International Journal of Pattern Recognition and Artificial Intelligence

K-means Clustering for Problems with Periodic Attributes

M. Vejmelka*

Institute of Computer Science, Academy of Sciences of the Czech Republic, Pod Vodarenskou vezi 2, 182 07 Prague 8, Czech Republic, vejmelka@cs.cas.cz

P. Musilek

Department of Electrical and Computer Engineering, University of Alberta, W2-030 ECERF, Edmonton, Alberta, T6G 2V4 Canada, musilek@ece.ualberta.ca

M. Paluš, E. Pelikán

Institute of Computer Science, Academy of Sciences of the Czech Republic, Pod Vodarenskou vezi 2, 182 07 Prague 8, Czech Republic, mp/emil@cs.cas.cz

The K-means algorithm is very popular in the machine learning community due to its inherent simplicity. However in its basic form it is not suitable for use in problems which contain periodic attributes, such as oscillator phase, hour of day or directional heading. A commonly used technique of trigonometrically encoding periodic input attributes to artificially generate the required topology introduces a systematic error. In this paper, a metric which induces a conceptually correct topology for periodic attributes is embedded into the K-means algorithm. This requires solving a non-convex minimization problem in the maximization step. Results of numerical experiments comparing the proposed algorithm to K-means with trigonometric encoding on synthetically generated data are reported. The advantage of using the proposed K-means algorithm is also shown on a real example using gas load data to build simple predictive models.

Keywords: Clustering Algorithms, Similarity measures, K-means, Periodic Attributes

1. Introduction

Clustering is one of the most important problems of data analysis, concerned with finding structures intrinsic to a set of unlabeled patterns. Typical goals of clustering include finding prototypes for homogeneous groups of data, description of unknown properties of data or detecting unusual data points. To aid the clustering process, it is common to use a distance measure to assess the (reciprocal of) similarity between patterns[5].

K-means[7] is one of the best known clustering algorithms. Due to its simplicity and ease of implementation, it has been used in numerous practical applications. There have also been many works that extend the original form of the algorithm

[*] Department of Cybernetics, Czech Technical University, Technicka 2, 166 27 Prague 6, Czech Republic

to improve its convergence[2], efficiency[6], cluster shape[3], etc. The distance measure used for K-means clustering is usually the Euclidean distance or the more general Minkowski distance[5], which provides a variety of metrics to suit many different types of attributes. However, these conventional distance measures cannot be used to effectively assess similarity of attributes that are periodic.

To deal with periodic attributes, trigonometric encoding[11] can be used to preprocess the data before clustering. However, this ad hoc approach has several disadvantages. First, the dimensionality of the problem is increased by one for each periodic attribute treated this way. Second, the non-linearities introduced by trigonometric functions distort the feature space and create artificial structures not present in the original data.

To avoid the above mentioned problems, we propose an alternative metric suitable for assessing distances of periodic attributes. This metric is first formally introduced and subsequently used in the development of a modified K-means algorithm. It is then generalized so that it is applicable to problems involving both periodic and non-periodic attributes. The main virtue of the suggested metric is its *unbiasedness*: it does not deform the feature space and thus does not impose any structure on the data analyzed. In addition, its use does not require the *dimensionality* of the feature space to increase.

The new algorithm has been compared to both standard K-means and K-means with trigonometric encoding of periodic attributes. The results of numerical simulations show the superiority of the proposed Hybrid K-means in problems with mixed periodic and non-periodic attributes.

The remainder of this paper is organized as follows. Sec. 2 provides a brief overview of partitioning clustering algorithms. The problems arising when clustering patterns with periodic attributes are introduced in Sec. 3. Subsequently, the K-means algorithm for clustering patterns with periodic attributes is formalized in Sec. 4. In Sec. 5, the new distance measure is combined with the Euclidean distance and a K-means algorithm suitable for clustering patterns containing both periodic and non-periodic attributes is formulated. Results of numerical simulations using the proposed algorithm and its variants are presented in Section 6. Finally, Sec. 7 provides the major conclusions and points to potential directions of future work. The Appendix contains the pseudocode of the algorithm introduced in the paper.

2. Clustering Algorithms

Clustering algorithms are broadly divided into *hierarchical* and *partitioning*[1]. While hierarchical techniques build clusters gradually (either by merging or splitting), partitioning algorithms form clusters directly by dividing data into several subsets. In partitioning algorithms, that are the subject of this work, certain heuristics are used to iteratively optimize the location and size of the subsets. There are different reallocation schemas facilitating this optimization process that can be grouped into *probabilistic* clustering and clustering based on the use of an *objective* *function.* Typical instances of these approaches are mixture-density models and the K-means clustering, respectively. These two algorithms are briefly reviewed in the following text. In addition to the conventional view of K-means as an algorithm based on objective-function optimization, it is also described in probabilistic terms. This probabilistic description is later used to derive a new algorithm applicable to periodic attributes.

2.1. *Mixture-density Models*

Probabilistic clustering models assume that the data to be clustered come from a mixture model of several random populations. The clustering problem is to find the distributions and prior probabilities of these populations.

Assume that the prior probability $P(w_k)$ for cluster w_k,$k = 1, \dots, K$, and the form of conditional probability density $p(x|w_k, \theta_k)$ are known, with the unknown parameter vector θ_k. The mixture probability density for the whole data set is

$$p(x|\theta) = \sum_{k=1}^{K} p(x|w_k, \theta_k) P(w_k), \tag{1}$$

where $\theta = (\theta_1, \dots, \theta_K)$, and $\sum_{k=1}^{K} P(w_k) = 1$. After finding the parameter vector θ, the posterior probability for assigning a pattern to a cluster can be calculated using Bayes' theorem[12].

The unknown parameters that maximize the probability of generating all given observations can be found using maximum likelihood method, usually expressed in the following logarithmic form

$$l(\theta) = \sum_{i=1}^{N} \log p(x_i|\theta). \tag{2}$$

As the maximum of the log-likelihood equation at $\partial l(\theta)/\partial\theta_k = 0$ usually cannot be found analytically, iterative methods are required to approximate the solutions. The expectation-maximization (EM) algorithm is the most popular approach. It augments the observable features x_i^o by vector $x_i^m = (x_{i1}^m, \dots, x_{iK}^m)$ whose components are assumed missing and attain values $x_{ik}^m \in \{0, 1\}$ depending on whether $x_i = \{x_i^o, x_i^m\}$ belongs to the cluster k ($x_{ik} = 1$) or not ($x_{ik} = 0$). The log-likelihood of the complete data is

$$l(\theta) = \sum_{i=1}^{N} \sum_{k=1}^{K} x_{ik}^m \log \left[P(w_k) p(x_i^o|\theta_k)\right] \tag{3}$$

The standard EM algorithm generates a sequence of parameter estimates $\{\theta^0, \theta^1, \dots, \theta^*\}$, where $*$ represents the reaching of convergence criterion by executing the following steps[12]

(1) Initialize θ^0 and set $t = 0$;

(2) E-step: determine the expectation of the complete data log-likelihood

$$Q(\theta, \theta^t) = E[\log p(x^o, x^m|\theta)|x^o, \theta^t];$$

(3) M-step: Select a new parameter estimate that maximizes the Q-function,

$$\theta^{t+1} = \arg\max_{\theta} Q(\theta, \theta^t);$$

(4) Increment t and repeat steps E and M until convergence.

