Geoscience Frontiers 15 (2024) 101823

Contents lists available at ScienceDirect

Geoscience Frontiers

journal homepage: www.elsevier.com/locate/gsf

Research Paper

Using adjacency matrix to explore remarkable associations in big and small mineral data

Xiang Que [a,b], Jingyi Huang [a,c], Jolyon Ralph [d], Jiyin Zhang [a], Anirudh Prabhu [c], Shaunna Morrison [c], Robert Hazen [c], Xiaogang Ma [a,c,*]

[a] Department of Computer Science, University of Idaho, Moscow, ID 83844, USA
[b] College of Computer and Information Sciences, Fujian Agriculture and Forestry University, Fuzhou 350002, China
[c] Earth and Planets Laboratory, Carnegie Institution for Science, Washington, DC 20015, USA
[d] Hudson Institute of Mineralogy, Keswick, VA 22947, USA

ARTICLE INFO

Article history:
Received 31 July 2023
Revised 10 February 2024
Accepted 8 March 2024
Available online 12 March 2024
Handling Editor: Kristoffer Szilas

Keywords:
Adjacency matrix
Association analysis
Data exploration
Mineral informatics
Open data

ABSTRACT

Data exploration, usually the first step in data analysis, is a useful method to tackle challenges caused by big geoscience data. It conducts quick analysis of data, investigates the patterns, and generates/refines research questions to guide advanced statistics and machine learning algorithms. The background of this work is the open mineral data provided by several sources, and the focus is different types of associations in mineral properties and occurrences. Researchers in mineralogy have been applying different techniques for exploring such associations. Although the explored associations can lead to new scientific insights that contribute to crystallography, mineralogy, and geochemistry, the exploration process is often daunting due to the wide range and complexity of factors involved. In this study, our purpose is implementing a visualization tool based on the adjacency matrix for a variety of datasets and testing its utility for quick exploration of association patterns in mineral data. Algorithms, software packages, and use cases have been developed to process a variety of mineral data. The results demonstrate the efficiency of adjacency matrix in real-world usage. All the developed works of this study are open source and open access.

1. Introduction

Data-driven knowledge discovery plays an increasingly effective role in the advancement of geoscience research. Many scientific discoveries have been published recently, which demonstrated the enormous potential of geodata science (Hey et al., 2009; Hazen et al., 2019; Wang et al., 2021; Ma, 2023). As indicated by Hey et al. (2009), the paradigms of science have evolved from empirical, theoretical, and computational approaches to today's data exploration. Their thoughts resonate with the methodology of exploratory data analysis initiated by Tukey (1977), which in our regards is an efficient approach to tackle the challenges occurred in the era of big data. In the field of mineralogy, Xiao and Chen (2012) proposed a fractal projection pursuit classification (FPPC) model to help identify anomalies for mineral exploration. Hazen (2014) discussed the applications of exploratory data analysis and minted a term "abductive analysis" for the general approach. Chen and Xiao (2023) presented a projection pursuit random forest (PPRF) for exploring the deep hidden features in datasets. Very recently, Hazen et al. (2021) and Prabhu et al. (2023) proposed "mineral informatics" as a research field that leverages open data facilities and data science methods to decipher the patterns and trends hidden in datasets of mineralogy, petrology, geochemistry, and many other related disciplines. In mineral informatics, data exploration helps scientists obtain a quick overview of the datasets under study and generate hypotheses for new discoveries. Accordingly, a lot of methods, algorithms, and visualization techniques can be developed and used in mineral data exploration (Yousefi et al., 2021; Zuo et al., 2021; Wang et al., 2022).

Existing studies of mineral informatics have already shown that data exploration can lead to efficient pattern recognition, research hypothesis construction, and exciting scientific discoveries (Ma et al., 2017; Hazen et al., 2019; Hazen and Morrison, 2022). On the other hand, massive volumes of data on the chemical and physical properties, spatial distribution, and evolutionary diversity of minerals are being made available by research communities, such as Mindat (Ralph et al., 2022), RRUFF (Lafuente et al., 2015), and

* Corresponding author at: Department of Computer Science, University of Idaho, Moscow, ID 83844, USA..
E-mail address: max@uidaho.edu (X. Ma).

https://doi.org/10.1016/j.gsf.2024.101823
1674-9871/

the Mineral Evolution Database (MED) (Golden et al., 2019). For instance, the Mindat database has records of over 5,800 mineral species, over 390,000 localities, and over 1,472,000 mineral occurrences by February 2023. Exploration and study of those large mineral data resources have a high value in further extending the field of mineral informatics. Specifically, the associations of minerals hidden behind the data are of great value to many applications: (1) They can provide valuable insights into geological processes, aka, geological understanding, based on understanding how different minerals associated with each other (Sadeghi, 2021). For example, the link between diamonds and chromium-bearing diopside, chromspinel, and magnesian ilmenite may be caused by kimberlite, an igneous rock, which transported diamonds from the mantle to the Earth's surface via volcanic pipes (Field et al., 2008). The occurrence of these minerals together is used as a primary method for diamond exploration and indicates the past geological processes that occurred in that region. (2) Mineral associations can guide exploration and mining by indicating the presence of economically valuable deposits. This is because the discovery of one mineral may lead to the discovery of another, more valuable one (Jowitt et al., 2013). For example, regions with large concentrations of copper minerals, such as chalcopyrite ($CuFeS_2$), may also contain gold deposits. The discovery of the gold deposit was initially as a byproduct of copper mining in the world's largest copper and gold mines, Grasberg mine in Indonesia (Pollard et al., 2005). (3) Understanding the mineral associations can also aid in refining industrial processes, as some minerals can interfere with the processing of others. For example, plants often employ methods to reduce the silica (SiO_2) content before refining in the processing of bauxite to produce alumina. It is because that bauxite ores often also contain various amounts of silica (SiO_2) in the form of quartz or kaolinite, which is not desirable in the alumina production process (Rayzman et al., 2003).

Although the associations among minerals can lead to new scientific insights that contribute to the fields of crystallography, mineralogy, and geochemistry, the process of explorations is often daunting due to the wide range and complexity of factors involved (e.g., chemical composition, geologic structure, temperature, pressure, etc.). Many mineral association studies were based on statistical analysis or association rules, and most of them focused on specific geoscience issues. Keskinen et al. (2022) employed compositional statistical analysis to analyze the association between elements and surface-active minerals of organic matter. Morrison et al. (2023) utilized association rules analysis to predict new mineral occurrences. In recent years, an increasing number of data analysis techniques including cluster diagrams, Klee diagrams, chord diagrams, and network analyses, have been introduced into mineral informatics research for mineral association and some interesting results have been found (Hazen et al., 2019). However, rapid exploration and visualization of undiscovered patterns from extensive open mineral data remains challenging. For instance, the network analysis technique can provide a quantitative visualization framework to explore complex patterns in mineral systems (Morrison et al., 2017). However, network visualization faces challenges, especially when the network is dense, resulting in overlapping nodes and edge crossings. A large number of cross-links in the result may raise a confusing visual pattern (i.e., the "spaghetti" pattern) that masks the topological structure of the network. Moreover, it is difficult to compare two or more random-layout networks given the densely overlayed nodes and edges in them. Compared with network analysis, the adjacency matrix is a complementary and sometimes more effective technique for visualizing networks, especially when dense (Fekete, 2009; Okoe et al., 2019). This representation allows fast navigation among many records and is more readable for specific tasks, such as those in mineral informatics.

This paper aims to present our work of software development and data exploration on the use of adjacency matrix to study the associations in mineral data. We have developed workflows in Python and R languages and an R Shiny application to analyze a variety of datasets retrieved from Mindat, RRUFF, and MED. A few representative use cases, such as those on the associations between chemical elements and mineral species as well as the patterns of mineral–mineral co-existence across localities, are used in this paper to illustrate the utility of the adjacency matrix for mineral data exploration. The presented use cases are just a small part of the adjacency matrices generated in our study. Yet, we have made the application, datasets, and source code open access online. Interested researchers are encouraged to analyze patterns in those adjacency matrices, reuse the application in their own work, and adapt and extend the code for other studies. The remainder of the paper is organized as follows: Section 2 presents the general method of adjacency matrix and the workflow of software development in this study. Section 3 illustrates and analyzes the results of several use cases, including both small and large datasets. Section 4 discusses the highlights of the adjacency matrix for mineral data exploration, the limitations, and ideas for future improvement. At the end, Section 5 concludes the paper.

2. Methods and datasets

The major result of our work is an R Shiny application that deploys adjacency matrix to analyze open mineral data. In this section, we present a quick overview of the method of adjacency matrix, the software development process, and the data resources used in the work.

