Konrad-Zuse-Zentrum
für Informationstechnik Berlin

Takustraße 7
D-14195 Berlin-Dahlem
Germany

TOBIAS GALLIAT

# Clustering Data of Different Information Levels

Preprint SC 99-42 (December 1999)

# Clustering Data of Different Information Levels

Tobias Galliat[1,2]

[1] Konrad-Zuse-Zentrum Berlin, Takustr. 7, 14195 Berlin, Germany
e-mail: galliat@zib.de
[2] Risk-Consulting, Prof. Dr. Weyer, An der Kemperwiese 3a, 51069 Köln

**Abstract**

For using Data Mining, especially cluster analysis, one needs measures to determine the similarity or distance between data objects. In many application fields the data objects can have different information levels. In this case the widely used Euclidean distance is an inappropriate measure. The present paper describes a concept how to use data of different information levels in cluster analysis and suggests an appropriate similarity measure. An example from practice is included, that shows the usefulness of the concept and the measure in combination with KOHONEN'S Self-Organizing Map algorithm, a well-known and powerful tool for cluster analysis.

**Keywords.** cluster analysis, Data Mining, data preprocessing, information theory, missing values, Self-Organizing Maps, similarity measures.

## 1 Introduction

Using Data Mining [6] methods in practice, one often has the problem how to deal with vague, inaccurate, possibly false or even missing values in the data. While the first three cases are more or less neglected in research, even the problem of *missing values* has not been solved [2]. In situations, where it makes no sense or is not practicable to replace missing values by a 'typical' value that usually depends on the given data distribution as, for example, the mean value, the only way out is to code them into a new value of the related attribute[1]. But by doing this, the question arises, how to deal with this special values when measuring the similarity between different data objects.

The present paper tries to answer this question by introducing the concept of information levels and by describing a method—including an appropriate similarity measure—that realizes this concept in Data Mining, especially in cluster analysis. The suggested approach is general in the sense that not only missing values are considered, but also other kinds of values with a lower information level as, for example, vague or inaccurate values.

[1] Some authors simply suggest, not to use data objects with missing values. But this is impossible in practice, because there it is quiet usual, that almost each data object contains a missing value for at least one of its several attributes.

In Section 2 we motivate the use of information levels—given in the data implicitly—in Data Mining. We describe a concept and proper data transformations, to represent the information levels explicitly in the data. In Section 3 we show, how existing similarity measures can be transformed into a measure, that directly considers the explicit information levels, without loosing the advantages of the original measure. Based on this new kind of measure, we introduce a general method that uses the explicit information levels within cluster analysis. Finally, in Section 4, an illustrative example is given that shows the usefulness of the suggested method within the Self-Organizing Map algorithm [5, 3] for clustering high dimensional data of different information levels.

## 2 Different information levels

It is obvious, that data can represent different amounts of information, but it is not so evident, why it is important to consider this fact in Data Mining, especially in cluster analysis. Therefore we will first give a short motivation, before the general concept is introduced.

**Motivation** The following two examples from practice illustrate, why it is necessary to consider the implicitly given information levels, when performing a cluster analysis.

**Example: Insurance business** An insurance company wants to cluster their customers. For simplicity we suppose, that each customer is only described by ten attributes[2] $a_1, \ldots, a_{10}$ and that each attribute has the valid values 'no' and 'yes', coded by 0 and 1. It is also possible, that the value of an attribute is missing. For each customer $x$, the value of attribute $a_i$ is denoted by $x_i$. Because we want not to use the missing values for the similarity measurement, we use a weighted Euclidean distance (with $Q(x) := \{\, i \,|\, x_i \in \{0,1\}\})$ :

$$d_{\text{weighted-Euclid}}(x,y) := \sqrt{\frac{1}{|Q(x) \cap Q(y)|} \sum_{i \in Q(x) \cap Q(y)} (x_i - y_i)^2}.$$

