Unsupervised feature selection in binarization of real attributes for conceptual clustering

Guzel Shkaberina[1,2*], *Igor* Masich[1,2], *Egor* Markushin[2], and *Ekaterina* Kraeva[2]

[1]Laboratory "Hybrid Methods of Modeling and Optimization in Complex Systems", Siberian Federal University, Krasnoyarsk, Russia
[2]Reshetnev Siberian State University of Science and Technology, Krasnoyarsk, Russia

Abstract. This paper proposes an approach for processing noisy data to form homogeneous subgroups of objects based on Formal Concept Analysis (FCA). The approach involves binary encoding of heterogeneous features and unsupervised feature selection using the Laplacian Score. The selected feature set is then used to generate formal concepts. The main idea of our research is to use the concepts derived through FCA as new features for clustering. This process transforms the original feature space into a concept-driven space, where each feature corresponds to the extents of the derived concepts. The proposed approach enhances clustering performance in the presence of noise, outperforming the traditional K-means clustering algorithm in terms of cluster coherence and accuracy. By utilizing concept-based features, the method is able to better capture the underlying structure of the data, leading to more robust and meaningful groupings compared to conventional attribute-based clustering techniques.

1 Introduction

Conceptual clustering algorithms differ from traditional clustering methods by not only grouping objects based on their features but also by attempting to uncover and explain the underlying concepts behind these groups. These algorithms focus on understanding the data structure and interpreting clusters through rules or concepts that can be easily explained. One of the most well-known conceptual clustering algorithms is COBWEB [1], which constructs a hierarchical cluster tree where each node represents a concept (or a set of features) describing the objects in that cluster. COBWEB updates incrementally, incorporating new objects as they arrive and uses a category utility function to assess cluster quality. Its key advantages include suitability for dynamic data and the creation of interpretable cluster structures. However, it may struggle with scalability for large datasets and can be sensitive to the order of data input.

The CLASSIT [2] algorithm is a modified version of COBWEB, adapted for continuous data. Unlike COBWEB, which works with categorical features, CLASSIT applies to numerical data and uses Gaussian distributions to describe clusters. Like COBWEB, it

* Corresponding author: z_guzel@mail.ru

updates the cluster structure as new objects are added, maintaining a conceptual structure for numerical data. However, it shares COBWEB's sensitivity to data entry order.

The GALOIS algorithm [3], based on lattice theory, is designed to find conceptual relationships between objects and their attributes. This approach constructs hierarchies of concepts, with each cluster described by a set of shared properties. GALOIS builds a concept hierarchy based on the shared use of features and represents objects and their attributes using conceptual lattices. While particularly useful for analyzing data with clear concepts and attributes, it can be computationally expensive for large datasets.

Formal Concept Analysis (FCA) [3] is a methodology that uses formal mathematical tools to identify concept hierarchies in binary data. FCA constructs a concept lattice where each node represents a concept, combining a set of objects and their common attributes. It is well-suited for semantic data analysis and creating a clear and interpretable hierarchy of concepts. However, it is limited to binary features and can struggle to scale for very large datasets.

A concept can be viewed as a Boolean function that assigns a true or false (1 or 0) value to each object in the dataset. We propose using these concepts as additional features for clustering. Since generating concepts requires binary features, and the number of resulting binary features can be large, we suggest employing the Laplacian Score [LS, LS1] as an unsupervised feature selection method.

2 Feature generation based on Formal Concept Analysis

Conceptual clustering can be formally defined as follows [1]: Let $O = \{o_1, o_2, \dots, o_n\}$ represent a set of objects, each described by a set of attributes $A = \{a_1, a_2, \dots, a_m\}$, which can be either quantitative or qualitative, with values drawn from $V = \{V_{11}, \dots, V_{nm}\}$. The goal of conceptual clustering is to partition the set O into a collection of clusters $C = \{c_1, c_2, \dots, c_k\}$, ensuring the following conditions are met:

1. Each cluster c_i (i = 1,...,k) is associated with both an extensional and an intentional description. The extensional description lists the objects in the cluster, while the intentional description contains the concepts (statistical or logical properties) that describe the objects within the cluster in terms of the attribute set A. The interpretability of these concepts for the user depends on the complexity of the language used to express them.

2. For every object $o_i \in O, i = 1, \dots, n,$ there exists a cluster $c_j \in c, j = 1, \dots, k$ such that $o_i \in c_j$. This ensures that every object in the set belongs to at least one cluster.

3. There is no empty cluster in the final partition, i.e., $\nexists c_j \in c, j = 1, \dots, k$ such that $c_j = \emptyset$. This ensures that each cluster contains at least one object from the dataset.

Formal Concept Analysis (FCA) [3] is a conceptual modeling approach that examines how objects can be hierarchically grouped based on their shared attributes. This process is essentially a form of biclustering, maintaining the object-attribute relationship to capture similarities within clusters. The result of FCA is a concept lattice (also known as a Galois lattice), which consists of hierarchically organized formal concepts, each representing a bicluster [3].

In FCA, the In-Close family of algorithms [4] is regarded as the most advanced. These algorithms are all derived from the Close-by-One (CbO) algorithm, originally proposed by [5].

Algorithm 1. Close-by-One algorithm (CbO)

CbO(A, B, y):
Input: A—extent, B—intent, y—last added attribute
print(<A,B>)
for $i \leftarrow y+1$ to n do
  if $i \notin B$ then

$C \leftarrow A \cap i$
D ←C
if $D_i = B_i$ then
CbO(C, D, i)
end for

The following idea is proposed in our study.

1. Normalize the initial real dataset $O = \{o_1, o_2, \dots, o_n\}$ (Figure 1, Step 1). Normalizing data is a fundamental step in the preprocessing phase of multidimensional statistical analysis, particularly when dealing with variables measured in different units. Various techniques exist for data normalization [6]. In this study, we adopt the following method to perform the transformation:

$$x\_scaled_{ij} = \frac{x_{ij} - min}{max - min}, \quad (1)$$

where x_scaled_{ij} is the normalized value, x_{ij} is initial real value, *min* is the minimum value of the feature, *max* is the maximum value of the feature. Applying formula (1) to each data point and feature scales the values to a range between 0 and 1.

2. Then we transform the real features to binary form (Figure 1, Step 2).

All possible values of each real feature are sorted in ascending order:

$$o_{i_1 j} \le o_{i_2 j} \le \cdots \le o_{i_n j}. \quad (2)$$

Cut points $t_s^j, s = 1, \dots, k, j = 1, \dots, m$ (where m is the number of features, k is the number of cut points) are determined using frequency discretization, meaning they are positioned so that each interval (bin) contains approximately the same number of samples. For each cut point, a new binary feature o_binary_{ij} is created, based on whether the original value exceeds the cut point [7]:

$$\text{o}\_binary_{ij}^{s} = \begin{cases} 1, & o_{ij} \ge t_s^j, \\ 0, & o_{ij} < t_s^j, \end{cases} \quad (3)$$

where i is the index of the object, $i = 1, \dots, n$, and j is the index of the feature, $j = 1, \dots, m$. Thus, for each real feature describing the object, we have an ordered set of potential cut points $t_1^j, t_2^j, \dots, t_k^j$.

3. In the next step, we select significant features using the Laplacian Score [8, 9] and rank them in ascending order (Figure 1, Step 3). This method is based on the principle that data points from the same class tend to cluster closely together, allowing the importance of a feature to be evaluated by its ability to capture this proximity.

The Laplacian Score is a feature selection method that evaluates the significance of each feature based on its ability to preserve the local structure of the data. This approach is rooted in the idea that data points from the same class or similar category tend to cluster closely together in feature space. The Laplacian Score quantifies how well a given feature reflects this proximity, making it a useful tool for identifying the most informative features for tasks such as classification or clustering.

The key idea is that features that preserve the local structure of the data – meaning they ensure that similar data points remain close in the feature space – are more likely to be useful for distinguishing between different classes or clusters. The Laplacian Score for each feature is calculated by measuring how much the feature disrupts or preserves the proximity of

similar points as encoded by the graph. Features with lower Laplacian Scores are considered more significant because they better maintain the natural structure of the data.

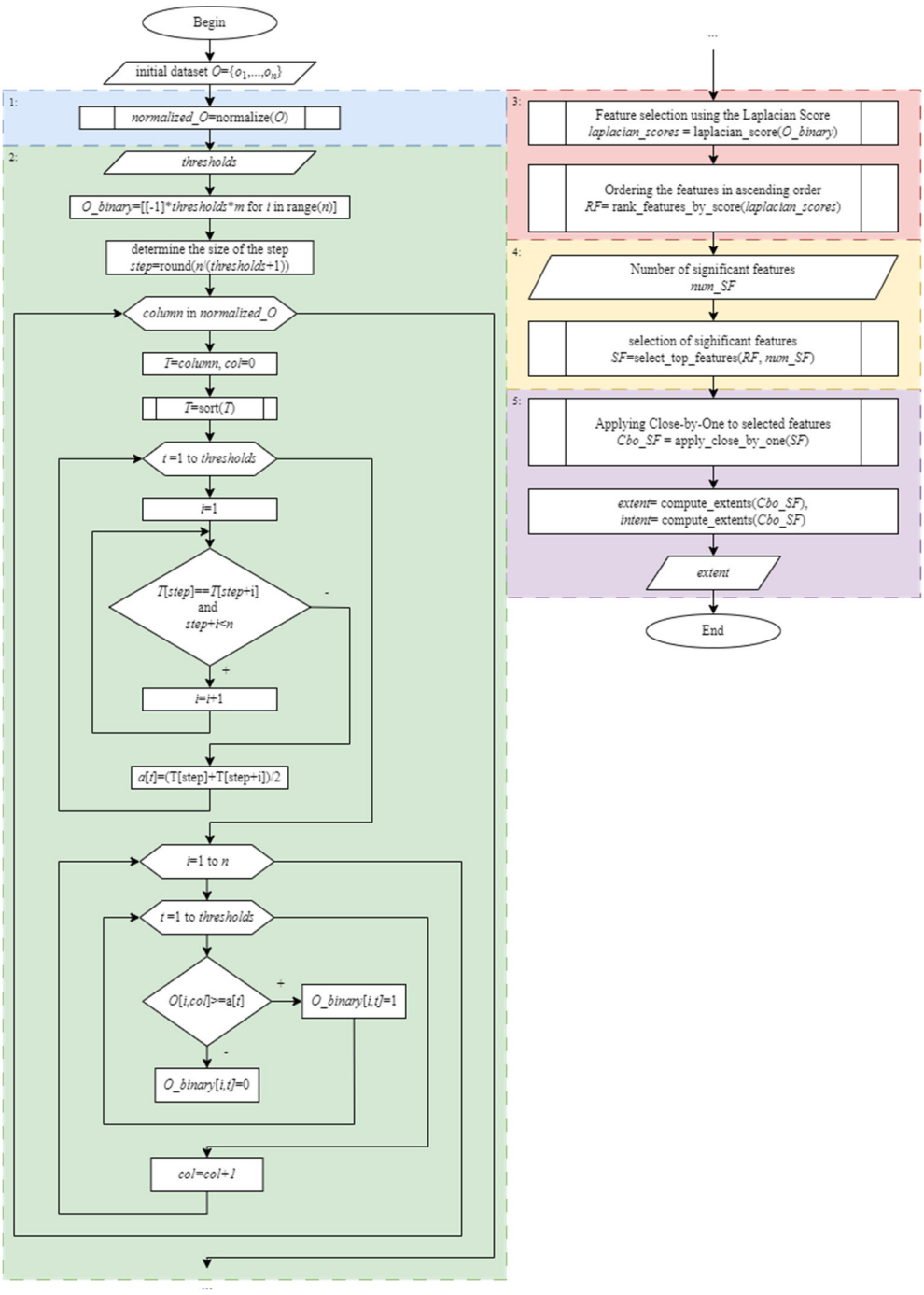

Fig. 1. Feature generation based on the CbO

After computing the Laplacian Score for each feature, the features are ranked in ascending order based on their scores, with the most important features having the lowest scores. This ranking allows for the selection of the most informative features, which can then be used in subsequent steps of analysis, such as clustering or classification. By focusing on features that capture local proximity, the Laplacian Score helps improve the performance of machine learning models by reducing noise and enhancing the interpretability of the selected features.

4. The expert determines the number of top-significant features (Figure 1, Step 4). The number of features used in further analysis is determined based on expert judgment, carefully balancing the trade-off between increasing dimensionality and minimizing information loss. The features are selected according to the ranking obtained from the Laplacian Score in the previous step.

5. To the resulting set of top-significant features, we apply Close-by-One algorithm (CbO) (Figure 1, Step 5). The extents obtained are the input features for the clustering problem.

The core idea of our research is to improve the clustering process by utilizing concepts derived from FCA as new features. Initially, each object in the dataset is described by a set of attributes, forming a multidimensional feature space. FCA is employed to discover structured relationships between objects and their attributes, resulting in the identification of formal concepts. Each concept consists of an extent (the set of objects that share a common set of attributes) and an intent (the set of attributes shared by those objects).

Our approach involves transforming the original feature space into a new feature space based on the extents of the formal concepts derived from FCA. In this new space, each feature corresponds to the extent of a specific concept, indicating whether an object belongs to that concept. For each object, a binary value (1 or 0) is assigned to show whether the object is part of a given concept's extent. This replaces the original attributes with a set of binary features representing membership in various concepts.

This transformation shifts the focus from individual attributes to higher-level conceptual groupings of objects. The new features, based on formal concept extents, capture relationships between objects in terms of shared properties, which may be more informative for clustering than the raw data alone. Once this transformation is complete, traditional clustering algorithms (e.g., k-means, hierarchical clustering) can be applied in this new concept space. The aim is that these concept-based features will provide a more meaningful structure for clustering, as they reflect deeper relationships between objects based on shared attributes rather than individual raw features.

The advantages of this approach include improved interpretability of the resulting clusters, since they are based on formal concepts that describe meaningful groupings of objects. Additionally, the concept-based features may better capture latent relationships in the data, leading to more coherent and insightful clusters.

3 Computational experiments

For the experiments, we analyzed two samples of industrial products: Transistors 2P771A and Single-channel transistor optocouplers 3OT122A.

The Transistors 2P771A dataset consists of 182 transistors characterized by 12 real-valued features, representing the results of electrical effect tests. This dataset includes a mixture of two homogeneous product groups.

The Single-channel transistor optocouplers 3OT122A dataset comprises 278 optocouplers, described by 14 real-valued features, also representing the results of electrical effect tests. Similar to the first dataset, it contains a mixture of two homogeneous product groups.

In the experiments, we utilized various distance measures: Euclidean distance (EuD), Manhattan distance (ManD), Squared Euclidean distance (SEuD), Cosine distance (CosD), and Chebyshev distance (ChD). Each experiment was repeated 30 times, and the best (maximum) Rand Index value is reported in Tables 1 and 2.

The algorithm was implemented in Python. The following test system was used for computational experiments: 11th Gen Intel(R) Core (TM) i7-1165G7 2.80GHz CPU, 16 GB RAM.

A computational experiment was carried out using two types of datasets: the initial dataset with real-valued (normalized) features and a dataset consisting of extents obtained by the Concept Formation algorithm (FCA), CloseByOne: binar_2_extent, binar_3_extent, binar_4_extent, binar_5_extent, binar_6_extent, binar_7_extent, binar_8_extent. The experiment was carried out using the k-means algorithm. In this study the initial solution is chosen randomly.

The computational experiment demonstrated that, in most cases, the use of extents in the k-means algorithm [10, 11] enhances the Rand Index [12] for the separation of homogeneous product groups (Tables 1 and 2).

Table 1. Rand Index for transistors 2P771A

dataset	EuD	ManD	SEuD	CosD	ChD
initial dataset (normalized)	**0.876**	**0.896**	**0.876**	0.770	0.848
binar_2_extent	0.830	0.688	0.830	0.886	0.830
binar_3_extent	0.839	0.839	0.839	**0.967**	0.936
binar_4_extent	0.812	0.812	0.812	0.876	0.848
binar_5_extent	0.795	0.786	0.795	0.915	0.946
binar_6_extent	0.754	0.746	0.754	0.915	0.857
binar_7_extent	0.731	0.731	0.731	0.896	**0.957**
binar_8_extent	0.786	0.786	0.786	0.896	0.896

For initial dataset, the Rand Index has maximum values for Euclidean distance (EuD), Manhattan distance (ManD), Square Euclidean distance (SEuD). For binar_7_extent dataset, the Rand Index has maximum values for Chebyshev distance (ChD). Also, For binar_3_extent dataset, the Rand Index has maximum values for Cosine distance (CosD).

Table 2. Rand Index for Single-channel transistor optocouplers 3OT122A

dataset	EuD	ManD	SEuD	CosD	ChD
initial dataset (normalized)	0.499	0.505	0.499	0.499	0.512
binar_2_extent	0.508	**0.546**	0.508	0.630	0.570
binar_3_extent	0.519	0.517	0.519	0.621	0.697
binar_4_extent	0.531	0.528	**0.531**	0.702	0.565
binar_5_extent	**0.529**	0.529	0.529	0.686	**0.712**
binar_6_extent	0.514	0.506	0.514	**0.704**	0.697
binar_7_extent	0.531	0.531	**0.531**	0.531	0.697
binar_8_extent	0.514	0.514	0.514	**0.704**	0.700

For the binar_2_extent dataset, the Rand Index reaches its highest values with the Manhattan distance (ManD). For the binar_4_extent and binar_7_extent datasets, the highest Rand Index values are observed with the Squared Euclidean distance (SEuD). For the binar_5_extent dataset, the Rand Index achieves maximum values with Euclidean distance (EuD), and Chebyshev distance (ChD). For the binar_6_extent and binar_8_extent datasets, the highest Rand Index values are observed with the Cosine distance (CosD).

4 Conclusion

In summary, our methodology transforms the original attribute space into a concept-driven feature space where formal concepts provide a higher-level abstraction. This transformation can enhance the effectiveness of clustering, allowing us to group objects based on shared conceptual properties and potentially yielding more interpretable and meaningful clustering results.

The computational experiment revealed that, in the majority of cases, incorporating extents derived from Formal Concept Analysis into the K-means algorithm improves the Rand Index, which measures the accuracy of clustering. This enhancement leads to better differentiation of homogeneous product groups, suggesting that using concept-based features helps achieve more precise and meaningful cluster separation compared to traditional attribute-based clustering methods. The results highlight the potential of formal concepts to refine clustering performance.

This work was supported by the Ministry of Science and Higher Education of the Russian Federation (Grant No. 075-15-2022-1121).

References

1. D.H. Fisher, Mach. Learn., **2,** 139–172 (1987)
2. J. Gennari, P. Langley, D. Fisher, Artificial Intelligence, **40,** 11-61, (1989)
3. B. Ganter, R. Wille, *Formal concept analysis* In Mathematical Foundations (Springer, Berlin, Heidelberg, Germany, 1999)
4. J. Konecny, P. Krajča, Inform. Sciences, **575,** 265–288 (2021)
5. S.O. Kuznetsov, Scientific and Technical Inf. Series 2 Inf. Processes and Systems, **1,** 17–20 (1993)
6. D. Singh, B. Singh, Appl. Soft Comput., **97,** 105524 (2020)
7. I. Masich, N. Rezova, G. Shkaberina, S. Mironov, M. Bartosh, L. Kazakovtsev, Algorithms, **16,** 246 (2023)
8. X. He, D. Cai, P. Niyogi, Adv. Neur. In., **18,** 2830 (2005)
9. M. Belkin, P. Niyogi, Adv. Neur. In., **14,** 585-591 (2001)
10. Y. Li, H. Wu, A Clustering Method Based on K-Means Algorithm, Physcs Proc., **25,** 1104-1109 (2012)
11. S.P. Lloyd, Least Squares Quantization in PCM, IEEE T. Inform. Theory, **28,** 129-137 (1982)
12. W.M. Rand, Objective Criteria for the Evaluation of Clustering Methods, J. Am. Stat. Assoc., **66,** 846-850 (1971)