2.2. *K-means Clustering*

K-means is the best known clustering algorithm based on quantization error minimization[12]. The error can be described as the average distance between each pattern and its closest prototype

$$E(w) = \frac{1}{2}\sum_{i=1}^{N} \min_k (x_i - w_k)^2, \tag{4}$$

where E is the quantization error, $w_k, k \in \{1, \ldots, K\}$ is the prototype of cluster k, and $x_i, i \in \{1, \ldots, N\}$ is the i-th sample of the data set to be clustered.

The equation above can be reformulated by writing $s_w(i)$ for the subscript of the prototype closest to the sample x_i[2]. Then

$$E(w) = \frac{1}{2}\sum_{i=1}^{N} (x_i - w_{s_w(i)})^2, \tag{5}$$

The quantization error can be minimized using gradient descent algorithm $\Delta w = -\epsilon_t(\partial E(w)/\partial w)$, to perform either batch or online updates of the cluster prototypes, with learning rate ϵ_t. In this paper, the batch variant in the form

$$\Delta w_k = \sum_i \begin{cases} \epsilon_t(x_i - w_k) & \text{if } k = s_w(i) \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

is considered. This step is repeated until convergence which is guaranteed for suitable chosen learning rates ϵ_t.

2.3. *Probabilistic Framework of K-means*

The sum of squares appearing in (5) can be viewed as (a negative of) log-likelihood for normally distributed mixture model that is widely used in statistics. Therefore K-means can be understood under a general probabilistic framework[1]. Subsequently, K-means can be derived as an EM style algorithm. The following notation has been adopted from[2].

The equation (3) can be modified for the K-means algorithm by assuming the missing parameters to be the assignments of patterns x_i to the prototypes w_k. Instead of considering the expected value over the distribution on these missing parameters, the values of parameters w' that minimize the cost given their previous values w are considered

$$Q(w', w) = \sum_{i=1}^{N} \frac{1}{2}(x_i - w'_{s_w(i)})^2. \tag{7}$$

In the case of K-means, the new set of prototypes w' that minimizes $Q(w', w)$ can be found analytically by solving the equation $\partial Q(w', w)/\partial w'_k = 0$

$$w'_k = \frac{1}{N_k} \sum_{i:k=s_w(i)} x_i, \tag{8}$$

where N_k is the number of samples assigned to prototype w_k. This algorithm can be reformulated to a form similar to the gradient descent algorithm (6)

$$\Delta w_k = \sum_i \begin{cases} \frac{1}{N_k}(x_i - w_k) & \text{if } k = s_w(i) \\ 0 & \text{otherwise} \end{cases}, \tag{9}$$

where $\Delta w_k = w'_k - w_k$ and is thus equivalent to batch gradient descent[2], with the learning rate dependent on cluster size, $\epsilon_t = 1/N_k$.

3. Clustering with Periodic Attributes

In this section, periodic attributes are introduced and some examples of such attributes arising in practice are listed. A popular technique for dealing with periodic attributes is presented and a systematic error arising from its use is shown.

3.1. *Periodic Attributes*

Intuitively a periodic attribute is an attribute the topology of which can be naturally represented by a circle connecting the supremum and infimum of the set of values together. The length of segments between points on the circle correspond to our perception of distance. A fitting example is an attribute representing an hour (of day) with values $\langle 0, 24)$. What is the distance between hour 23:00 (11 o'clock PM) and 01:00 (1 o'clock AM) ? Events occurring at these times every day are commonly understood to be 2 hours apart, not 22 hours apart as would be indicated by their Euclidean distance.

Examples of periodic attributes conforming to the above considerations include azimuth, cyclic time indices (minute of hour), oscillator phases or the hue of the HSV color model (the saturation and value can be considered standard non-periodic attributes). The part of statistics mainly concerned with periodic attributes and spherical sample spaces is circular statistics. This subject has been treated at length by Mardia and Jupp[8] who suggest the *cosine distance* as a useful distance metric

$$d_{\cos}(\theta, \xi) = 1 - \cos(\theta - \xi), \tag{10}$$

where θ, ξ are angles in radians. Any periodic attribute can be mapped onto $\langle 0, 2\pi)$ and represented by an angle (a direction). To elucidate the relationship between a common technique employed in clustering periodic attributes, which shall be termed *K-means with trigonometric encoding* and the cosine distance, some additional quantities will be defined

$$\begin{aligned}\bar{C} &= \tfrac{1}{N}\textstyle\sum_{i=1}^{N}\cos(\theta_i),\\ \bar{S} &= \tfrac{1}{N}\textstyle\sum_{i=1}^{N}\sin(\theta_i),\end{aligned} \tag{11}$$

where θ_i are the samples of the periodic attribute mapped onto $\langle 0, 2\pi)$. Then the *mean direction* is defined as

$$\bar{\theta} = tan^{-1}\Big(\frac{\bar{S}}{\bar{C}}\Big). \tag{12}$$

The mean direction is similar to the mean of a set of samples in $\mathcal{R}$. It also has an associated *mean resultant length*

$$\bar{R} = \sqrt{\bar{C}^2 + \bar{S}^2}, \tag{13}$$

which characterizes the spread of points around the mean direction $\bar{\theta}$.

If the dispersion of a set of samples of a periodic attribute around a value α is computed as[8]

$$D(\alpha) = \frac{1}{N}\sum_{i=1}^{N}(1 - \cos(\theta_i - \alpha)), \tag{14}$$

then the mean direction $\bar{\theta}$ is the minimizer of the dispersion $D(\alpha)$ for the set of samples θ_i. The dispersion $D(\alpha)$ is in fact the sum of cosine distances from α to each θ_i.

The K-means algorithm with trigonometric encoding works by first transforming the periodic attribute $t \in \langle 0, D)$ by the pair of functions

$$\begin{aligned}c &= \cos(2\pi t/D)\\ s &= \sin(2\pi t/D)\end{aligned} \tag{15}$$

and applying the standard maximization step (8). Optionally, the values c, s can be linearly mapped onto the range $\langle 0, D)$ to preserve the scale of the attribute. The new prototype coordinate resulting from applying the maximization step (8) for the c dimension is exactly $\bar{C}$ and the new prototype coordinate for the s dimension is $\bar{S}$. The corresponding value in the original (unmapped) space is $\bar{t} = \frac{\bar{\theta} D}{2\pi}$ — the *mean direction* linearly scaled to $\langle 0, D)$. We have thus shown that given a set of samples $t_1, t_2, ..., t_N$, the K-means algorithm minimizes (14) which is in fact a sum of cosine distances from the prototype to each of the samples t_i mapped to $\langle 0, 2\pi)$.