We further suppose that there are customers called $v$, $w$, $y$ and $z$, with the following attribute values (missing values are denoted by 'm'):

| | $a_1$ | $a_2$ | $a_3$ | $a_4$ | $a_5$ | $a_6$ | $a_7$ | $a_8$ | $a_9$ | $a_{10}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| $v$ | 1 | 1 | 1 | $m$ | $m$ | $m$ | $m$ | $m$ | $m$ | $m$ |
| $w$ | 1 | 1 | 0 | $m$ | $m$ | $m$ | $m$ | $m$ | $m$ | $m$ |
| $y$ | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 |
| $z$ | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 |

While the customers $v$ and $w$ have for only two attributes the same valid value, $y$ and $z$ have six valid hits. If we compute the weighted Euclidean distance between $v$ and $w$, we get the value 0.58, while for the distance between $y$ and $z$ we get 0.63. Therefore the similarity between the first both customers seems to be greater than between the last two. But this is a contradiction to the fact, that

[2] In reality there are often hundred or more attributes.

$y$ and $z$ have three times more equal valid attribute values than $v$ and $w$. Note that only the valid values represent full information, while the missing values represent no or even less information[3]. Obviously the Euclidean distance—weighted or not[4]—is not able to consider these different information levels. When performing a cluster analysis, this has negative results: The clustering process is always determined by groups of objects with great inter-similarity. If now the similarity between objects with many missing values is over-valued, the clustering is heavily determined by objects with less information amount, instead of those with full information.

**Example: Opinion poll** In an opinion poll 4000 citizens are asked ten questions $a_1, \ldots, a_{10}$ as, for example, 'Do you agree with the policy of the government?'. They can always give five possible answers: 'I totally agree', 'I rather agree', 'I am indifferent', 'I slightly disagree' and 'I totally disagree'. The answers are coded as 1, 0.75, 0.5, 0.25 and 0. For a person $x$, the answer of question $a_i$ is denoted by $x_i$. It is also possible not to answer. But because we have already examined the missing value case in the example from insurance business, we suppose now, that all asked people have given one of the five possible answers. Note, that usually there are always a lot of people who are indifferent and that indifference normally represents less information than a clear vote.

Suppose that there are four persons, called $v$, $w$, $y$ and $z$. Let $v$ and $w$ be indifferent for most questions, while $y$ and $z$ have always clear, but slightly different votes:

| | $a_1$ | $a_2$ | $a_3$ | $a_4$ | $a_5$ | $a_6$ | $a_7$ | $a_8$ | $a_9$ | $a_{10}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| $v$ | 1 | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 |
| $w$ | 0 | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 |
| $y$ | 1 | 1 | 1 | 0 | 0.25 | 0.75 | 1 | 1 | 0.75 | 1 |
| $z$ | 0 | 1 | 0.75 | 0.25 | 0 | 1 | 0.75 | 1 | 1 | 1 |

The Euclidean distance between $v$ and $w$ is 1.00, while it is 1.17 between $y$ and $z$. Even if we choose another coding, for example, 1, 0.9, 0.5, 0.1 and 0, the second distance will be greater than the first one. But this is not what we want: The opinions of $y$ and $z$ are more or less the same and so they should have a great similarity value. The opinions of $v$ and $w$ are also quiet the same, but also indifferent, i.e. the similarity of $v$ and $w$ is based on less information, than the similarity between $y$ and $z$ and should therefore be smaller. But the Euclidean distance generates exactly the opposite.

Note that even if one uses other similarity or distance measures, the described problems will still be there, as long as one does not consider the implicit information levels of the data. But to consider these levels, one has to develop a concept, how to represent them in the data explicitly.

[3] Sometimes the fact, that a value is missing, implies important information (e.g. a customer with many missing values could be a bad risk). But usually 'mining' such informations is the task of classification rather than cluster analysis.

[4] The situation becomes even worser, if we substitute the missing values by 'typical' values or special values (see introduction) and use the normal Euclidean distance. The results are so called *blind spots* or *no information clusters* (see Section 4).