2.1. Adjacency matrix and the relevant concepts

In the fundamentals of graph theory, an adjacency matrix is an $n \times n$ square matrix A that represents a finite graph (Fekete, 2009). For the undirected graph, as shown in Fig. 1a, the adjacency matrix is symmetric, while for the directed graph, such as Fig. 1d, it is often not. Fig. 1b and 1e show the numerical values of adjacency matrices corresponding to the graphs of Fig. 1a and 1d, respectively. In Fig. 1b and 1e, the value of an element in A_{ij} is one when there is an edge connecting node i to node j, and is zero if the edge does not exist (Biggs et al., 1993). Fig. 1c and 1f are the visualizations based on these two adjacency matrices in Fig. 1b and 1e, respectively, where the round nodes of different colors indicate that they belong to different communities (generated by a specified community detection algorithm), and different cell colors indicate different types of connections (edges) between the nodes. The cell's color matches the node's color if the edge is within a community; the color is gray if the edge is between communities, and white if the edge does not exist. This adjacency matrix-based visualization can be easily rearranged according to the nodes' community, name, and some other relevant attributes in the graph, and is an effective way to facilitate data science discoveries. To better understand the rearrangements and pattern analyses of the adjacency matrix, here is a brief list of several concepts. Suppose a graph G consists of a collection V of nodes and a collection edges E, thus $G = (V, E)$. Each edge $e \in E$ join two nodes. If e join $i, j \epsilon V(G)$, i.e., $e =< i, j >$, then node i and j in this case is adjacent. The distance between node i and j, denoted as $d(i, j)$, is the length of a shortest (i, j)-path. For an undirected graph, $d(i, j) = d(j, i)$, but this is usually not the case for a directed graph. The number of edges with a node i is called the "degree" of i, normally denoted as $\delta(i)$. Loops are counted twice (Diestel, 2017). For the whole graph G, the sum of all node degrees is twice the number of edges, that is,

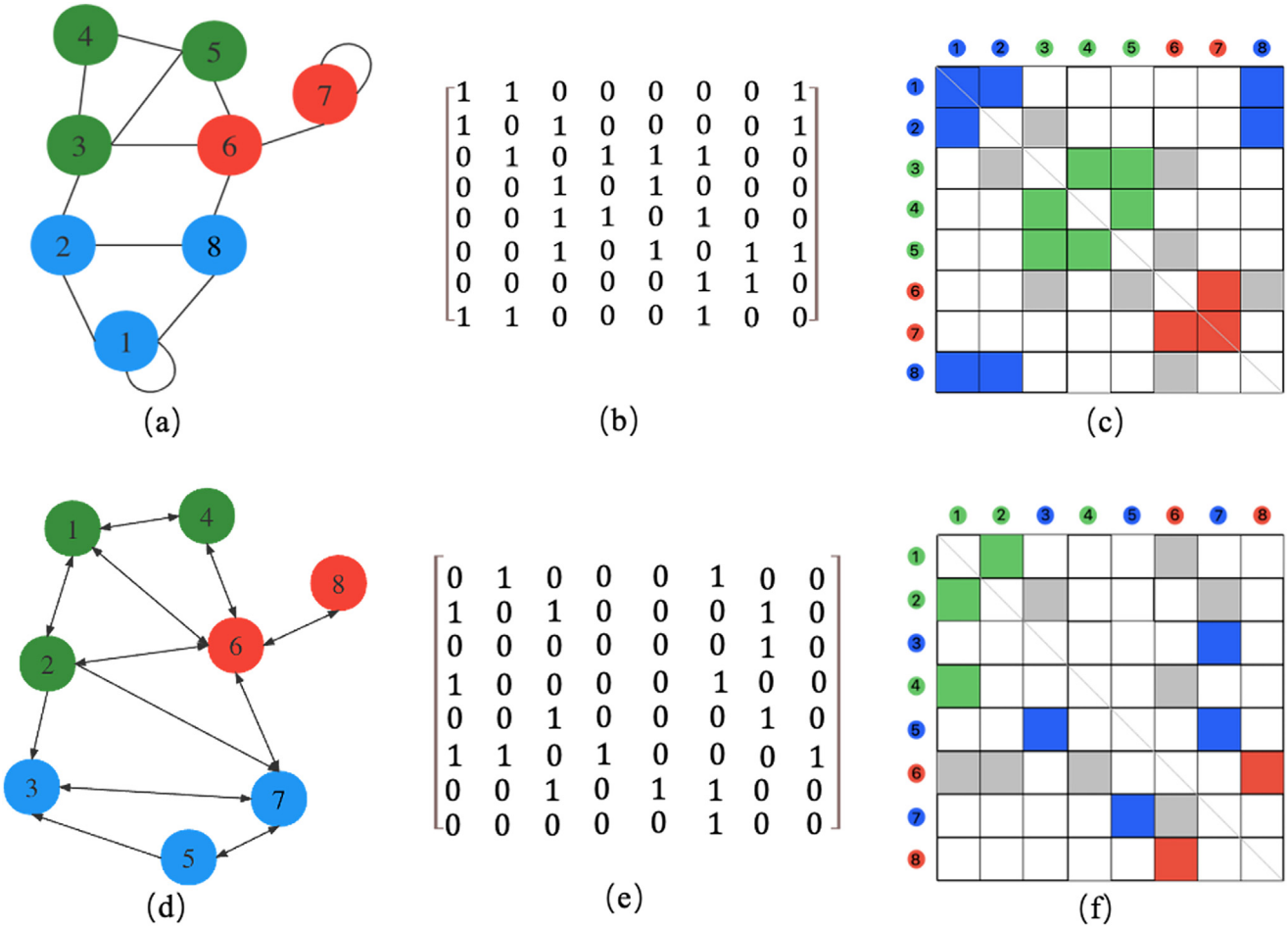


Fig. 1. Adjacency matrix representation of undirected and directed graphs: (a) and (d) undirected and directed graph, respectively; (b) and (e) the numerical values of the adjacency matrices corresponding to (a) and (d), respectively; (c) and (f) the visualizations of the adjacency matrices corresponding to (b) and (e), respectively. The round nodes denote the nodes of graphs, and the cells represent the connections (edges) between them. The different colors of cells and round nodes indicated that they belong to different communities.

$$\sum_{i \in V(G)} \delta(i) = 2|E(G)| \tag{1}$$

where $|E(G)|$ denotes the edge number of graph G. Let us consider Fig. 1a, the degree of node $\delta(1)$, $\delta(2)$, and $\delta(3)$ are 4, 3, and 4, respectively. The sum of the degrees of the nodes in the graph is 26, which is twice the 13 of total edges. The "closeness" $C_c(i)$ of a node $i \in V(G)$ is defined as

$$C_c(i) = 1/\sum_{i,j \in V(G), i \neq j} d(i,j) \tag{2}$$

The lower the closeness $C_c(i)$ is, the more central a node i is to other nodes (Bavelas, 1950). Another concept is "betweenness", which is based on a simple idea: if a node lies on many shortest paths connecting two other nodes, it is an important node. The removal of such a node would directly affect the cost of connective between other nodes (Freeman, 2002). Formally, let $S(i,j)$ be the set of shortest paths between i and j, and $S(i,v,j) \subseteq S(i,j)$ denote the ones that pass through node $v \in V(G)$. The betweenness is defined as

$$C_b(v) = \sum_{i \neq j} \frac{|S(i,v,j)|}{|S(i,j)|} \tag{3}$$

For G is connected, the $|S(i,j)| > 0$ for all distinct i and j. In addition, the eigenvalues of the adjacency matrix characterize the topological structure of the graph (Cvetković et al., 1995; Farkas et al., 2002). For an undirected graph, since its corresponding adjacency matrix is real-valued symmetric, all its eigenvalues are real numbers, and its eigenvectors are mutually orthogonal. Further information about the eigenvalues of adjacency matrices can be found in the spectral graph theory (Brouwer and Haemers, 2012).

2.2. Community detection algorithms based on adjacency matrix

A community in a network is generally defined as a group of nodes that have more and/or stronger connections among themselves than with nodes outside the group. It can be detected through employing specific community detection algorithm, which is a fundamental technique for uncovering the structure within graph (network) with the goal of identifying places where the node connections are tighter or denser than the rest of the network. It's a critical tool for finding natural groupings in graphs based on node connectivity patterns, which helps to understand the underlying structure and dynamics of complex systems, thus providing insights into different domains. Table 1 lists four common community detection algorithms that have been implemented in our R software application.

Table 1
Comparison of four commonly used community detection algorithms.

