above idea, in Figure 1 the leftmost quantile can be identified as 1 while the rightmost quantile can be identified as 10. Even any other numerical basis can be used (instead of 10 base), as long as the order is preserved.

Now consider Figure 2 which illustrates a possible encoding of three-dimensional quantiles. The quantile code is formed by combining of the sequence number of the intervals that define q_i in each dimension. It means, that a three-dimensional quantile encoded as 111 is defined by the leftmost intervals per dimension. Likewise, a three-dimensional quantile encoded as 333 is defined by the rightmost intervals per dimension. In such encoding, we can see that the quantile encoded as 113 precedes the quantile encoded as 323. The evident precedence order given in one-dimensional space is preserved in higher order spaces.

3 Mapping High-dimensional Data to One-dimensional Space

So far, we have shown how to divide the data space into a set of n-dimensional quantiles. Now, the idea is to map the dataset instances to the quantile to which they belong. Those instances that belong to q_i can be represented (encoded) through the code assigned to it. Thus, given a set of quantiles Q defined from a n-dimensional dataset Y, a one dimensional dataset Y', composed only by encoded instances, can be obtained. Table 1 shows a set of quantiles Q for an hypothetical dataset $Y \in \mathbb{R}$.

Possible instances of Y have been encoded, according to the quantile code to which they belong, as it is shown in Table 2.

From this example the encoded dataset Y' is:

$$\begin{aligned} Y' = \{&01, 01, 02, 02, 03, 03, 04, 04, 05, 05, \\ &07, 07, 08, 08, 09, 09, 10, 10, 10, 06\} \end{aligned} \tag{8}$$

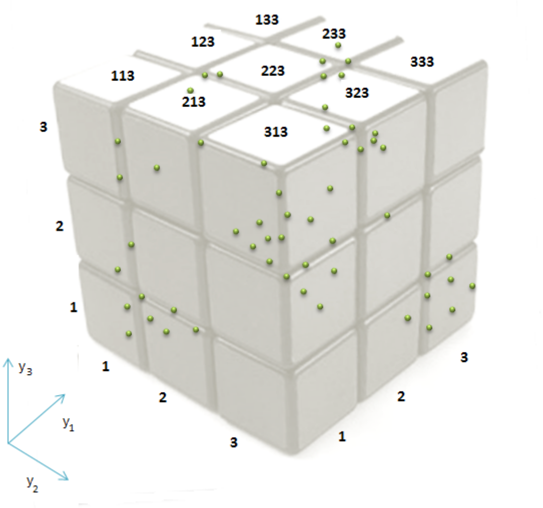

Fig. 2: Possible encoding of three-dimensional quantiles

Code	Lower Upper
01	[-0.63 -0.60)
02	[-0.60 -0.58)
03	[-0.58 -0.55)
04	[-0.55 -0.53)
05	[-0.53 -0.50)
06	[-0.50 -0.48)
05	[-0.48 -0.45)
08	[-0.45 -0.43)
09	[-0.43 -0.40)
10	[-0.40 -0.38]

Table 1: Example of quantile encoding for a one-dimensional dataset.

$\boldsymbol{Y}$	Code	$\boldsymbol{Y}$	Code
-0.63	01	-0.48	07
-0.62	01	-0.46	07
-0.60	02	-0.45	08
-0.59	02	-0.44	08
-0.58	03	-0.43	09
-0.57	03	-0.41	09
-0.55	04	-0.40	10
-0.54	04	-0.39	10
-0.53	05	-0.38	10
-0.52	05	-0.50	06

Table 2: Example of an encoding of a one-dimensional dataset.

The above idea can be extended to n-dimensional case. For instance, let Y be a dataset in $\mathbb{R}^3$. Assume that the space of the dataset in the k^{th} dimension (Y_k) is divided into five intervals (for illustrative purposes) where $min(Y_1) = -1.22$, $max(Y_1) = 1.94$, $min(Y_2) = -1.21$, $max(Y_2) = 2.70$, $min(Y_3) = -1.26$ and $max(Y_3) = 1.12$. Also, assume that $\delta_1 = 0.64$, $\delta_2 = 0.78$ and $\delta_3 = 0.48$. In this way, the leftmost quantile is comprised by the intervals set $[-1.22, -0.58)$, $[-1.21-0.43)$, $[-1.26-0.78)$, which correspond to the leftmost intervals per dimension. Using a decimal numerical code of two digits to identify every interval, the corresponding quantile can be encoded as $01-01-01$ or merely 010101. In Table 3, this and other quantiles are illustrated including their corresponding boundaries per dimension.

Quantile Code	$\boldsymbol{Y_1}$	$\boldsymbol{Y_2}$	$\boldsymbol{Y_3}$
010101	[-1.22 -0.58)	[-1.21 -0.43)	[-1.26 -0.78)
010103	[-1.21 -0.57)	[-1.21 -0.43)	[-0.79 -0.31)
010104	[-1.20 -0.56)	[-1.21 -0.43)	[0.17 0.65)
010201	[-1.21 -0.57)	[-0.43 0.35)	[-1.26 -0.78)

Table 3: Sample of quantiles for an hypothetical three-dimensional dataset.

Given the set of quantiles Q, assume an instance $y \in Y$ of the form $[-1.15, -0.5, -0.4]$. Since this instance lies into the quantile 010103, it can be represented by the quantile code.

Based on this representation process, we can obtain a one-dimensional dataset denoted as Y'. Note that the above representation method may involve a loss of information allegedly depending on the number of quantiles. This value is implicitly associated to the number of bins in which the space of Y_k (the dataset in the k^{th} dimension) is divided. In this regard, an appropriate selection of this value is compulsory. Typically, Sturges's rule is used [3,4]. Alternative rules, which attempt to improve the performance of Sturges's rule without a normality assumption, are Doane's formula [2] and the Rice rule [5]. In this paper, we prefer the Rice rule, which is to set the number of intervals to twice the cube root of the number of instances. In the case of 1000 instances, the Rice rule yields 20 intervals instead of the 11 recommended by Sturges' rule. One of our main goals in near future is to drive experiments to minimize the loss information.

Having defined the way to represent a n-dimensional data, in the next sections we show the experimental results which allow us to confirm that our proposal is promissory.

4 Experiments and Results

In this section, first we show in subsection 4.1 some preliminary results of the effectiveness of our method to preserve the properties of the original dataset in clustering analysis. Subsequently, in subsection 4.2, we evaluate the effectiveness through a wide sample of systematic experiments that allow us to generalize the results.

4.1 Preliminary Results: preserving clustering properties

We applied a non supervised classification algorithm over the well known Iris dataset (available at `https://archive.ics.uci.edu/ml/datasets/Iris`). This dataset (in what follows Y_{iris}) contains 3 classes of 50 instances each, where each class refers to a type of iris plant.

Figure 3 shows the iris data classes obtained by the k-means algorithm for three centers, using Y_{iris} and the encoded dataset Y'_{iris}. The plot was obtained using the *plotcluster* function of *fpc* R module with the default parameters.

Table 4 is the confusion matrix obtained for three classes. We can see that in this case the ratio of well classified instances is 80%.

	Reference class (Y_{iris})		
Predicted class Y'_{iris}	Class 1	Class 2	Class 3
class 1	39	6	0
class 2	11	35	4
class 3	0	9	46

Table 4: Confusion Matrix of k-means.

The above results show that the encoded dataset achieves to retain the information conveyed by the original data. However, these results are not enough to generalize this observation. In the following subsection, we present an experimental methodology that allows us to generalize the effectiveness of our proposal.

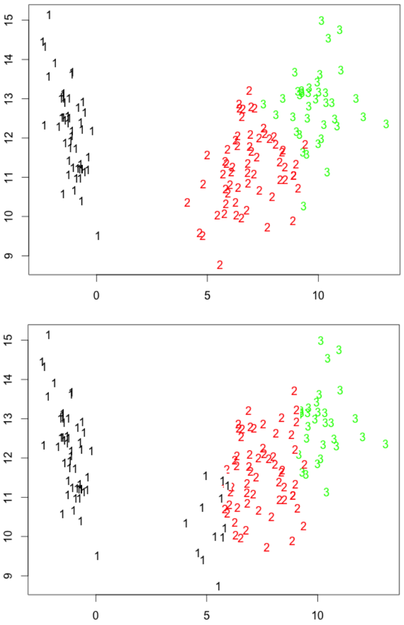

Fig. 3: Classification of k-means using Y_{iris} vs. encoded dataset Y'_{iris}.

4.2 General Results

In order to generalize the above results, we generated a wide set of synthetic datasets (about 5000) in $\mathbb{R}^n$ for $n = 2, 3, ...100$. To generate each dataset, a number of classes was defined a prior. For each dataset an encoded dataset (based on our proposal) was obtained. Subsequently k-means algorithm is applied to both non-encoded (Y) and encoded dataset (Y'). As mention the effectiveness is given by the percentage matching between the class labels found with Y dataset and the class labels found with Y'. The average result is shown in Table 5. For completeness, we show the confidence interval of the results with a -value of 0.05.

Table 5: Average Effectiveness

	matching(%)
Average	0.8895
Standard Deviation	0.2181
Lower Limit	0.8806
Upper Limit	0.8983
Confidence Level	95%

The experiments show that in average the encoded data allows us to preserve the properties (at least those associated to the proximity of instances) of the dataset in more than 80%.

5 Conclusions and future work

In this paper we have described a method to encode multivariate data. Such encoding, could be interpreted as a data reduction method, in a way that using the codes we are able to apply data mining methods and obtain similar results as using the original data. We applied the reduction method over iris data and results show good performance in classification. Also we show that statistically our method achieves to preserve about 80% of information conveyed by the original data.

However, there are some more hypotheses to prove in order to generalize the reduction method. For instance, we suspect that precision in classification tasks could improve if a finer number of quantiles is used; more experiments are needed in order to explore this idea. Another future research will be the inclusion of non numerical data types.

6 Acknowledgments

The authors acknowledge the support of Consejo Nacional de Ciencia y Tecnología (CONACyT), Centro de Investigación y Estudios Avanzados-CINVESTAV and Autonomous University of Tamaulipas.

References

1. K. A. Cox, H. M. Dante, and R. J. Maher. Product appearance inspection methods and apparatus employing low variance filter, Aug. 17 1993. US Patent 5,237,621.
2. D. P. Doane. Aesthetic frequency classifications. *The American Statistician*, 30(4):181–183, 1976.
3. R. J. Hyndman. The problem with sturges' rule for constructing histograms. *Monash University*, 1995.
4. R. J. Hyndman and Y. Fan. Sample quantiles in statistical packages. *The American Statistician*, 50(4):361–365, 1996.
5. D. M. Lane. Online statistics education: An interactive multimedia course of study. `http://onlinestatbook.com/2/graphing_distributions/histograms.html`, 2015. Accessed: 2015-12-03.
6. J. Shlens. A tutorial on principal component analysis. *arXiv preprint arXiv:1404.1100*, 2014.
7. L. Yu and H. Liu. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *ICML*, volume 3, pages 856–863, 2003.