3.2. *Effects of Trigonometric Encoding*

Problems involving periodic attributes are sometimes approached by mapping the periodic attributes using trigonometric functions (15) onto a circle as shown above, for example in[11,?]. The dimensionality of the clustering problem is effectively increased by one for each periodic attribute that is mapped in this way. More importantly, the non-linearities of the trigonometric functions distort the spatial relationships between patterns. The problem manifests itself in both steps of the K-means algorithm.

In the expectation step of K-means, distances are measured using the standard Euclidean metric, which measures distances along straight lines in the plane spanned by the encoding variables c, s (15). This means that the ratio of distances between points a, c and a, b in Fig. 1 is $\sqrt{2}$, whereas a correct representation would indicate the ratio of distances to be 2. This distortion is an additional effect stemming from the use of Euclidean distance to partition the samples for the maximization step. The deformation caused by measuring distances along straight lines preserves the ordering of distances measured along the circle segments but not the ratios of these distances thus distorting the value of the cost function.

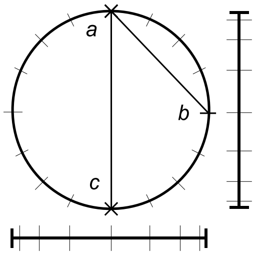
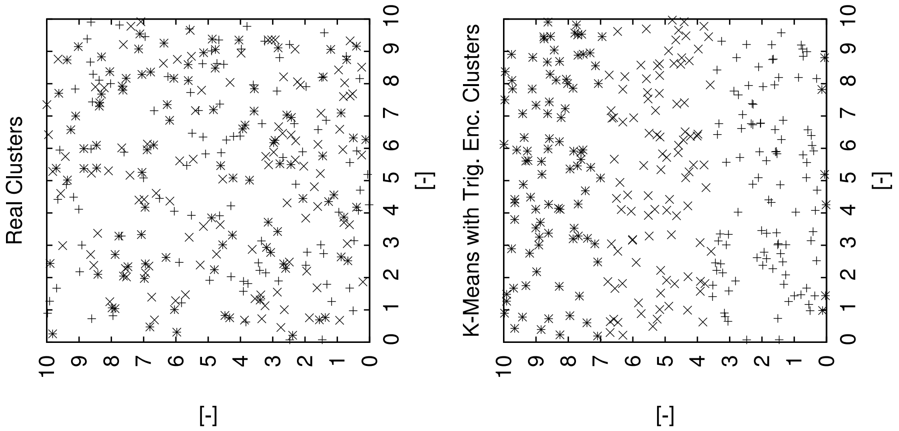


Fig. 1. Deformation of distances caused by trigonometric encoding.

In the maximization step, each coordinate is considered separately when optimizing prototype locations. The K-means algorithm thus works with the projections of the points on the unit circle into the horizontal and vertical axes shown in Fig. 1. Here, another type of spatial deformation comes into play, where the projected points equally spaced on the circle are not equally spaced on the axes. The result of the maximization step has been described in Sec. 3.1 and corresponds to using the cost function (14) based on the metric (10).

The resulting clusters are biased toward a structure artificially created by the selective contraction of the original feature space resulting from the use of the trigonometric encoding (15). This can be easily seen if there is no real structure in the input data as depicted in Fig. 2(a), where three 'clusters' were generated from a uniform distribution spanning the space $\langle 0, 10)2$. The first attribute, depicted on the vertical axis, is periodic with period $D = 10$ and the mapping (15) was applied to it making the problem three-dimensional. The range of the transformed variables c, s resulting from the application of the trigonometric encoding was linearly mapped to $\langle 0, D)$ so that scaling of the attribute was preserved. The clusters computed by

K-means with trigonometric encoding shown in Fig. 2(b) have a horizontal stripe structure. Even though runs with different seeds may produce slightly different configurations (e.g. the stripes are shifted), the bias in the results is clearly visible. Indeed for uniformly distributed data, any K-means algorithm, including the one proposed in this paper, will converge to some (meaningless) cluster configuration. However, the use of trigonometric encoding causes a systematic error that is present no matter what input is provided to the algorithm.



(a) Generated clusters.

(b) Clusters identified by K-means with trig. encoding.

Fig. 2. The clusters identified by the K-means algorithm with trigonometric encoding display a horizontal striped structure 2(b), the actual 'clusters' with all points drawn from a uniform distribution on $\langle 0, 10)2$ shown in 2(a).

In some cases, a smooth transformation of the feature space may be advantageous or may be required by the preprocessing method. In our case, however, deformation occurs as a by-product of a preprocessing method intended to approximate the spatial relationships of periodic attributes.

4. *K*-means for Periodic Attributes

In this section, a new EM problem will be formulated incorporating the features of a metric that does not distort the feature space. This metric has also been referenced in Mardia and Jupp[8]

$$d_{\mathrm{uni}}(\theta, \xi) = \pi - |\pi - |\theta - \xi||. \tag{16}$$

It can be perhaps more intuitively rewritten as

$$\rho_D(x, y) = \min\{|x - y|, D - |x - y|\}, \tag{17}$$

for $D = 2\pi$. The behavior of the metric is now clearer. If the term $|x-y|$ is smaller (or equal to) the term $D - |x - y|$, then the distance is exactly equal to the Euclidean distance. Otherwise the metric ‘wraps around’, which intuitively corresponds to passing through point $D \equiv 0$ when tracing the shortest distance path on the circle of values of the periodic attribute. The form (17) can be used with any period $D > 0$. This situation is depicted in Fig. 3.

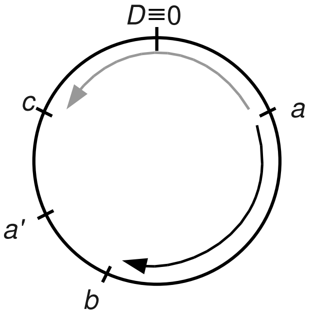


Fig. 3. Shortest paths between the points $0 < a < b < c < D$. Between a, b the shortest path does not pass through $D \equiv 0$. The shortest path between points a, c ‘wraps’ around the circle and passes through the point $D \equiv 0$. The point $a' = a + \frac{D}{2}$ separates the two regions.

In the following, the expectation and maximization steps will be modified to work with this metric. The derivations will be performed for a single periodic attribute with period D. The variables w, x, y will be used to represent elements of the set $\langle 0, D)$ of this periodic attribute.