Name	Principle	Impacts of edge weights
Random Walktrap (Pons and Latapy, 2006)	Based on random walks, this algorithm identifies communities in a network by exploiting the idea that random walks on a graph tend to get "trapped" in densely connected parts (communities).	(1) Edge weights usually represent the strength or intensity of the connections. (2) The paths taken by random walks will be biased towards stronger connections.
Edge betweenness (Girvan and Newman, 2002)	This algorithm detects communities by progressively removing edges with the highest edge betweenness centrality, which is a measure of the number of shortest paths that pass through an edge.	(1) The "shortest" path is defined in terms of the lowest total weight, rather than the fewest edges. (2) If an edge has a high weight (implying a weaker connection), it is less likely to be part of many shortest paths. Conversely, lower-weight edges (stronger connections) might be part of shortest paths and thus have higher betweenness centrality. (3) Communities are more strongly connected internally (with lower-weight edges) and separated by weaker links (higher-weight edges).
Optimal Modularity (Brandes et al., 2007)	This algorithm seeks to maximize the modularity of a network, a measure that quantifies the strength of division of a network into modules or communities.	(1) A strong edge (with a high weight) contributes more to the modularity than a weak edge (with a low weight). (2) Communities are formed such that the sum of the weights of the edges within communities is maximized, relative to what would be expected in a random network. (3) Nodes with heavily weighted edges are more likely to be included within the same community.
Spinglass (Reichardt and Bornholdt, 2006)	Inspired by statistical mechanics, specifically the spin glass model in physics. It treats community detection as an energy minimization problem where each node is a spin that can be in one of several states (communities).	(1) The 'energy' of a configuration (i.e., a particular assignment of nodes to communities) depends on the connections between nodes. Heavier weights on edges can be interpreted as stronger connections or interactions. (2) Edges with higher weights contribute more significantly to the energy calculation. (3) In the process of minimizing the system's energy to find communities, the algorithm inherently gives more importance to heavier (stronger) edges.

2.3. Adjacency matrix construction and data exploration

To quickly explore the association patterns in mineral datasets, a workflow was designed and implemented to generate adjacency matrices on several topics (Fig. 2). First, mineral data were retrieved via several open data resources, including the mineral chemistry records of 72 mineral-forming elements from the Mindat database, the mineral name list approved by the International Mineralogical Association (IMA), the mineral classifications such as igneous minerals dataset from the rruff.info/ima (https://rruff.info/ima/), etc. The Mindat open data API (Ma et al., 2024) can help us save a lot of time in collecting datasets from the Mindat database. For example, the Python script (https://github.com/ChuBL/3DHeatmapDataPreprosses/blob/main/mindat_data_processor.py) to collect the element-based records from Mindat API. However, at the time of this work (February 2023), the location-related records in Mindat database were not available through its API. Alternatively, we crawled the records of location-based mineral lists and oxides and spinel minerals from rruff.info/evolution (https://rruff.info/evolution/). The Mindat API technical team has made more data available during the spring and summer of 2023, including part of the location-based records. The data retrieval step of our work will become easier once the API is fully established. The element-based mineral quantity, location-based mineral, igneous mineral and oxygen spinel minerals records were collected and uploaded to https://github.com/ChuBL/3DHeatmapDataPreprosses/tree/main/mindat_data and https://github.com/quexiang/Adjacency_Matrix_4_Mineral_Informatics/tree/main/data. Second, data cleansing and preprocessing were conducted to generate the nodes list and the edges list. Using the R package 'igraph', these lists can be used to build graphs and, thus, numerical adjacency matrices. Third, four community detection algorithms including Spinglass, Random Walktrap, Edge betweenness, and Optimal Modularity were used to detect communities based on the lists of nodes and edges generated in the previous step. All the detected communities, after being arranged by name or other attributes, can be sent to an R Shiny software application developed by our team. This application can handle the settings and then output the visualization results of the adjacency matrix. Fourth, based on the R Shiny application, one can quickly browse and explore patterns in the adjacency matrices by switching community detection algorithms and datasets. If an interesting pattern is detected, further analysis and new hypotheses can be proposed for in-depth research. Otherwise, users can rearrange the adjacency matrix for visualization or try other community detection algorithms.

To generate the nodes and edges lists, the 'BeautifulSoup4' Python package was used to retrieve the Mindat_ID, Locality_Name, Minerals_count, Minerals_list fields from MED (URL: 'https://rruff.info/mineral_list/MED/minerals_per_locality.php?element= {}', where {} should be replaced by the symbol for a specific mineral-forming element). Then, all the crawled location-based mineral records were exported to the ".csv" format file for each of the 72 mineral-forming elements. Algorithm 1 was used to generate the nodes and edges list files for each element, in which the nodes are the mineral species that contain a certain element, and the weight of each edge represents the number of locations where two mineral species co-occur. To include all the adjacency matrices in an online application for quick share and access, we used the R language for the graph visualization work and the Shiny package to wrap and present all the results (details in Section 3). The nodes

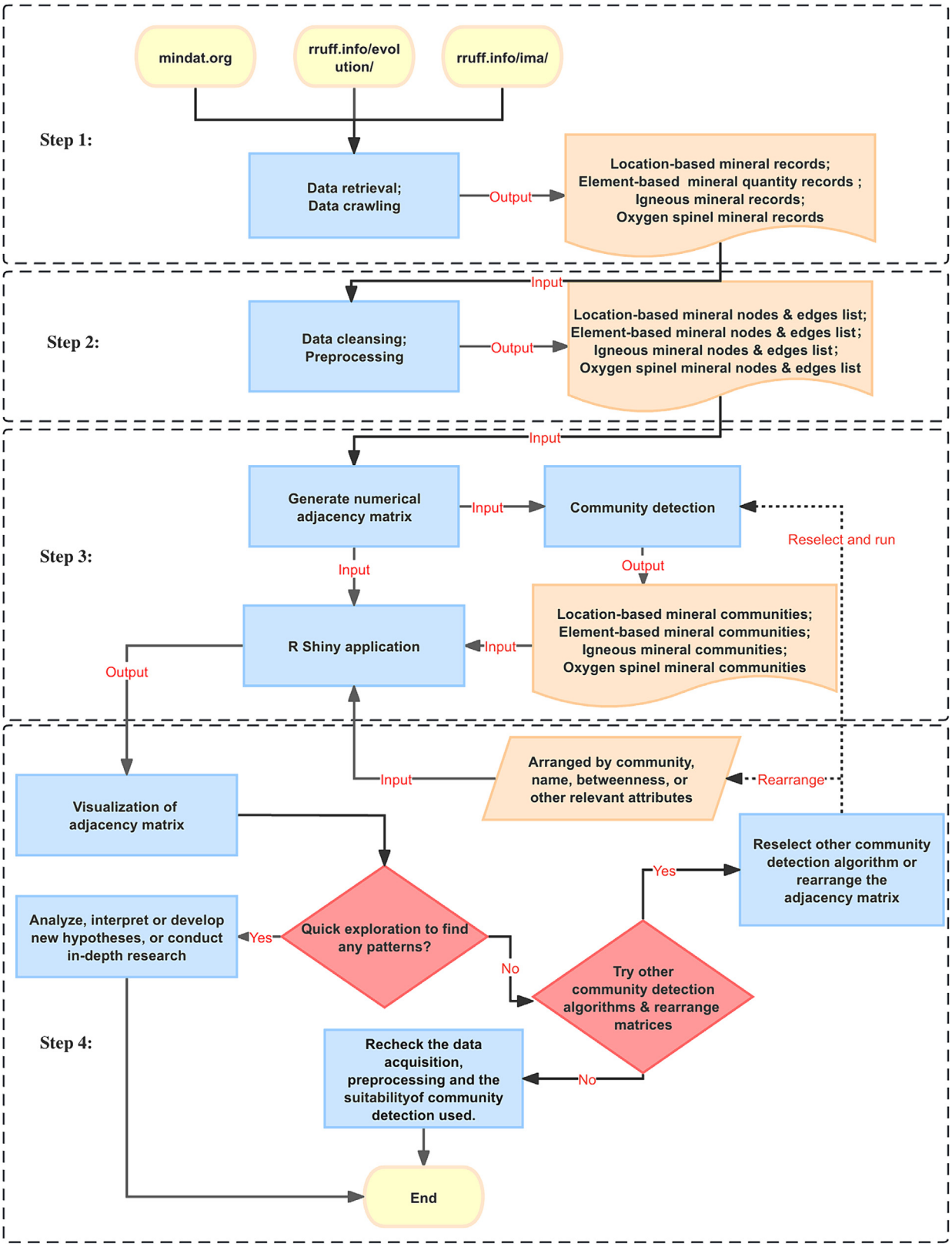


Fig. 2. A workflow for data retrieval, cleansing and software development in mineral data exploration.

and edges generated in Algorithm 1 were used to construct the graph objects (as well as the corresponding numerical adjacency matrix) with the 'igraph' package, and the community detection algorithms were applied to the graph objects to identify mineral communities of interest.