**A concept for representing information levels** Suppose that each data object $x$ is described by $q$ different attributes $a_1, \dots, a_q$. For each attribute $a_i$, a set $A_i$ of valid values is defined and for each data object $x$, we denote the value of attribute $a_i$ by $x_i$, with $x_i \in A_i \cup \{missing\}$. Furthermore $x_{*i}$ denotes the information level of $x_i$, with $x_{*i} \in [0,1]$. If $x_i = missing$, we always set $x_{*i} := 0$. For all other values, we choose a suitable $x_{*i} > 0$. The larger the amount of information, represented by $x_i$, the larger the value for $x_{*i}$. The highest information level is indicated by $x_{*i} = 1$. It is obvious that by this construction, we can easily represent vague and inaccurate values, or values that describes a kind of indifference. Note, that the explicit information levels $x_{*i}$ have to be chosen by an expert, who is familiar with the data. False values (i.e. $x_i \notin A_i$) should always be transformed to $missing$, so that $x_{*i} = 0$. The following example illustrates the concept:

**Example: Car manufacturer** A German car manufacturer wants to analyze his vendors. One of the vendors attributes denotes the sales in the last year. For some vendors there are only sale estimates, for others there are no sales information (information level is zero). In the last five years the differences between the estimates and the real sale values were in 20% of the cases significant. So the data analyst decides to fit the estimates with an information level of 0.8 instead of 1. Note, that it is possible, that there are vendors with equal sale values, but different information levels.

By now, we have introduced a natural concept to represent different levels of information in our data explicitly[5]. Before we use this concept within Data Mining methods, we have to perform some standardized transformations.

**Standardized Transformations** The necessary transformations depends on the attribute types. We have to distinguish metric, ordinal and nominal attributes. For metric attributes $a_i$ we suppose that they are bounded, i.e. there exist $l_i, u_i \in R$, with $l_i \leq v \leq u_i$ for all $v \in A_i$, while for ordinal and nominal attributes there only exists a finite number of valid values, i.e. $A_i$ is finite.

Let $a_i$ be a metric variable and suppose for simplicity that $A_i = [l_i, u_i]$. The following transformation[6] normalizes $x_i$ on the interval $[0,1]$:

$$T_{[0,1]}(x_i) = \begin{cases} \frac{x_i - l_i}{u_i} & \text{if } x_i \in A_i \\ 0 & \text{if } x_i = missing. \end{cases}$$

This transformation is reversible, because we can separate the original zeros from the transformed $missing$ values by checking the information level $x_{*i}$.

Let $a_i$ be an ordinal variable, i.e. $A_i = \{v_{i_1}, \dots, v_{i_{k_i}}\}$ with an arbitrary $k_i \in \mathbf{N}$ and an ordering $v_{i_1} \leq \dots \leq v_{i_{k_i}}$. For simplicity, we suppose $A_i \subset \mathbf{R}$.

[5] The reader, familiar with fuzzy logic, will recognize the similarities. But concepts from fuzzy logic are often used in cluster analysis in a different setting [4]. To avoid confusion, the present concept is formulated in a way, such that it is independent of fuzzy logic concepts.

[6] If it is possible to compute the mean-value $\mu_i$ and the standard deviation $\sigma_i$ for attribute $a_i$, the following transformation should be done previously: $T(x_i) = \frac{x_i - \mu_i}{\sigma_i^2}$

Setting $l_i = v_{i_1}$ and $u_i = v_{i_{k_i}}$, we can normalize $x_i$ by the same transformation $T_{[0,1]}$ as in the metric case.

Let $a_i$ be an nominal variable, i.e. $A_i = \{v_{i_1}, \ldots, v_{i_{k_i}}\}$ with an arbitrary $k_i \in \mathbf{N}$. Because there is no natural ordering on $A_i$, we have to dichotomize this attribute, i.e. we have to replace $a_i$ by $k_i$ new attributes $a_{i,v_{i_1}}, \ldots, a_{i,v_{i_{k_i}}}$, with

$$x_{i,v_{i_r}} = \begin{cases} 1 & \text{if } x_i = v_{i_r} \\ 0 & \text{else} \end{cases}, \qquad r = 1, \ldots, k_i$$

Finally we set $x_{*i,v_{i_r}} = x_{*i}$ for $r = 1, \ldots, k$.

In the next section we describe a method, how to use the explicit information levels within cluster analysis.

# 3 Information levels within cluster analysis

If we want to use information levels within cluster analysis, we need similarity measures that consider the different levels adequately.

Suppose we take a traditional measure $d(x, y)$ into account, for example, the Euclidean distance, that computes the similarity or distance between the two data objects $x$ and $y$ based on an arbitrary number $q$ of ordinal or metric attributes[7], all normalized to $[0, 1]$. The explicit information levels $x_{*i}$ and $y_{*i}$ are already defined for each attribute $a_i$. It is obvious that such a measure is not able to consider the explicit information levels adequately[8]. So we have to construct a new measure. Such a measure should match the following requirements:

- It should behave as much as possible as the given measure $d(x, y)$. Mainly in the case, when all information levels of $x$ and $y$ are 1, the similarity value should be the same for both measures[9].
- If $d(x, y)$ is differentiable, also the new measure should be differentiable. This is necessary, if one wants to use the measure within some popular cluster algorithms as, for example, the c-mean-algorithm [2].
- The new measure should compute the similarity based on the attribute values and the corresponding information levels directly as a whole. It seems to be not sufficient first to compute the similarity based on the attribute values and afterwards to weight this value by something like an 'information level factor'. Such an indirect approach opens the door for arbitrariness.

[7] Nominal attributes are dichotomized as described in the previous section.

[8] One could have the following simple idea: Consider the information levels as additional attributes, i.e. a data object $x$ will be described by $x_1, \ldots, x_q$ *and* $x_{*1}, \ldots, x_{*q}$. But one easily checks, that this approach does not solve the problems described in the previous section.

[9] This requirement is necessary to ensure the acceptance of the new measure. Many researchers and practicians use their own, special measures. They should be able to use them—only slightly changed—together with the concept of information levels.

While the first two requirements cause no restriction for the original measure $d(x,y)$, the last requirement makes it necessary to suppose, that $d(x,y)$ is as a function of $q$ one-dimensional measures $d_i(x_i, y_i)$:

$$d(x,y) := f(d_1(x_1,y_1),\ldots,d_q(x_q,y_q))$$

Note, that many popular distance measures have this feature. For the Euclidean distance we have $d_i(x_i,y_i) = (x_i - y_i)^2$.

After this preliminary remarks, we can introduce the new measure $d^*(x,y)$, based on the given measure $d(x,y)$.

**A similarity measure that directly considers information levels** There are several possibilities to define measures, that use explicit information levels, but the most of them do not match our requirements. After exhaustive research, we suggest the following measure that directly considers information levels and matches all requirements:

$$d^*(x,y) := f(d_1^*(x_1,y_1,x_{*1},y_{*1}),\ldots,d_q^*(x_q,y_q,x_{*q},y_{*q})),$$

with

$$d_i^*(x_i,y_i,x_{*i},y_{*i}) = \log_2\left(2 - \left(2 - 2^{d_i(x_i,y_i)}\right)\sqrt{x_{*i}y_{*i}}\right)$$

For the eucledian distance the measure simply denotes as:

$$d^*(x,y) := \sqrt{\sum_{i=1}^{q} \log_2\left(2 - \left(2 - 2^{(x_i-y_i)^2}\right)\sqrt{x_{*i}y_{*i}}\right)}$$

If we look at the motivating examples and use the Euclidean distance as original measure $d(x,y)$, we see that the measure $d^*(x,y)$ fits our needs:

In the example from insurance business, we have $d^*(v,w) = 2.83 > d^*(y,z) = 2.45$. In the opinion poll example, we choose $x_{*i} = 0.5$ as information level at indifference ($x_i = 0.5$). Then we get also $d^*(v,w) = 2.50 > d^*(y,z) = 1.17$. Note that even for a higher information level for indifference, we will get the same qualitative result.

By construction, the measure $d^*(x,y)$ depends not only on the values $x_i$ and $y_i$, but also on the corresponding information levels $x_{*i}$ and $y_{*i}$. Therefore it is necessary to define, how to adapt information levels.

**Adaptation of information levels** In most cluster algorithms a kind of cluster representative—called cluster center or codebook vector—is computed and refined during the iterative process. Usually an object is projected to that cluster, which representative is most similar to it. Therefore it is necessary to assign information levels—one for each attribute—to each representative and to adapt them during the process. The adaptation will be done on the same way as usual, but with one difference: If an object $x$ has a missing value for attribute $a_i$, the value $x_i$ will usually not be used for adapting the representatives. But the value $x_{*i} = 0$ has to be used in the adaptation process, because he represents information and so has effects on the clustering quality.

# 4 Information levels and Self-Organizing Maps: A cluster example

In order to show the benefits of the information level concept and the suggested measure, we shortly present an application from practice.

We are interested in a clustering of objects from the field of chemistry, which are characterized by a hierarchical six-digit coding system, like e.g. books in a library, each code describing a certain property. Although each object can be characterized by an arbitrary number of this codes, in reality an object is often described by less than ten codes. Because the coding is done by several different persons, with different backgrounds, we can not be sure that each object is fully characterized by his coding. Therefore there are always two possible reasons, why an object is not characterized by a certain code: The object does not have the related property, or the object has the property, but the person, who did the coding, has forgotten to assign the related code to the object.

Because we have just about 30000 objects, we only look at the first three digits. We consider each three-digit code as many times as it appears in the six-digit coding[10]. By doing this, one observes that 36 three-digit-codes appears quiet often, while the other codes are rather rare. We therefore introduce an attribute for each of these 36 codes that denotes the number of appearance in the coding of the object. A typical object has for only $10\% - 20\%$ of these attributes a value greater than zero. The zero value represents the 'missing' of the related code.

Our task is to cluster the almost 30000 objects, using the Self-Organizing Map (SOM) algorithm [5, 3] in combination with $u$-matrix visualization [1]. We will show the results of the following four different approaches:

1. Use the Euclidean distance based on the 36 attributes.
2. Use the euclidian distance based on the 36 attributes, but with the zero values as missing values, i.e. the zeros are not used for the distance computing and the adaptation process.
3. Use the information level concept together with the suggested measure based on the Euclidean distance. Set the information level of the zero values to 0 (interpretation as missing values) and to 1 for all values greater than zero.
4. Use the information level concept together with the suggested measure based on the Euclidean distance. Set the information level of the zero values to 0.5 (interpretation as vague values) and to 1 for all values greater than zero.

Because we are interested in a comparison between these approaches, we have fixed the parameters of the SOM as follows: We use a $11 \times 9$ grid with hexagonal topology and a Gaussian neighbourhood. We perform 120000 steps,

[10] If an object is described by the codes 234567, 234875, 237234, 395864 and 395892, we consider 234 twice, 237 once and 395 twice.

while adapting the neighbourhood only for the first 30000 steps and use a log-inverse descending learning rate. All attributes are normalized to $[0,1]$.

The $u$–matrix visualization of the resulting maps for the first two approaches that make no use of information levels, are shown in Figure 1:

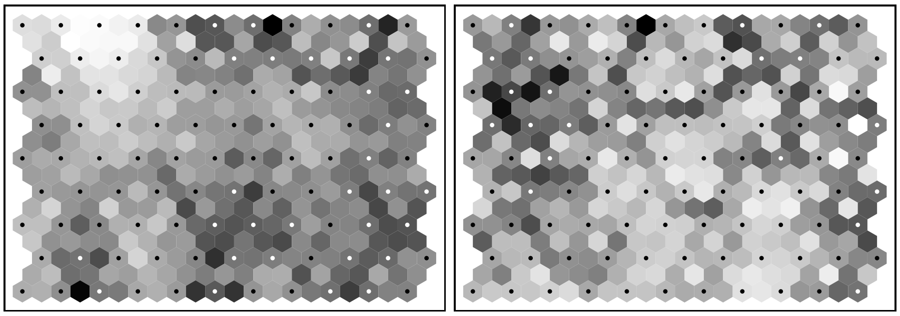

Figure 1: **U–matrix visualization of the SOM for the traditional approaches (Euclidean distance, no information levels)** Left hand side: zero values are normal values, Right hand side: zero values are missing values

The $u$-matrix of the first approach has the typical *blind spot*, i.e. a single bright area on a quite dark map. All the objects with zero values for almost all of the 36 attributes are projected to this area. Besides this *no information cluster*, it is more or less impossible to detect other clusters.

The $u$-matrix of the second approach has no *blind spot*, because the zero values are not used for the computing of the map. Although one is able to detect cluster borders, the clustering is quite bad.

Next we look at the $u$–matrix visualization of the resulting maps for the last two approaches that make use of information levels (see Figure 2). Note that the $u$-matrix visualization makes only use of the original 36 attributes and not of the information levels.

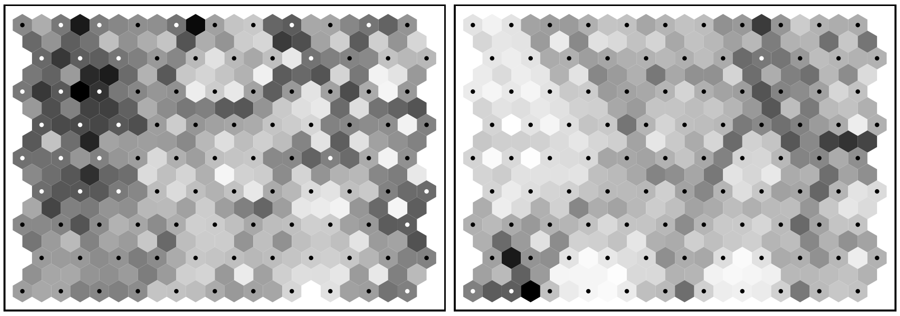

Figure 2: **U–matrix visualization of the SOM for the approaches that considers information levels** Left hand side: zero values are missing values, i.e. have an information level of 0, Right hand side: zero values are 'vague' values, i.e. have an information level of 0.5

In the case that we set the information level of the zero values to 0, i.e. interpret them as missing values, the clustering is rather similar to the one of the second approach. On the first view there seems to be no improvement. But if one looks closer, one detects that the process of cluster verification is easier in this case: If one has identified a cluster border, one normally looks first at the codebook vectors to detect the important attributes that are responsible for this cluster. The problem is that high values are not always a good indicator for importance. But this restriction is not valid, when using the information levels of the codebook vectors. Here a high level is an excellent indicator for the importance of the related attribute.

At last we use an information level of 0.5 for the zero values. The choice of the information level is rather uncritical: For $x_{*i}$ between 0.25 and 0.75 the results are not very different. The resulting clustering is much better than for the other approaches. The detection of borders is rather easy and the cluster verification makes no problems, when using the information levels of the codebook vectors. Experts of the application field have justified the resulting clustering.

# 5 Conclusion

The paper presents a powerful, generally applicable approach to consider information levels in the setting of Data Mining, especially in cluster analysis. Future work will focus on applications to large datasets with large numbers of missing or vague values, and the development of necessary extensions to use the suggested concept also within classification methods.

# References


[1] A.Ultsch and D.Korus. Integration of neural networks with knowledge-based systems. In *Proc. IEEE Int.Conf.Neural Networks, Perth*, 1995.

[2] B.D.Ripley. *Pattern Recognition and Neural Networks.* Cambridge University Press, 1996.

[3] G.Deboeck and T.Kohonen (Eds.). *Visual Explorations in Finance using Self-Organizing-Maps.* Springer, London, 1998.

[4] H.J.Zimmermann. *Fuzzy set theory – and its applications.* Kluwer Academic Publishers, 3rd edition, 1996.

[5] T.Kohonen. *Self-Organizing Maps.* Springer, Berlin, 2nd edition, 1997.

[6] U.M.Fayyad, G.Patetsky-Shapiro, P.Smyth, and R.Uthurusamy (Eds.). *Advances in Knowledge Discovery and Data Mining.* AAAI Press / The MIT Press, California, 1996.