4.1. *Expectation Step*

In the expectation step it is necessary to replace the Euclidean metric with the metric (17). The expectation step thus becomes

$$s_w(i) = \arg\min_k \rho_D(w_k, x_i). \tag{18}$$

The modified expectation step ensures that each sample x_i will be assigned to the cluster prototype closest in the sense of the metric (17).

4.2. *Maximization Step*

The maximization step computes a new cluster prototype location that minimizes the functional

$$Q(w, w') = \frac{1}{2} \sum_{i=1}^{N} \rho_D(w'_{s_w(i)}, x_i)^2, \tag{19}$$

where w represents the current cluster prototypes and w' represents the new cluster prototypes with respect to which the functional is minimized. This is not a simple convex quadratic function and has in general multiple local minima. The above minimization problem is solved separately for each cluster as the positions of the cluster centers are mutually independent. The set of patterns belonging to the cluster k is

$$X_k = \{x_i | s_w(i) = k\}, \tag{20}$$

where $N_k = \operatorname{card} X_k$ is the number of patterns assigned to the cluster k. An algorithm computing the global minimum of $Q(w, w')$ in $O(N_k \log N_k)$ time is presented. It is possible to optimize the algorithm further to run in $O(N_k)$ time by pre-sorting the input data and using indirect indexing to access the sorted array. However, for the sake of simplicity and clarity, only the $O(N_k \log N_k)$ algorithm is shown.

For each pattern $x_i \in X_k$, the interval $\langle 0, D)$ can be divided into two subintervals such that if the new cluster prototype were located in one of them, the metric would wrap around when computing the distance, and it would not wrap around if the new prototype was in the other one. This is illustrated in Fig. 4 for one pattern located in the interval $\langle \frac{D}{2}, D)$ and another pattern located in the interval $\langle 0, \frac{D}{2})$. The point where the shortest distance path changes for the pattern in the upper interval is $x_i - \frac{D}{2}$. The same happens for patterns located in the interval $\langle 0, \frac{D}{2})$, but the switch point is at $x_i + \frac{D}{2}$. A function can therefore be defined assigning a switch point to each x_i

$$m(x_i) = \begin{cases} x_i + D/2 \text{ if } x_i < D/2 \\ x_i - D/2 \text{ if } x_i \geq D/2. \end{cases} \tag{21}$$

The set $Y_k = \{y_i | y_i = m(x_i), x_i \in X_k\}$ is a set of switch points which split the interval $\langle 0, D)$ into at most $N_k + 1$ subintervals. If any two points coincide exactly, then their switch points coincide as well. This means that the number of unique subintervals is decreased by one. Starting at the point $w'_k = 0$, the metric wraps for all $x_i \in X_k$, $x_i > \frac{D}{2}$ and does not wrap in the interval $\langle 0, \frac{D}{2})$. Moving the potential

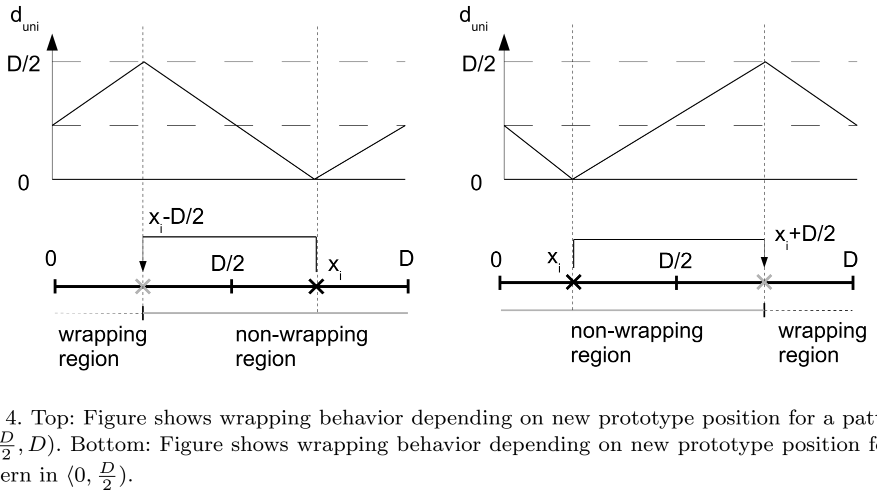


Fig. 4. Top: Figure shows wrapping behavior depending on new prototype position for a pattern in $\langle \frac{D}{2}, D)$. Bottom: Figure shows wrapping behavior depending on new prototype position for a pattern in $\langle 0, \frac{D}{2})$.

new prototype in the positive direction, N_k switch points are passed on the interval $\langle 0, D)$. All switch points of the patterns in the interval $\langle \frac{D}{2}, D)$ are reached before all switch points of the patterns in the interval $\langle 0, \frac{D}{2})$. The interval $\langle 0, D)$ can be divided into subintervals inside which the potential new prototype can move and the shortest distance path does not change discontinuously for any x_i. The optimal position in such a subinterval can be computed analytically

$$w'_k = \sum_{i=1}^{N} \begin{cases} x_i \text{ if } \rho_D \text{ does not wrap} \\ x_i - D \text{ if } \rho_D \text{ wraps and } x_i \geq \frac{D}{2} \\ x_i + D \text{ if } \rho_D \text{ wraps and } x_i < \frac{D}{2} \end{cases} . \tag{22}$$

It is possible that the new w' might not be in the range $\langle 0, D)$ and will have to be corrected by $\pm D$. The optimal position with the smallest cost function (19) over all the subintervals is then set as the new cluster prototype w'_k.

The optimal location and cost of the center if it lies in the first subinterval can be computed in $O(N_k)$ time using (22). However this is only necessary in the first subinterval. After the initial computation it possible to update the potential optimal location and it's associated cost in other subintervals in constant time $O(1)$. The total computational cost is thus dominated by the $O(N_k \log N_k)$ cost of the sorting procedure. The pseudocode for the algorithm is shown in the Appendix.

5. Hybrid *K*-means

A metric measuring distances in problems with mixed periodic and non-periodic attributes is presented and a Hybrid K-means algorithm suitable for finding clusters in such problems is described.

Using the distance metric (17) for periodic attributes and the Euclidean metric for non-periodic attributes, it is possible to define a metric for problems in which only some attributes are periodic. Let $\mathcal{I} \subset \{1, 2, ...M\}$ denote the set of indices of periodic attributes, where M is the total number of attributes. The set $\mathcal{D}$ is the set of periods of the periodic attributes $\mathcal{D} = \{D_j | j \in \mathcal{I}\}$. A metric representing distances in this space can be written as

$$\rho_{\mathcal{I},\mathcal{D}}(x, y) = \sqrt{\sum_{j=1}^{M} \begin{cases} \rho_{D_j}(x_j, y_j)^2 & \text{if } j \in \mathcal{I} \\ (x_j - y_j)^2 & \text{otherwise} \end{cases}} . \tag{23}$$

The function (23) is a metric as it satisfies all necessary conditions: it is non-negative and symmetric with respect to its arguments and zero if and only if $x = y$. The triangle inequality for the function (23) is satisfied for each coordinate individually, therefore also for the square root of the sum of squares. This metric is useful in practice when the clustering problem contains attributes which belong to different domains altogether (hour of day, temperature, wind speed) and some of the attributes are periodic. Introducing coefficients to weigh the contributions of each attribute to the total distance preserves the metric properties of function (23) so standard attribute scaling procedures can be applied. A modified version of the standard K-means algorithm can now be presented which is able to effectively find clusters in problems where only some attributes are periodic.

In the expectation step of the Hybrid K-means, the assignments of data points to centers are computed as

$$s_w(i) = \arg\min_k \rho_{\mathcal{I},D}(x_i, w_k). \tag{24}$$

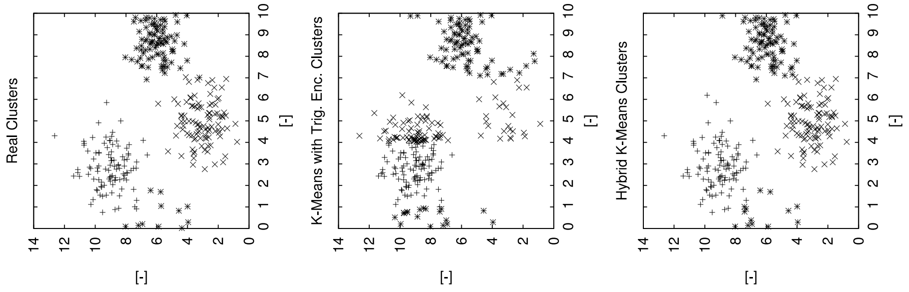
The Hybrid K-means algorithm uses the original standard K-means maximization step (8) for non-periodic attributes and the maximization step introduced in Sec. 4.2 for periodic attributes.

6. Experiments

In this section it is shown how the Hybrid K-means algorithm compares to K-means with trigonometric encoding on synthetic data sets containing both periodic and non-periodic attributes and on real data describing the consumption of natural gas as a function of time.

6.1. *Case Study*

The presented case study has two attributes so that the results can be visually examined. In the following intentionally constructed example, three clusters were generated from a normal distribution with $\sigma = 1$ and the respective prototypes $\{(3, 9), (5, 3), (9, 6)\}$ in the space $\mathcal{R}^2$ with one attribute periodic with $D = 10$, shown on the horizontal axes in Figs. 5(a), 5(b), 5(c). The original clusters, each containing 50 points are shown in Fig. 5(a).



(a) Original clusters (b) Trigonometric encoding (c) Hybrid K-means

Fig. 5. A case study of the problems caused by using trigonometric encoding with K-means.

To estimate the clusters using K-means with trigonometric encoding, (15) was first applied to each periodic attribute and then the range $\langle -1, 1\rangle$ of both resulting variables was linearly mapped to $\langle 0, D\rangle$ to preserve the scale of the attribute. The result from the K-means algorithm with trigonometric encoding is shown in

Fig. 5(b). It can be clearly seen that the points near the center of the image are frequently misclassified. The result is reminiscent of the stripe structure seen in Fig. 2(b). Repeated runs with the above parameters produce similar results.

The Hybrid K-means algorithm clustered the points as depicted in Fig. 5(c). The clusters are quite clear in the original data and it is immediately seen that the Hybrid K-means algorithm does not suffer the problems of the K-means algorithm with trigonometric encoding.

6.2. *Numerical studies*

The proposed algorithm has been compared to K-means clustering with trigonometric encoding using synthetic data generated similarly to the case study above. To asses the merit of using the Hybrid K-means, experiments have been performed for all combinations of the following parameters: dimension $M \in \{2, 4, 6, 8\}$, number of clusters $K \in \{2, 3, ..., 6\}$, and standard deviation of patterns around cluster prototypes $\sigma \in \{0.8, 1.2, ..., 2.4\}$. The standard deviations were chosen so that the clusters did not overlap excessively as this only confounds the results because no method can separate overlapping points. In each problem, the patterns were generated in the space $\langle 0, 10)^M$. Three scenarios were considered: in the first one, only one attribute was periodic, in the second one half of the attributes were periodic and in the third one, all of the attributes were periodic. The period of all periodic attributes was set to $D = 10$.

For each combination of the above parameters, 100 different realizations of the clusters were created. First, the required number of cluster prototypes were drawn from a uniform distribution. If the cluster centers were too close, the a new set of cluster centers was drawn. The minimum distance between any pair of cluster centers was selected to be 3 to ensure that clusters with $\sigma = 2.4$ did not overlap excessively. Each cluster was then populated with 500 points distributed normally around the cluster prototype with the given standard deviation. The clustering algorithms were initialized with the Forgy approach[4] and run 5 times. For each trial, the standard clustering criterion

$$J = \frac{1}{2} \sum_{i=1}^{N} \rho_{\mathcal{I},\mathcal{D}}(x_i, w_{s_w(i)})^2, \tag{25}$$

summing the squares of distances of all samples to their assigned prototypes was computed and the trial with the smallest criterion was used. This is standard practice when applying K-means with random initialization. Performance of the algorithms has been evaluated by computing the number of erroneously assigned points according to standard clustering method performance evaluation criteria.

The deformation caused by the trigonometric encoding severely affects the performance of the K-means algorithm in many configurations. The histograms clearly show that for mixed attribute sets (periodic and non-periodic), the advantage of Hybrid K-means over K-means with trigonometric encoding is much larger than when all the attributes are periodic. This stems from the fact that there are two distinct deformation effects caused by the trigonometric encoding as described in Sec. 3.2. If all of the attributes are periodic, then the deformation in the expectation step is the same for all attributes and its negative impact is much smaller. The use of the cosine metric itself does not cause large performance degradation, this is obvious by examining the range of differences of misclassification counts in histogram 6(c). However when a mixed attribute set is provided as input, the distortion effect caused by the expectation step is also present as a side effect of using trigonometric encoding and 'fitting' periodic attributes into the standard K-means framework. Mixed attribute sets occur quite often and constitute a much more important practical case.

Plots in Fig. 7 show the dependencies of misclassification rates on the number of clusters and on their size for one periodic attribute and for all attributes periodic. The plots confirm that when all attributes are periodic, the difference in performance is very small compared to the setup with mixed attributes.

Summarizing the performance tests, there are some configurations where the algorithms perform comparably and many configurations where the proposed algorithm is significantly better than K-means with trigonometric encoding.

6.3. *Application to Real Data*

To demonstrate the utility of the proposed Hybrid K-means algorithm, it has been applied to a real data set describing consumption of natural gas as a function of calendar days. It contains 1153 data points collected over the period of about three years between 2003 and 2005. The clustering algorithm has been used to separate the data into several segments with the goal of building a simple regression model for each segment. The distribution of data, shown in Fig. 8(a), lends itself to separation into four clusters corresponding to the heating season (upper data points corresponding to high gas consumption at the beginning and end of a calendar year), the non–heating season (lower data points corresponding to the low gas consumption during summer), and two transient periods (two slanted data segments, one with negative and one with positive slope).

One of the attributes, the calendar day, is periodic and thus makes this data set suitable for application of the Hybrid K-means algorithm. By visual inspection, it is obvious that the data points at the beginning and at the end of the year should be grouped together into a cluster corresponding to the heating season. The standard K–means algorithm completely fails to achieve such grouping, as shown in Fig. 8(b). Both remaining algorithms, K–means with trigonometric encoding and Hybrid K-means, group the beginning and the end of the heating season together. Each of the two algorithms does this, however, with a different quality of separation between the heating/non–heating season and the transient periods. In particular, the results obtained by Hybrid K-means provide much better vertical separation of the clusters (with respect to the values of gas consumption). The greater vertical overlap of the neighboring clusters obtained by K–means with trigonometric encoding is caused by the deformation of the feature space described in Section 3.2.

To quantify the effect of different partitions obtained by application of the three variants of K-means, a simple regression model has been built for each cluster to predict the value of consumption from the value of day index. The model has the following form

$$\text{Consumption} = m \cdot \text{Day} + b. \tag{26}$$

Mean absolute percentage errors (MAPE) of predictions obtained by applying the regression models on the data set used for clustering are summarized in Table 1. The results clearly show the superiority of the proposed Hybrid K-means algorithm. The regression models based on partition obtained by this algorithm reduce MAPE by 1.42% with respect to the standard K–means algorithm, and by 1.06% compared to K–means with trigonometric encoding. The overall high values of MAPE are caused by the simplicity of the prediction model, considering only day index as the driver of gas consumption. More sophisticated forecasting systems include values of temperature and past consumption to build a non–linear autoregressive model and achieve MAPE around 3.6%[10].

Algorithm	Standard	Trigonometric	Hybrid
MAPE	13.13%	12.87%	11.81%

Table 1. Mean absolute percentage error (MAPE) obtained by applying the regression models to clusters from Fig. 8.

7. Conclusions and Future Work

In this paper, we have examined the clustering problem with periodic attributes. It has been shown that the standard K-means algorithm combined with trigonometric encoding suffers from some drawbacks. Trigonometric encoding distorts spatial relationships in the pattern space, which affects both the expectation step and the maximization step of K-means. The resulting partition is burdened with a systematic error caused by the fact that clusters tend to form in regions where trigonometric encoding contracts the feature space most. In addition, dimensionality of the feature space is increased when using this approach.

To alleviate these problems, a solution of the minimization problem using an alternate metric has been shown. This metric correctly represents distances for periodic attributes and its use does not entail increasing the dimensionality of the feature space. Based on this algorithm, a novel expectation/maximization step for the K-means algorithm has been derived. Subsequently, a modified K-means clustering algorithm, Hybrid K-means, for problems containing both periodic attributes and non-periodic attributes has been developed.

To explore the behavior of the Hybrid K-means algorithm on synthetic data, a large number of numerical simulations has been performed. The simulations show that the Hybrid K-means algorithm compares favorably with K-means with trigonometric encoding. The statistics of the results have clearly shown that the dominant cause of performance degradation is in the expectation step of K-means with trigonometric encoding. As a consequence, the performance boost of the proposed algorithm over the K-means algorithm with trigonometric encoding is largest in mixed attribute datasets with only a small number of periodic attributes. Such datasets are most frequently encountered in practice as opposed to sets with only periodic attributes or many periodic attributes. As the number of periodic attributes increases, the distortion of the feature space caused by the trigonometric mapping itself has a smaller impact on the quality of the clustering.

The proposed algorithm has been compared to K-means with trigonometric encoding and to the standard K-means algorithm on a gas load data set. To quantify the quality of separation of the dataset into heating, non-heating and two transient periods, simple regression models have been built in each time period identified by each of the three variants of the K-means algorithm. The linear regression models based on clusters generated by the Hybrid K-means algorithm show the smallest mean absolute prediction error.

The algorithms described in this paper will be applied to several practical problems, such as time series prediction and processing of complex biological signals.

Acknowledgment

This work was supported in part by the 6RP EU project BRACCIA (Contract No 517133 NEST) and the Institutional Research Plan AV0Z10300504, by the Grant Agency of the Academy of Science of the Czech Republic (grant No. 1ET400300513), and by the Natural Sciences and Engineering Research Council (NSERC) of Canada.

Appendix A. Optimal Maximization Step Pseudocode

```
Input  $X = \{x_1, x_2, ..., x_{N_j}\}, D > 0$
Output  $w_k$
Init
  $Y = \mathrm{sort}(X)$
  $Z = \{i \,|\, y_i > D/2\}$, if $Z = \emptyset$ then $m = N_k + 1$ else $m = \min_{y_i \in Z} i$
  $sum = \sum_{i=1}^{m-1} y_i + \sum_{i=m}^{N_k} (y_i - D)$
  $w_k = sum/N_j$
```

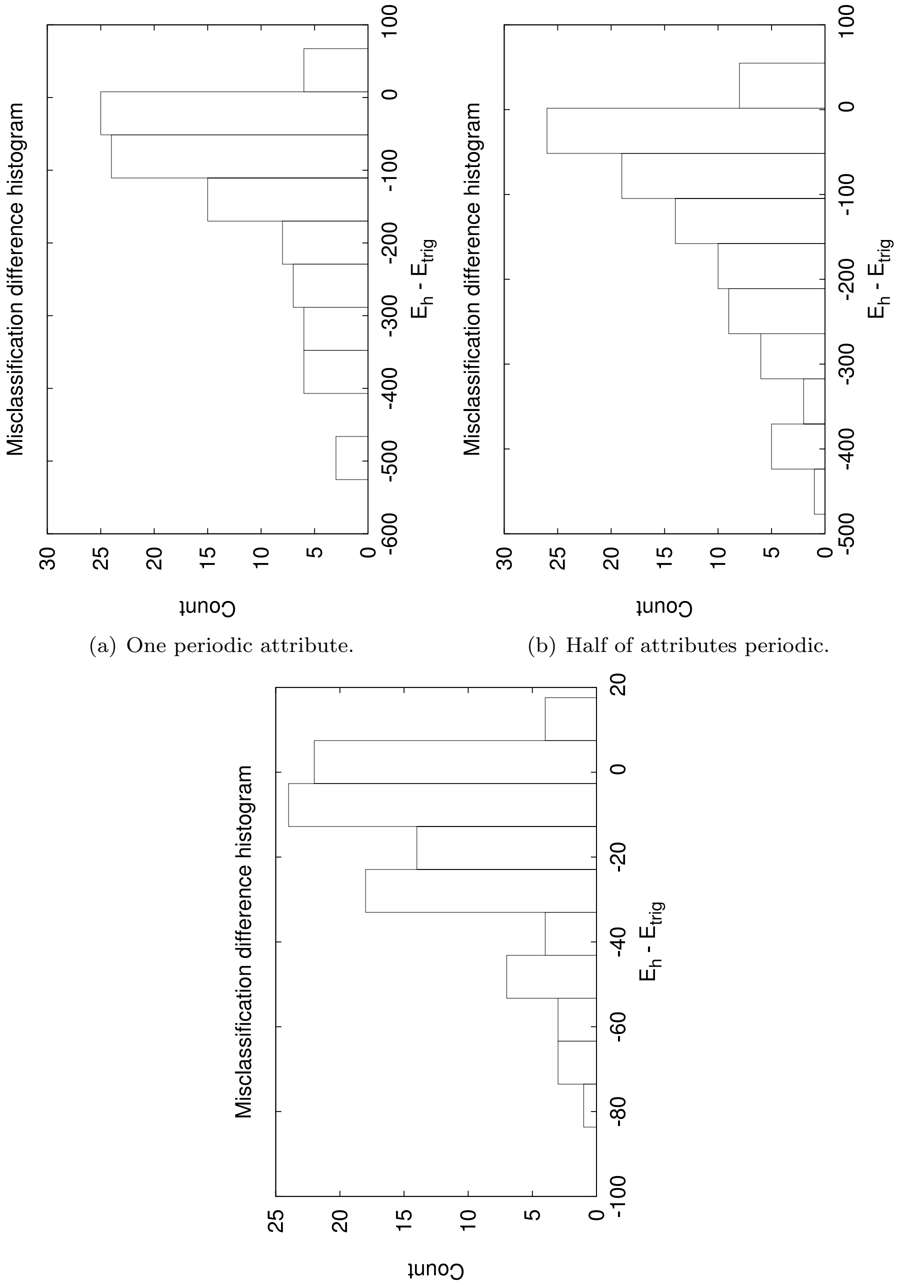
```
$c = \sum_{i=1}^{m-1} (w_k - y_i)^2 + \sum_{i=m}^{N_k} (w_k - y_i + D)^2$
$c^* = c, w_k^* = w_k$

Loop I
  for i=$m : N_k$ ;the unwrapping phase
  $sum = sum + D, w'_k = sum/N_k$
  $c = c + (w'_k - y_i)^2 - (w_k - y_i + D)^2 + (N_k - 1)(w'^2_k - w^2) - 2(w'_k - w_k)(s - y_i)$
  $w_k = w'_k$
  if $c < c^*$ then $c^* = c, w_k^* = w_k$
Loop II
  for i=$1 : m - 1$ ;the wrapping phase
  $sum = sum + D, w'_k = sum/N_k$
  $c = c + (w'_k - y_i - D)^2 - (w_k - y_i)^2 + (N_k - 1)(w'^2_k - w^2) - 2(w'_k - w_k)(s - y_i - D)$
  $w_k = w'_k$
  if $c < c^*$ then $c^* = c, w_k^* = w_k$
Termination
  return $w_k^*$
```

References

1. P. Berkhin, *Survey of clustering data mining techniques*, Tech. rep., Accrue Software, San Jose, California (2002).
2. L. Bottou, Y. Bengio, Convergence properties of the *K*-means algorithms, in: G. Tesauro, D. Touretzky, T. Leen (Eds.), *Advances in Neural Information Processing Systems*, Vol. 7, The MIT Press, 1995, pp. 585–592.
3. M.-C. Su, C.-H. Chou, A modified version of the k-means algorithm with a distance based on cluster symmetry, *IEEE Trans. Pattern Analysis and Machine Intelligence* **23** (6) (2001) 674–680.
4. E. Forgy, Cluster analysis of multivariate data: efficiency vs. interpretability of classifications, *Biometrics* **21**, 1965, pp. 768–769.
5. A. Jain, M. Murty, P. Flynn, Data clustering: A review, *ACM Computing Surveys* 31 (3) (1999) 264–323.
6. T. Kanungo, D. Mount, N. Netanyahu, C. Piatko, R. Silverman, A. Wu, An efficient k-means clustering algorithm: analysis and implementation, *IEEE Trans. Pattern Analysis and Machine Intelligence* **24** (7) (2002) 881–892.
7. J. B. MacQueen, Some methods for classification and analysis of multivariate obser-

vations, in: *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability*, Vol. 1, University of California Press, 1967, pp. 281–297.

8. K. Mardia, P. Jupp, *Directional Statistics*, Wiley, Chichester, 2000.
9. R. Mitchell, Forecasting electricity demand using clustering, *Applied Informatics* **378**, 2003, pp. 378–134.
10. P. Musilek, E. Pelikan, T. Brabec, M. Simunek, Recurrent neural network based gating for natural gas load prediction system, In: *Proc. WCCI 2006, World Congress on Computational Intelligence*, Vancouver, BC, Canada, 2006, pp. 7127–7132.
11. N. H. Viet, J. Mandziuk, Neural and fuzzy neural networks in prediction of natural gas consumption, *Neural, Parallel & Scientific Computations* **13**, 2005, pp. 265–286.
12. R. Xu, D. Wunsch, Survey of clustering algorithms, *IEEE Transactions on Neural Networks* **16** (3), 2005, pp. 645 – 678.

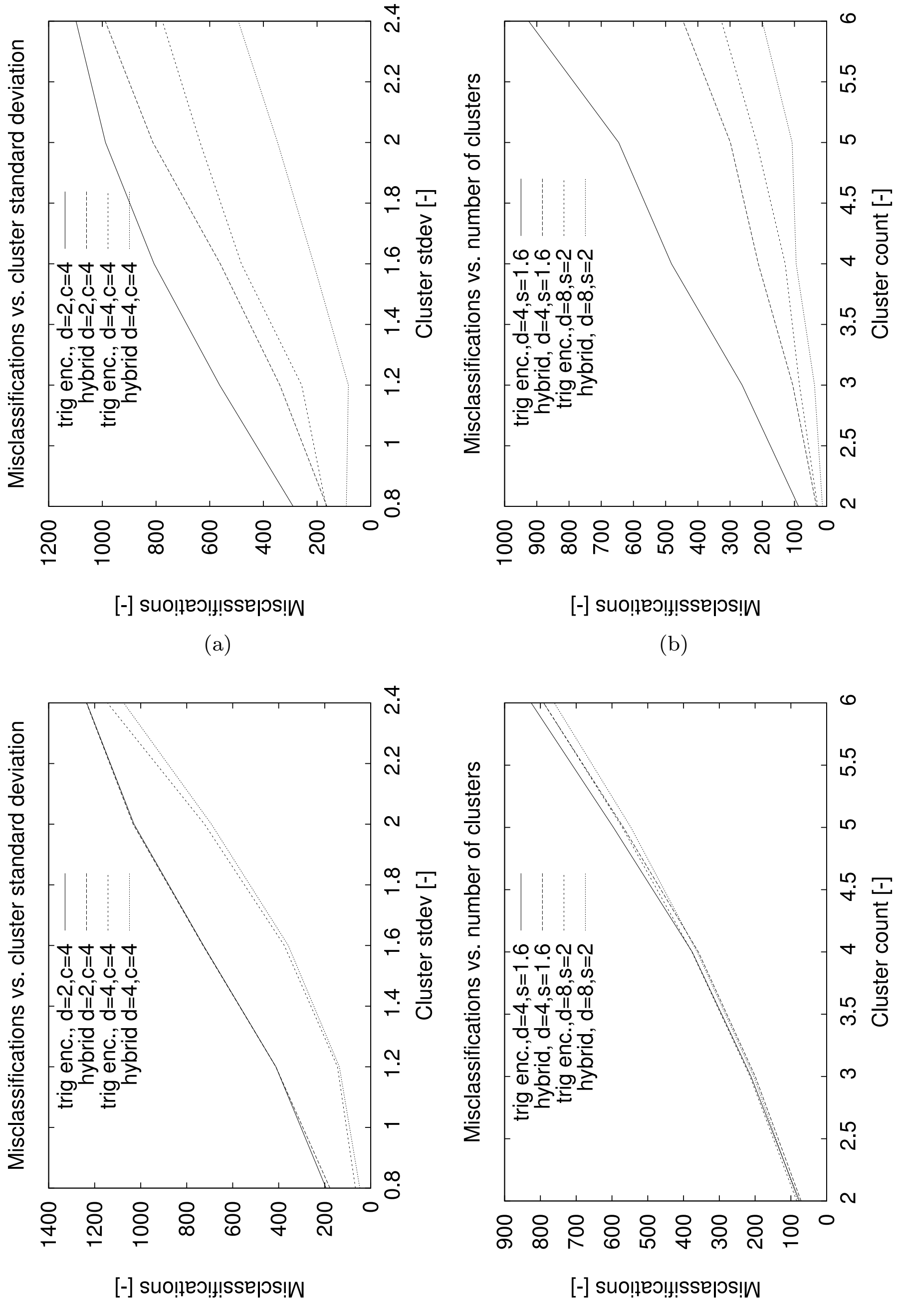


(a) One periodic attribute.

(b) Half of attributes periodic.

(c) All attributes periodic.

Fig. 6. Histograms of differences of misclassification counts of the Hybrid K-means E_h and K-means with trigonometric encoding E_{trig} for different proportions of periodic attributes. Histograms collect misclassification differences over all the tested parameters.



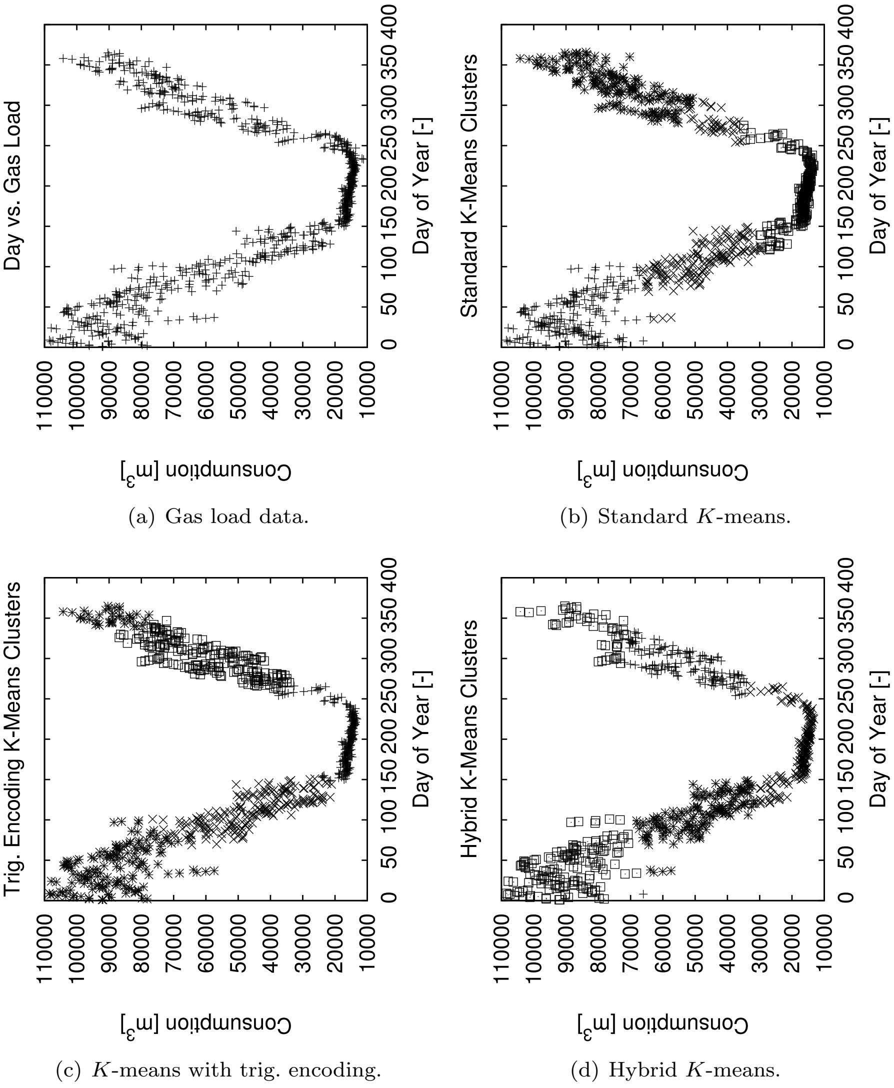
(a)

(b)

(c)

(d)

Fig. 7. Dependencies of the misclassification rate on the cluster standard deviation and on the number of clusters for one periodic attribute 7(a),7(b) and for all attributes periodic 7(c),7(d).



(a) Gas load data.

(b) Standard *K*-means.

(c) *K*-means with trig. encoding.

(d) Hybrid *K*-means.

Fig. 8. Gas load data (a) and clustering obtained using three variants of *K*–means algorithm (b–d)