Algorithm 1 Nodes & edges list generation

```
Specify the folder path of the mineral records files (crawled by BeautifulSoup4)
Iterate over every file in a folder
    Define an empty mineral species set Species_set, traverse all mineral records, and obtain a set with unique mineral names list (list_species)
    Generate a square matrix (admatrix) with all elements equal to 0, according to the length of species_set
    For each row in mineral records
    Get the mineral list values of the Minerals_list field
    If the mineral list is not equal to 1 (which means that there are minerals co-occurring at a same location)
        for each mineral rrow in the mineral list
        for each mineral jrow in the mineral list
            if (rrow is not equal to jrow)
                find the index numbers of rrow and jrow in the mineral list
                admatrix[rrow, jrow] +=1
    Define empty from list (list_from), to list (list_to) and weights list (list_weight)
    Iterate over each row of the matrix (cur_row)
        Iterate over each column of the matrix (cur_col)
            If (cur_row >= cur_col) and (admatrix[cur_row, cur_col] is not equal to 0)
                list_from.append(list_species[cur_row])
                list_to.append(list_species[cur_col])
                list_weight.append(admatrix[cur_row, cur_col])
    Output the list_from, list_to, and list_weight as edges file
    Output the mineral names list (list_species) as nodes file
```

For some specific mineral lists (such as spinel and oxide minerals), the number of times they co-occur cannot be directly retrieved from the element-based mineral quantity list records. Because not all the minerals in the specific mineral list necessarily contain the same element. Therefore, for a user-specified mineral list, we developed Algorithm 2 to extract the number of mineral co-occurrences from the MED_export data set in MED (https://rruff.info/mineral_list/MED/exporting/2019_05_22/). To begin the work, Algorithm 2 should receive a specified mineral list from which we would like to obtain their paired co-occur counts from the records in MED_export. The fields of the output file include Mindat_ID_list, location_name_list, count_list, and specific_minerals_occur_list, where the last field records the names of minerals that occur at the same location in the specified minerals list. The Python code for both Algorithms 1 and 2 were shared on GitHub (https://github.com/quexiang/Adjacency_Matrix_4_Mineral_Informatics/tree/main/DataExtration_Cleaning).

Algorithm 2 Location-based mineral co-occurrence records extraction

```
Specify a list of mineral names (specified_minerals_list), define a list for the fields in the output file; Mindat_ID (Mindat_ID_list), location name (location_name_list), count (count_list), and the occurred minerals in the specified_minerals_list in the location (specific_minerals_occur_list)
    Read each line (cur_line) of the MED_export file
        Identify and preprocess the whitespace in quotes of cur_line
    Split cur_line by whitespace into a list of split values corresponding to the fields defined by the header of the MED_export file (med_fields_list)
    If the location recorded in cur_line is bottom level*
        Define an empty list for each field of the med_fields_list
        For each value (f_val) in the list of split values
            append f_val to its corresponding field list
            If current f_val is corresponding to the MED_mineral_at_loc_list field
                Define an empty list c_minerals_list and a 0-count variable cnt
                Split f_val by comma to get the minerals list (cur_minerals)
                for each element (c_mineral) in cur_minerals
                    if c_mineral is in the specified_minerals_list
                        append c_mineral to the c_minerals_list
                        increase cnt by 1
                        if the length of c_minerals_list is not equal to 0
                            join elements of the c_minerals_list by comma to get a string (c_minerals_str)
                            append c_minerals_str and the Mindat_ID, location name, and the count to the specific_minerals_occur_list, Mindat_ID_list, location_name_list, and count_list, respectively
Output the Mindat_ID_list, location_name_list, count_list, and the specific_minerals_occur_list to a file
```

* The bottom level in the dataset refers to whether the address is at the finest level. If the address is not at the bottom level, the processing of the address and its corresponding records should be ignored. For example, if the town level is the most fine-grained address, records of non-bottom addresses such as state records should be ignored. Otherwise, it will cause issues such as double counting.

3. Use cases and result analysis

With the above-mentioned workflows, we have developed four use cases to demonstrate the utility of adjacency matrix for rapid exploratory analysis of open mineral data. The resulting adjacency matrices for two small datasets (oxide and spinel, and igneous minerals) and two large datasets (element-based mineral counts

and location-based paragenetic minerals) were presented in the online R Shiny application (https://quexiang.shinyapps.io/Adjacency_Matrix_4_Mineral_Informatics). If the label text of figures in this section is too small or dense to read, we encourage readers to use the Shiny application to find corresponding results for better readability. The datasets and source code for data cleansing and the Shiny application are shared on GitHub (https://github.com/quexiang/Adjacency_Matrix_4_Mineral_Informatics).

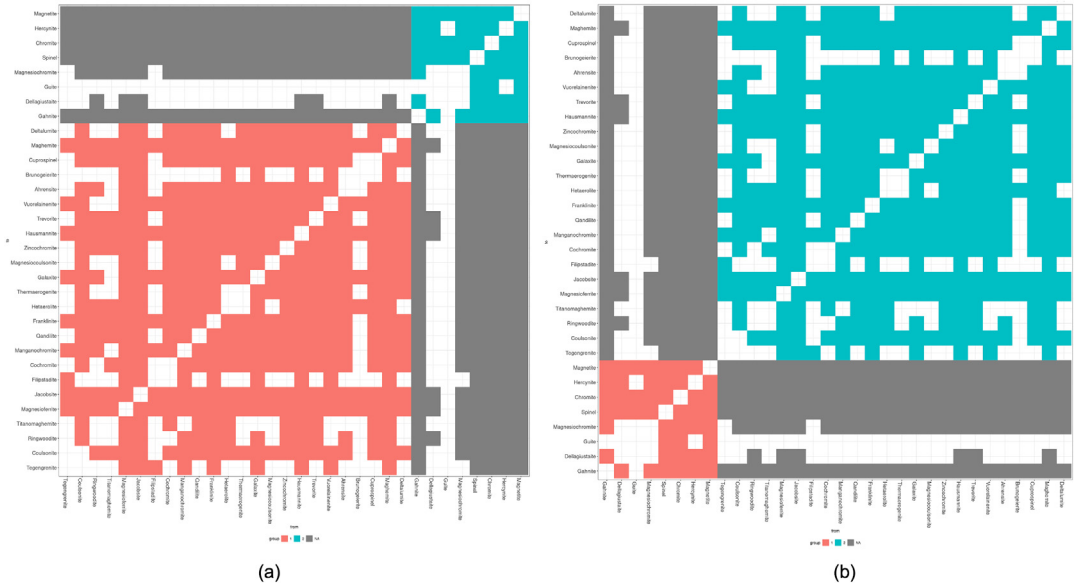
3.1. Small datasets: Oxide and spinel

This dataset is crawled from MED (https://rruff.info/evolution). The database was last updated on May 22, 2019. There are 295,584 locations in total, and each location records a list of minerals that occur there. We first obtained the list of oxide and spinel minerals from the IMA mineral name list (https://rruff.info/ima/), and then wrote code to compare the IMA oxide and spinel names with the mineral species list at each location. Thus, each pair of co-occurred oxide and spinel minerals at any location can be recorded. Using the number of co-occurrences as edge weight, an adjacency matrix was generated. Applying two community detection algorithms Optimal and Spinglass, the results show the same patterns of two groups (Fig. 3). The first group includes gahnite, dellagiustaite, guite, magnesiochromite, spinel, chromite, hercynite, and magnetite; the second group includes deltalumite, maghemite, cuprospinel, brunogeierite, ahrensite, vuorelainenite, trevorite, hausmannite, zincochromite, magnesiocoulsonite, galaxite, thermaerogenite, hetaerolite, franklinite, qandilite, manganochromite, cochromite, filipstadite, jacobsite, magnesioferrite, titanomaghemite, ringwoodite, coulsonite, and tegengrenite. A quick explanation/hypothesis is that the former group includes minerals that are usually found in magmatic and metamorphic rocks with compositions of zinc-aluminum spinel, magnesium-chromium spinel, and chromite. The latter group includes minerals common in sedimentary and volcanic rocks, such as copper-iron spinel, iron oxide, and magnetite.

3.2. Small datasets: Igneous minerals

The adjacency matrices of igneous minerals were generated using two algorithms for community detection: Walktrap (Fig. 4a) and Spinglass (Fig. 4b). In Fig. 4a, there are four clustered groups of minerals. The group in green is the largest, including actinolite, coesite, omphacite, titanite, zoisite, and more. The group in red is the second largest, including andalusite, biotite, cristobalite, grunerite, magnetite, and others. Next, most of the minerals in the light blue group, including tilleyite, spurrite, rankinite, monticellite, merwinite, melilite, larnite, and bredigite, are silicates and share similarities in crystal structure (i.e., the so-called "melilite structure"), chemical composition (i.e., often composed of calcium, magnesium, aluminum, and silicon), and formation conditions (i.e., high pressure, high temperature environments). The other two species brucite ($Mg(OH)_2$) and periclase (MgO) are included in the light blue group perhaps due to the metal element magnesium, and also because they and the other mineral species in the light blue group are all related to certain types of metamorphic and igneous rocks. The group in purple includes wollastonite, vesuvianite, tremolite, spinel, serpentine, scapolite, pyrrhotite, phlogopite, perovskite, humite, grossular, forsterite, feldspathoid, diopside, calcite, and anorthite. They are minerals that are more widely distributed on Earth, and they usually contain silicon and other mineral-forming elements, such as calcium, magnesium, and aluminum. Besides variations in occurrence and composition, the crystal structures of the light blue and purple group are different. Representative crystal structures of minerals in the light blue group are chain, ring, and layered structures, while the minerals in the purple group are primarily octahedral, tetrahedral, and orthogonal structures. Additionally, the physical properties of minerals in those groups were briefly checked in our work. For instance, the brucite in the light blue group is a soft, brittle mineral, while the pyrrhotite in the purple group is a harder magnetic mineral. A more detailed examination of the physical properties of those clustered minerals in each group might lead to new insights.

The groups generated by the Walktrap and Spinglass algorithms show some similar patterns. For example, all minerals in the green group of the Spinglass result (Fig. 4b) are also in the green group of the Walktrap result (Fig. 4a). The other minerals present in the green group of Fig. 4a but not in the green group of Fig. 4b, are also worth noting. Most of them, such as zircon, staurolite, spessartine, riebeckite, rhodonite, piemontite, and actinolite, have beautiful colors and crystal forms, and are widely used in jewelry and ornament making.



(a) (b)

Fig. 3. Adjacency matrices of oxide and spinel minerals. Diagrams in (a) and (b) are based on community detection algorithms Optimal and Spinglass, respectively.

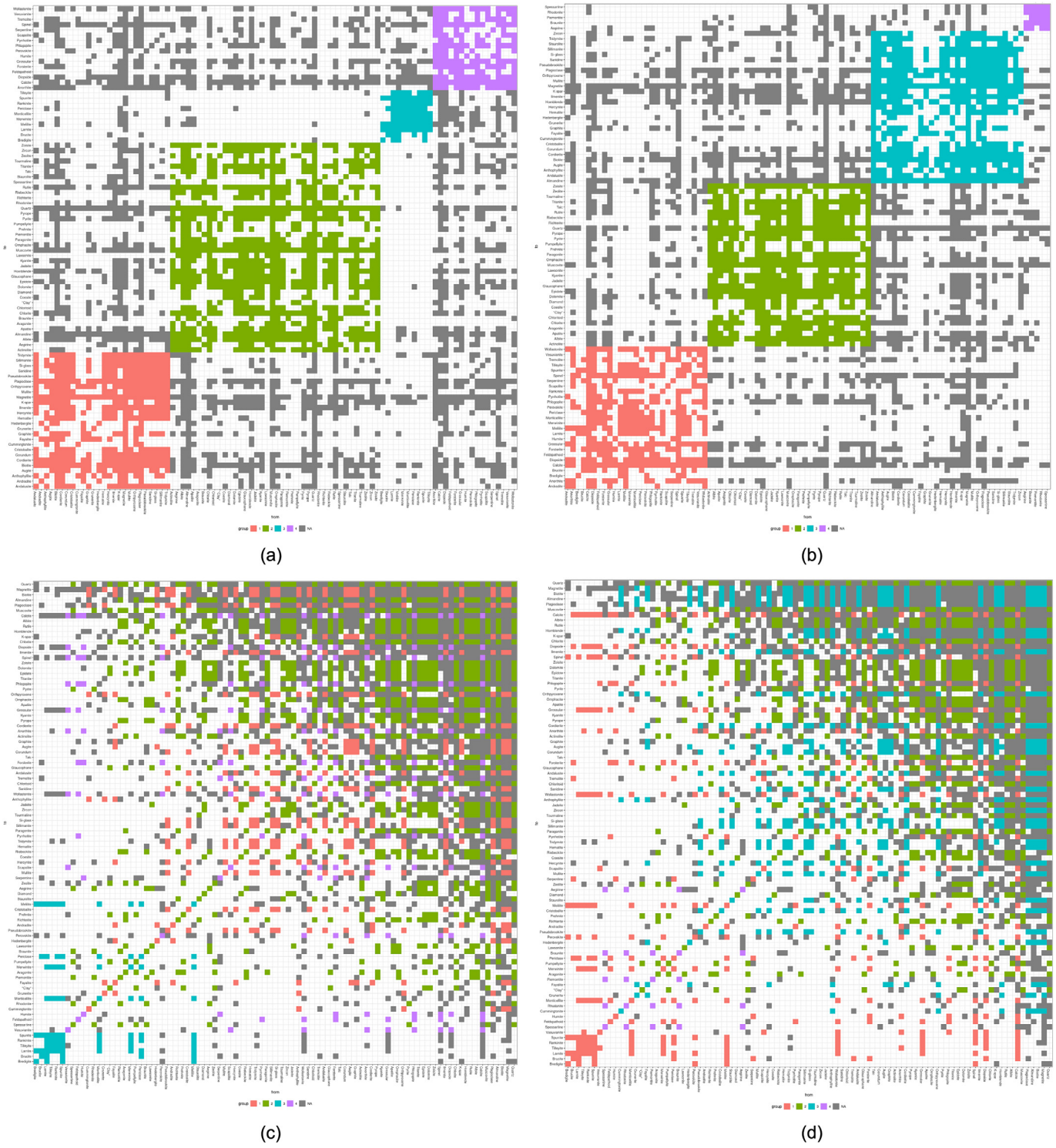


Fig. 4. Adjacency matrices of igneous minerals. (a) and (b) Results of Walktrap and Spinglass algorithms, respectively. (c) and (d) Results sorted by the eigenvalues of (a) and (b), respectively.

The matrices in Fig. 4a and 4b were further sorted by the eigenvalues, and the results are presented in Fig. 4c and 4d, respectively. It is interesting that in both figures, spurrite, rankinite, tilleyite, larnite, brucite, and bredigite are clustered in the lower left corner. These minerals are usually non-metallic, and most are either silicate or carbonate minerals, with similar chemical composition and crystallographic morphology. Moreover, most of them are rock-forming minerals that are associated with volcanic activities, metamorphism, or marine deposition. The colors of these minerals are usually white, light gray, or yellowish, while the hardness varies, ranging from 2.5–3.5 (brucite) to 5.5–6.5 (larnite).

3.3. Large datasets: Location-based paragenetic minerals

Location-based paragenetic minerals are a research topic of interest to many researchers. In our work, the dataset was

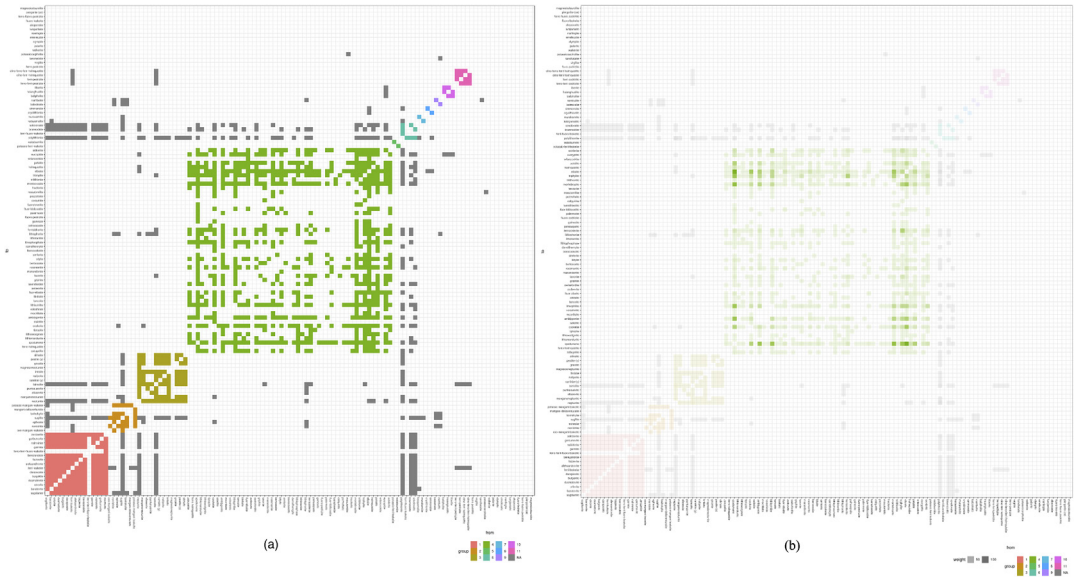


Fig. 5. Ajacency matrices of co-occurring lithium minerals. (a) Result of the Walktrap algorithm. (b) Result after changing the opacity of cells in (a) based on the number of co-occurrences in each cell (i.e., the weight of the edges). Higher opacity means a higher co-occurrence number.

retrieved from MED (https://rruff.info/evolution). On that page, an element in the "MED localities with an Element (aka Locality Registrar)" field was selected, and then a list of mineral locations containing the element along with its corresponding mineral list can be obtained. We went through all 72 mineral-forming elements and collected all the data tables. Then, Algorithm 1 in Section 2 was used to generate the nodes and edges lists for the minerals of each element, in which the nodes are the mineral species, the edges represent co-occurrences of minerals, and the numbers of co-occurrences are used as the weights of the edges.

Based on the resulting large dataset, adjacency matrices of minerals corresponding to the 72 elements were generated (see details in the "element-mineral" use case of the R Shiny application https://quexiang.shinyapps.io/adjacency_plot-master/). Many of them show interesting patterns. Here, we only use the results of lithium minerals as an illustration. Lithium is an important element in many aspects of human society (Karl et al., 2019). Lithium minerals are diverse, including clay minerals such as hectorite and pegmatite minerals such as lepidolite, amblygonite, and spodumene (Bradley et al., 2017). In our work (Fig. 5a), all the lithium minerals were clustered into 11 groups by the Walktrap algorithm. The opacity of cells can be changed to reflect the number of co-occurrences between mineral pairs (Fig. 5b). Higher opacity means a higher co-occurrence number, which further assists pattern discovery in the matrix. Fig. 5a shows four big groups (1-red, 2-brown, 3-dark yellow, and 4-green) and seven small groups.

The minerals in group 1-red, such as sogdianite, baratovite, orlovite, and dusmatovite, are silicate minerals containing rare elements, such as titanium, primarily formed in complex alkaline magma or hydrothermal deposits. The minerals in group 2-brown, including oxo-mangani-leakeite, norrishite, ephesite, sugilite, lavinskyite, mangani-dellaventuraite, potassic-mangani-leakeite, are manganese-containing silicate minerals, and valence states of manganese are high, ranging from + 2 to + 4. These minerals usually form in manganese deposits or metamorphic rocks. The members in group 3-dark yellow, including neptunite, manganoneptunite, eliseevite, etc. are silicate minerals, which contain metal elements such as sodium, iron, manganese, magnesium, cobalt, and copper, and their crystal structures are all in tunnel or chain forms. The members in group 4-green, including zabuyelite, ferro-holmquistite, spodumene, sicklerite, etc. are mainly magma or hydrothermal minerals.

The other seven groups are relatively small, but we also conducted a brief analysis of them. The group 5 includes potassic-ferri-leakeite and watatsumiite. Both are lithium-bearing mafic hornblende minerals with similar chemical compositions, and their crystal structures all belong to the monoclinic system with black to dark brown colors. Group 6 includes polylithionite, ferri-fluoro-leakeite, brannockite, and sokolovaite, which are all in the amphibole group. Group 7 includes katayamalite and murakamiite. Both are copper potassium silicate in the hexagonal crystal system. Group 8 includes cryolithionite and simmonsite. Both minerals are based on the monoclinic crystal system. Group 9 includes balestraite, and nambulite. Both minerals contain elements such as copper, bismuth, and selenium, and their crystal structures belong to the trigonal crystal system. The group 10 includes balipholite, hsianghualite and liberite. They all contain boron, and their crystal structure belongs to the silicate minerals. The group 11 includes ferro-ferri-pedrizite, ferri-pedrizite, ferro-pedrizite, etc. They are iron-containing pedrizite or holmquistite, where the prefix ferro- means Fe^{2+} and ferri- means Fe^{3+}. Their crystal structures all belong to the hornblende minerals.

3.4. Large datasets: Element-based mineral counts

For the large datasets of element-based mineral counts (first built with records from MED and then updated with records from mindat.org), a list of 73 adjacency matrices (each is 72 × 72) was built to explore patterns. The first matrix consists of 72 ore-forming elements represented along both the *X* and *Y* axes. Each cell within this matrix indicates the number of mineral species in which the *X* and *Y* elements coexist. Additionally, there are 72 other matrices, each representing an additional element along the *Z*-axis. Within these matrices, each cell denotes the number of mineral species in which the *X*, *Y*, and *Z* elements co-exist. In other words, the first adjacency matrix shows the associations between two elements amongst the 5865 IMA-approved mineral species (in February 2023), while the other 72 matrices reflect the associations between three elements amongst all the minerals. By comparing with the first matrix, we can also see the impact of

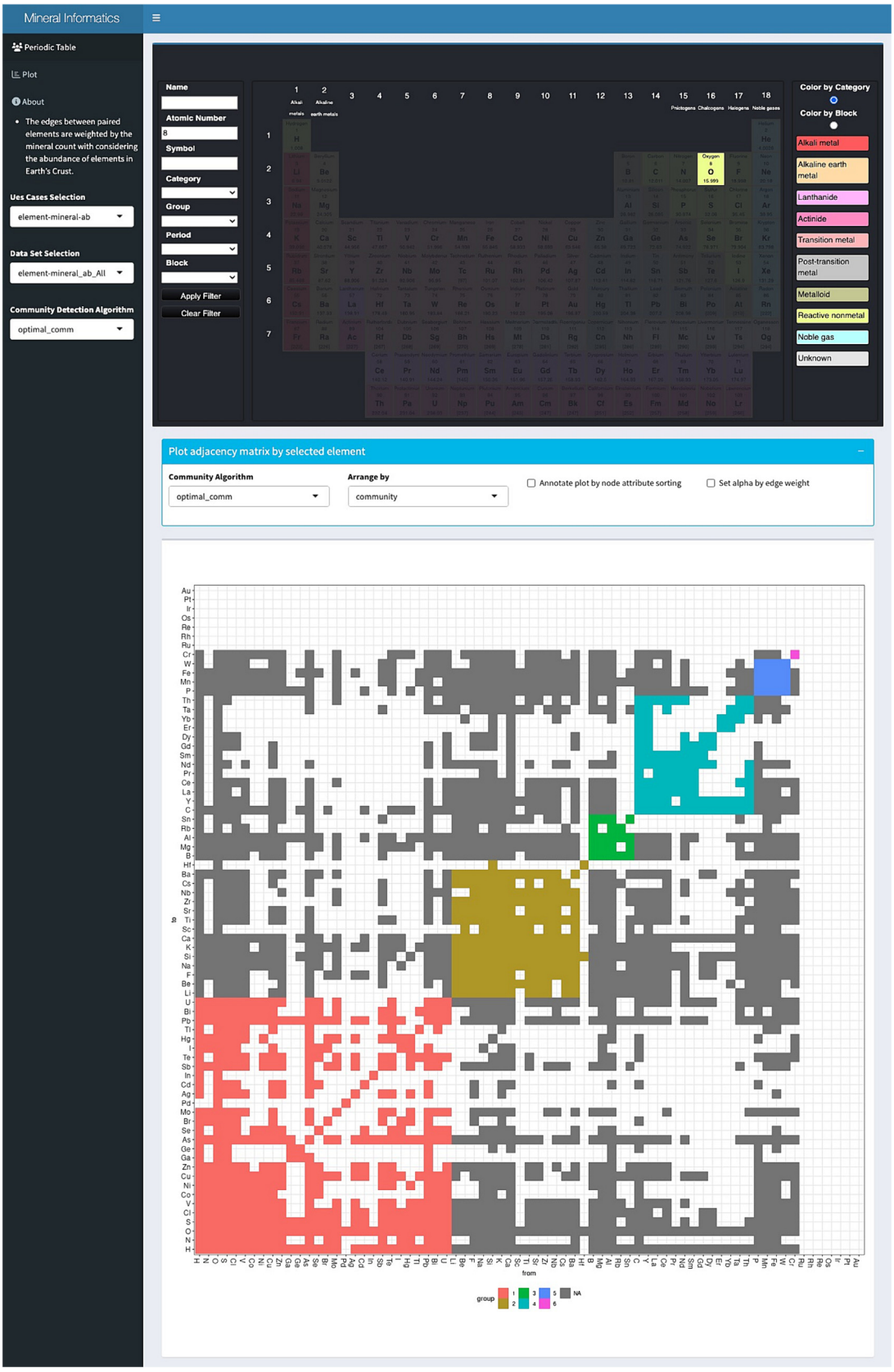


Fig. 6. Using an interactive periodic table as guide and filter to load adjacency matrices showing correlations of elements amongst mineral species. The demo is accessible at https://quexiang.shinyapps.io/Adjacency_Matrix_4_Mineral_Informatics/.

the third element on the relationship between the two original elements with regard to the number of mineral species.

An interactive periodic table (open-source code from Interactive Periodic Table in JavaScript version 1.0, https://www.codedrome.-com/interactive-periodic-table-in-javascript) was integrated into our R shiny application (Fig. 6) to quickly view the attributes of the corresponding elements and make selections of adjacency matrices to see. When an element on the periodic table is clicked, the corresponding adjacency matrix (with the clicked element on the *Z*-axis) will be displayed or switched out at the bottom of the page. Using the controls on the left of the periodic table, some special elements, such as alkaline earth metal elements including Be, Mg, Ca, Sr, Ba, and Ra, can be filtered out and highlighted in the periodic table to help users narrow the scope of elements to select and explore. Using controls below the periodic table, i.e., the drop-down lists in the "Plot adjacency matrix by selected element" box, users can switch community detection algorithms to generate an adjacency matrix, and change the opacity of cells based on the number of minerals in each cell (i.e., weight of the edge). With those controls, users can quickly bring up adjacency matrices and explore the association patterns hidden in them.

This use case has produced many adjacency matrices. Here, we take results with sulfur (S) on the *Z* axis as an example to illustrate the data exploration. Fig. 7a–e shows the adjacency matrices without sulfur (S) on the *Z*-axis, and Fig. 7as–es illustrates the results after the inclusion of S. In a simplified understanding, Fig. 7a–e shows the "friendship" of two elements amongst mineral species, and Fig. 7as–es shows the updated "friendship" after the inclusion of S.

Seven groups were clustered in Fig. 7as, comprising three big groups (1-red, 2-brown, 3-green) and four small groups. Group 1-red is the largest, and the minerals in this group are various, including sulfate minerals such as alunite and kaolinite, sulfide minerals such as connellite and chalcocite, carbonate minerals such as calcite and dolomite, phosphate minerals such as apatite, pyromorphite and nitrate minerals such as niter and sodium nitrate, and more. The elements H and O are in this group, and there are about 546 IMA-approved minerals that contain H, O, and S.

The group 2-brown in Fig. 7as contains elements Be, Cr, Fe, Zn, Ga, Ge, and W. Minerals composed of these elements and S share certain similarities in structures and properties, including (1) Minerals in this group are typically sulfides. S usually exists in the form of divalent anions (S^{2-}), bonding with metal or metalloid atoms through ionic or covalent bonds. For example, Zn and S are found together in the mineral sphalerite (ZnS), while Fe and S are found together in the minerals pyrite (FeS_2) and pyrrhotite ($Fe_{(1-x)}S$). (2) These minerals have diverse crystal structures, such as cubic, monoclinic, or hexagonal systems. (3) Minerals in this group can display a range of colors and may possess unique optical properties. For instance, germanium sulfide and gallium sulfide are applied in infrared optical materials. (4) These sulfide minerals generally exhibit good thermal stability at higher temperatures, although their melting points may differ due to variations in composition and crystal structure. (5) Some minerals in this group may exhibit electrical conductivity or semiconductor behavior, such as germanium sulfide and gallium sulfide. (6) The solubility of these sulfide minerals is generally low in water, contributing to the formation of ore deposits in the Earth's crust. Under specific geochemical conditions, these sulfides may dissolve and migrate, forming new ore deposits. Moreover, this group is consistent with geologic observations of the S-containing ore body. For example, the most common Fe and S mineral, pyrite, is in this group. Pyrite is a common mineral that appears in a variety of geological settings. It forms under a wide range of conditions, from sedimentary rocks to hydrothermal veins, and is often associated with other sulfide minerals such as

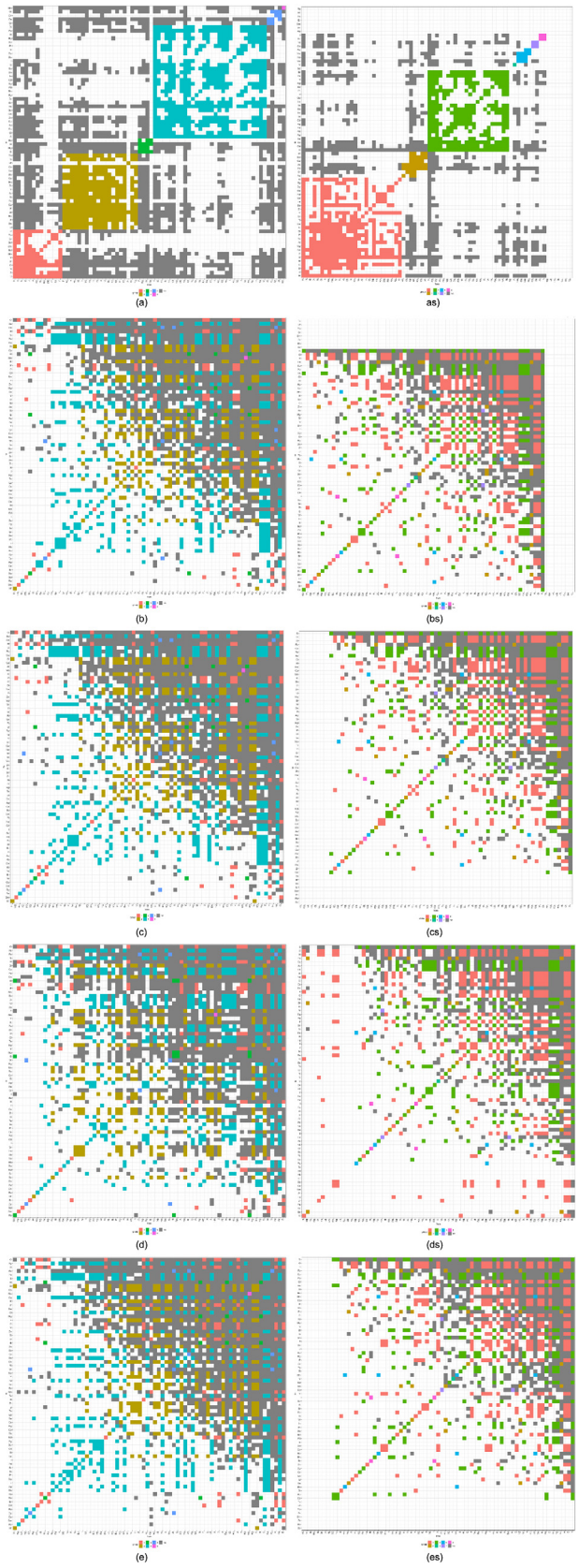


Fig. 7. Adjacency matrices showing the impacts of S on the element correlations amongst minerals. Matrix (a) is the original matrix without S on the *Z* axis, clustered with the Optimal algorithm, and (b) to (e) are results of resorting (a) by closeness, degree, betweenness, and eigen, respectively. Matrix (as) is the result when S joins on the *Z* axis, and (bs) to (es) are results of resorting (as) by closeness, degree, betweenness, and eigen, respectively.

galena and chalcopyrite. In sedimentary rocks, pyrite is commonly found in organic-rich shales and coal seams, where it forms through the decay of organic matter under reducing conditions. In hydrothermal veins, pyrite is often found along with other sulfide minerals in veins that have been deposited by hot fluids that have migrated through fractures in the rock. It can also be found in metamorphic rocks, particularly in those that have undergone regional metamorphism, such as schist and gneiss.

Group 3-green is the second largest, containing elements such as Cu, Pd, Ag, Os, Au, Ir, Pt, Pb, and others. Minerals formed by these elements and S share some similarities: (1) High density: The sulfide minerals in this group usually have high densities. (2) Conductivity: Sulfide minerals containing metal ions, particularly those with highly conductive metal elements such as Cu and Ag, possess specific conductivity due to the presence of these metal ions. (3) Higher melting point: Sulfide minerals in this group typically have higher melting points due to the stronger chemical bonds formed between the metal ions and the S ions. In addition, minerals in this group may come from a variety of deposits, such as (1) Magmatic deposits: metal elements combine with S, and during the crystallization of magma, sulfide deposits are formed. Examples include porphyry copper-nickel sulfide deposits and porphyry gold deposits; (2) Hydrothermal deposits: hydrothermal fluids are rich in metal elements and S, which cool and precipitate at the surface or underground, forming sulfide deposits. Examples include hydrothermal gold and copper deposits.

Another interesting pattern in Fig. 7as is that there are no mineral species formed between S and some rare earth elements: Rb, Pr, Sm, Er, Yb, Hf, and Ta (see the blank cells in the top and right parts of Fig. 7as). This can also be verified by checking the IMA mineral name list via https://rruff.info/ima.

The adjacency matrices can also be rearranged to do more comparisons and illustrate new patterns. Fig. 7b–e and Fig. 7bs–es are the resorted results of Fig. 7a and Fig. 7as, respectively. Through pairwise comparison between Fig. 7bs–es and Fig. 7b–e, it can be found that the addition of S changes the order of elements in the original matrices. The elements O, H, Fe, and Cu are always the top four most "friendly" to S minerals (Fig. 7bs–es). While in the natural environment, the elements O, As, Fe, S, and H are most closely connected amongst all minerals. Moreover, as Fig. 7bs–es has a lot more blank cells compared with Fig. 7b–e, it indicates that the presence of S may have affected the pairing of many elements in mineralization.

4. Discussion

Data exploration is a method of quickly analyzing data and discovering potential relationships, characteristics, and laws through visualization, statistics, and many other techniques. Mineralogy studies the composition, properties, and distribution of minerals. With vast amounts of open mineral data, such as Mindat, RRUFF, and MED, data exploration plays an increasingly influential role in data-intensive mineral informatics studies. As presented in Morrison et al. (2017), Ma et al. (2017), and Hazen et al. (2019), as well as the use cases of this study, data exploration has been successfully applied to understand mineral properties, identify mineral associations, and discover mineral occurrence patterns. Data exploration enables researchers to analyze the properties of minerals and their variations, such as chemical compositions, crystal structures, physical properties, and more. By analyzing large datasets, researchers can identify minerals that tend to coexist or associate with each other, leading to the discovery of previously unrecognized mineral associations. One interesting example is the potential relationship between rare earth minerals. For instance, there might be close associations in minerals containing neodymium and dysprosium, which were identified by using machine learning methods (Jahoda et al., 2021; Morrison et al., 2023). Further, by analyzing spatial and temporal data, researchers can discover the presence of minerals in specific regions or under certain geological conditions. Moreover, the observed association patterns from data exploration provide a valuable starting point for further in-depth study. Researchers can formulate new questions about mineral properties, occurrences, associations, and evolution, guiding them in designing experiments, collecting new datasets, and deploying advanced statistical analyses and machine learning algorithms.

In data exploration, various techniques of visualization can be utilized to represent mineral data as charts, images, and interactive diagrams, which help researchers browse data more intuitively and discover patterns such as relationships or trends in them. Many statistical and data mining techniques, such as spatial analysis, association rule mining, and prediction, can also be incorporated to boost the data exploration process. This study, to our knowledge, is the first time that adjacency matrix is applied to explore a large amount of mineral data. Our work illustrates that many interesting patterns can be found in the large element-based mineral count datasets and the location-based mineral co-existence datasets through the many community detection algorithms and visualization techniques in adjacency matrices. The results prove that adjacency matrix is a complementary method to other ways of data exploration and analysis in mineral informatics, especially in terms of network visualization. Together with workflows that retrieve, integrate and cleanse multi-source open mineral data and platforms that quickly deploy and share visualization results, adjacency matrix can significantly contribute to data-driven discovery in mineral informatics, helping researchers discover new mineral associations, improve the accuracy of mineral identification and classification, and possibly even predict undiscovered minerals at specific locations.

Besides the examples illustrated in Section 3, the R Shiny application provides interactive browsing of adjacency matrices for many other datasets in the four use cases. With the periodic table and other lists and controls on the graphical interface, users can quickly make a selection to obtain the adjacency matrix and then analyze it. The clear layout of cells in the adjacency matrix can effectively avoid node overlapping and edge crossings in the network visualization, especially when the network becomes dense. Moreover, it provides a variety of interactive operations for rearranging the matrix (e.g., using communities, closeness, degree, betweenness, and eigen), allowing users to observe changes and patterns in the adjacency matrix from different perspectives.

Although the Shiny application has the above-mentioned advantages, it is still in the preliminary stage and has some limitations that can be addressed in future work. First, the current application does not support automatic data updates and can only support data input in two formats: csv and RData. Data acquisition and integration still depend on web crawling and data file parsing. The minerals in the IMA mineral list are constantly updating, and our current results of adjacency matrices cannot catch up with the increasing list. In late spring and summer of 2023, the Mindat open data API (Application Programming Interface) had achieved a solid progress (Ma et al., 2024). In June 2023, The Mindat API was able to provide most of the datasets mentioned in our study, except location-based mineral lists. As more data subjects are released on the Mindat API, we believe it can be used to construct an automatic data pipeline for building adjacency matrices. Second, the current application only supports limited community detection algorithms. There are many other and increasing number of community detection algorithms such as the Louvain (Blondel et al., 2008), Infomap (Rosvall and Bergstrom, 2008), Fast Greedy (Clauset et al., 2004), Clique Percolation Method (CPM) (Palla

et al., 2005), and Label Propagation Algorithm (LPA) (Raghavan et al., 2007), etc., and some of them may be well suited for the community detections of mineral associations. Third, the current application only built the adjacency matrices of four use cases. In fact, there are still many association patterns in the mineral data that can be explored and analyzed using the adjacency matrix method. With the established Mindat API, we should be able to retrieve new datasets to make a long list of other use cases to explore. Fourth, the current application lacks rich filtering rules and operations to assist in browsing the adjacency matrices. Although many adjacency matrices were generated in this study and we were able to resort cells in the matrices by different ways, all the results are static images. A potential update is to make the matrix and data more interactive, such as using some visualization packages in JavaScript. The ideal situation is to interactively filter out some elements or minerals for better pattern recognition. Fifth, the current adjacency matrices are two-dimensional, and there should be ways to build three-dimensional matrices to illustrate the associations amongst minerals, elements, and localities (e.g., Zhang et al., 2024). Another alternative way is to build a set of two-dimensional adjacency matrices, each illustrating a certain part of the associations in a complex dataset. Those two- and three-dimensional matrices can be used together with other visualization methods, such as maps and timelines, for an even better data exploration experience.

5. Conclusions

Facing the increasing open mineral data, data exploration is an efficient method to analyze patterns in the data and initiate research questions for further studies. This paper focuses on applying adjacency matrix to explore a variety of associations in mineral properties and occurrences. Our work includes algorithms for open mineral data extraction and cleansing, visualization techniques for adjacency matrix creation and resorting, and an R Shiny application that shares and presents all the results online. An interactive periodic table and other controls have been created on the Shiny application to help users browse the results of interest. Besides the examples introduced in this paper, the Shiny application has many other adjacency matrices from the two big datasets mentioned in Sections 3.3 and 3.4. All the datasets and source code are also shared online (see Section 6). Researchers are welcome to reuse the Shiny application and adapt the code and datasets for their own works. Both the open mineral data environment and the data exploration techniques are quickly evolving. We are aware of those new opportunities and have planned a list of action items for future work, such as those listed in Section 4. We also welcome researchers in mineralogy and geoinformatics to send their feedback and collaborate on the future extension of the Shiny application, which includes automated data pipelines, new use cases, new visualization techniques, and more.

6. Computer code availability

Name of the code/library: Adjacency matrix for mineral informatics

Contact: Xiang Que (xiangq@uidaho.edu) or Xiaogang Ma (max@uidaho.edu), +1 208 885 1547.

Hardware requirements: CPU - Apple M1, Memory – 32 GB (or similar setting).

Program language: R and Python

Software required: R Studio for coding. Web browser such as Safari or Chrome for running the R Shiny application.

Program size: 48 MB (including datasets).

The source codes (R and Python scripts) and datasets for both data cleansing and the R Shiny application are shared on GitHub at: https://github.com/quexiang/Adjacency_Matrix_4_Mineral_Informatics.

The deployed R Shiny application for all the adjacency matrix results is accessible at: https://quexiang.shinyapps.io/Adjacency_Matrix_4_Mineral_Informatics.

CRediT authorship contribution statement

Xiang Que: Conceptualization, Methodology, Software, Writing – original draft, Writing – review & editing. **Jingyi Huang:** Conceptualization, Methodology, Validation, Writing – review & editing. **Jolyon Ralph:** Data curation, Resources, Writing – review & editing. **Jiyin Zhang:** Validation, Writing – review & editing. **Anirudh Prabhu:** Validation, Writing – review & editing. **Shaunna Morrison:** Validation, Writing – review & editing. **Robert Hazen:** Validation, Writing – review & editing. **Xiaogang Ma:** Methodology, Funding acquisition, Validation, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the U.S. National Science Foundation (Grant No. 2126315). The authors thank the discussion with Dr. Chengbin Wang and many other colleagues during the 2022 Mineral Informatics Datathon at the Earth and Planets Laboratory, Carnegie Institution for Science. The authors also thank three anonymous reviewers for their constructive comments on an earlier version of the paper.

References

Bavelas, A., 1950. Communication patterns in task-oriented groups. J. Acoust. Soc. Am. 22 (6), 725–730.
Biggs, N., Biggs, N.L., Norman, B., 1993. Algebraic Graph Theory (2nd Edition). Cambridge University Press, New York, p. 216.
Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E., 2008. