A Subspace Clustering Extension for the KNIME Data Mining Framework

Stephan Günemann  Hardy Kremer  Richard Musiol  Roman Haag  Thomas Seidl
RWTH Aachen University, Germany
{guennemann, kremer, musiol, haag, seidl}@cs.rwth-aachen.de

Abstract—Analyzing databases with many attributes per object is a recent challenge. For these high dimensional data it is known that traditional clustering algorithms fail to detect meaningful patterns. As a solution subspace clustering techniques were introduced. They analyze arbitrary subspace projections of the data to detect clustering structures.

In this demonstration, we introduce the first subspace clustering extension for the well-established KNIME data mining framework. While KNIME offers a variety of data mining functionalities, subspace clustering is missing so far. Our novel extension provides a multitude of algorithms, data generators, evaluation measures, and visualization techniques specifically designed for subspace clustering. It deeply integrates into the KNIME framework allowing a flexible combination of the existing KNIME features with the novel subspace components. The extension is available on our website.

I. INTRODUCTION

Clustering is a mining task for automatic grouping of similar objects while separating dissimilar ones. For today’s high dimensional data, however, traditional full-space clustering approaches fail to detect meaningful patterns since irrelevant dimensions obfuscate the clustering structure [5]. Using global dimensionality reduction techniques as principal components analysis is not sufficient to solve this problem: by definition, all objects are projected to the same lower dimensional subspace. However, as Figure 1 illustrates, each cluster might have locally relevant dimensions and objects can be part of multiple clusters in different subspaces. These effects cannot be captured by global dimensionality reduction approaches.

Subspace clustering techniques were introduced, aiming at detecting locally relevant dimensions per cluster [10], [15]. They analyze arbitrary subspace projections of the data to detect the hidden clusters. Typical applications for subspace clustering include gene expression analysis, customer profiling, and sensor network analysis. In each of these scenarios, subsets of the objects (e.g., genes) are similar regarding subsets of the dimensions (e.g., different experimental conditions).

Data mining functionality is provided to the end user by data mining frameworks as the Waikato Environment for Knowledge Analysis (WEKA) [8] or the Konstanz Information Miner (KNIME) [4]. These frameworks are easy to use, extensible, and they cover the whole KDD process by offering a large scale of different data mining algorithms, data transformations, and visualization tools.

However, until now, subspace clustering capabilities are only available in the WEKA framework, provided by the OpenSubspace [14] extension.

In this paper, we present our novel subspace clustering extension for KNIME. KNIME is a recent opensource data mining framework which offers several benefits and is widely been used in industry as well as academia. It has a modern, user-friendly interface which allows to model data mining workflows in an intuitive manner. In KNIME, a workflow is defined by a set of nodes, which can represent data sources and sinks, mining algorithms, transformations, visualizations, and further concepts. Each node has specific input and output ports depending on the node’s functionality. The user establishes a new workflow by selecting a set of nodes from the node repository and then connects the corresponding input and output ports to steer the data flow between these nodes. Data mining workflows can be stored for later re-use, modification, or extension.

A major benefit of KNIME is the easy-to-use plugin concept. It allows KNIME to be extended by new features, represented as new nodes in the node repository. These new nodes can freely interact with the existing KNIME components, achieving a deep integration of the extension.

In the following, we will present the first subspace clustering extension for KNIME which offers

- seamless integration into KNIME, e.g. existing transformation and visualization nodes can be flexibly used in a subspace clustering workflow.
- more than 10 different subspace clustering algorithms.
- generators for synthetic data containing hidden subspace clusters.
- measures for evaluation of subspace clustering results.
- specific subspace clustering visualizations.
II. KNIME SUBSPACE CLUSTERING EXTENSION

In this section, we introduce our subspace clustering extension for the KNIME data mining framework. Our tool is seamlessly integrated into KNIME. Thus, all techniques already integrated in KNIME can be combined with our extension for mutual benefit.

Figure 2 shows a screenshot of KNIME and our extension. On the left, the newly developed nodes are illustrated in the node repository. On the right, descriptions of each node and its corresponding input/output ports are given. In the center, the actual workflow is illustrated. In the following we provide details of our extension and the different types of nodes based on three different workflows.

A. Subspace Clustering Algorithms

Figure 2 shows a simple workflow where a data reader node (‘Node 1’; here: reading data from an ARFF file) is connected with a subspace clustering algorithm node (‘Node 2’; here: Proclus [1]). Thus, by specifying this workflow, the user applies Proclus on a given database.

Each subspace clustering algorithm node has one input and two output ports. The input corresponds to the database to be clustered. The two outputs describe the clustering result. At the first output port, a table is provided which describes the relevant dimensions of each cluster. In Figure 2 this table is illustrated at the bottom (‘Cluster Dimensions’) and shows three subspace clusters found in the Iris dataset [6]. The cluster with ID 2, for example, is located in the dimensions ‘sepallength’, ‘sepalwidth’, and ‘petalwidth’. The second output port provides information which objects belong to the detected clusters (table ‘Cluster Objects’). In the example, the object 149 belongs to cluster 0, while object 17 belongs to cluster 1. Please note that in subspace clustering each object might belong to multiple clusters, i.e. clusters might overlap due to different subspace projections. Thus, the output table corresponds to an \( n \times m \) relation. These two outputs can be forwarded to any other node included in the KNIME framework as we will show later on.

In our extension, the user can select among a multitude of different subspace clustering algorithm nodes. These algorithms include grid based clustering techniques (Clique [2], Doc/FastDoc [17], Mineclus [19], Schism [18]), DBSCAN-based techniques (Fires [9], INSCY [3], Subclu [11]) and optimization based techniques for subspace clustering (Proclus [1], Rescu [13], Statpc [12]). The same set of algorithms is used in our WEKA integration [14].
B. Data Generators & Evaluation Measures

A second workflow is illustrated in Figure 3. It models the task frequently performed in scientific literature: a) generate synthetic data with a given clustering ground truth, b) apply an algorithm on the data, and c) measure whether the detected result matches the ground truth.

![Workflow to evaluate the result of a subspace clustering algorithm](image)

To solve this task with our framework, the user has to select a data generator node ('Node 1'). The node constructs synthetic data where the subspace clustering structure is known, i.e. the ground truth for clustering is given. Therefore, each data generator node has three output ports: First, the generated data. Second, the relevant dimensions of each cluster. Third, the cluster memberships of each object. The last two outputs are of the same format as the outputs of the subspace clustering algorithm nodes as described above. Connecting the first port of the data generator with an algorithm node ('Node 2') allows to cluster the synthetic data.

Finally, to measure the quality of the detected results the user can use an evaluation measure node ('Node 3'; here: CE measure [16]). Such a node has five input ports: four ports are required to specify the two clustering results that should be compared (two ports for each clustering result), and one port for the database. Thus, in the figure, all three output ports of the data generator are connected to the measure as well as the two output ports of the Mineclus node. The output of each evaluation measure is a scalar value. Evaluation measures can also be used to compare the results of two different algorithms. Thus, we do not necessarily have to refer to synthetic data generators where the ground truth is known.

We provide several evaluation measure nodes specifically designed for subspace clustering in our framework. These measure include CE, RNIA, Entropy, F1P, F1R, and E4SC (see [7] for an overview).

C. Visualization

The two outputs of each algorithm node already allow to analyze the detected clustering structure on a basic level. That is, by inspecting the corresponding tables (cf. Fig. 2) the user might get an impression about the relevant dimensions of the clusters and the supporting objects. Though, analyzing these tables might be difficult to gain further knowledge. Thus, for easy interpretations of the clustering results we include different visualizations.

One possible visualization is realized with the workflow depicted in Figure 4. The subspace visualization node generates results as shown in the table on the right. The table represents the original database where each row corresponds to one object. The objects belonging to the same cluster are highlighted with the same color. In the example, three clusters are shown. Additionally, also the relevant dimensions of the clusters are depicted. A dimension is relevant, if and only if there is a colored bar on the right hand side of the number. In the example, the green cluster is located in the subspace of dimension 2 and 4, while the blue cluster is located in all four dimensions. Using this visualization, the user can easily compare the different subspaces of the clusters as well as the attribute values of the clustered objects. Considering for example the green cluster, we see that the attribute values in the first (and irrelevant) dimension are distributed in the broad range of 5.0-6.3, while the second (and relevant) dimension shows a deviation of only 2.0-2.4.

To obtain this visualization, the subspace visualization node requires three inputs: First, the database to be analyzed. Next, the relevant dimensions of each cluster with their corresponding coloring. This coloring is realized by using the Color Manager node provided by the KNIME framework. That is, the first output of the subspace clustering algorithm ('Node 2') is firstly forwarded to the Color Manager ('Node 3') before used as an input of the subspace visualization node ('Node 4'). In the Color Manager node, the user can choose the color of each cluster. The last input required for the visualization is the cluster membership information which can be directly transferred from 'Node 2'.

Besides this visualization type our extension allows to use scatterplots colored according to the cluster membership of the objects. Finally, due to our seamless integration into the KNIME framework, all visualization techniques already provided can now be reused to illustrate subspace clustering solutions.

III. Demonstration Scenario

The demonstration setup of our KNIME extensions enables the conference attendees to analyze the subspace clustering structure of real world datasets. We primarily provide data available at the UCI archive [6]. The attendees can design different workflows to test and compare the various algorithms and measures and to verify the soundness of the results. The participants can interactively model examples where the novel subspace clustering capabilities are combined with other data mining functionalities provided by KNIME. The demonstration setup allows to show the high flexibility and easy usability of our extension.
Our subspace extension is publicly available and can be easily integrated into the KNIME framework by simply adding the following update site:

http://dme.rwth-aachen.de/KnimeSCplugin

More details about the installation procedure can be found on http://dme.rwth-aachen.de/KnimeSC

Overall, our subspace clustering extension for KNIME supports the user to apply subspace clustering in various data mining scenarios. Since our extension can be flexibly combined with all KNIME nodes and is seamlessly integrated into the KNIME workflow concept, it allows an intuitive modeling of even complex data mining tasks.

Acknowledgment. This work has been supported by the UMIC Research Centre, RWTH Aachen University, Germany.

REFERENCES


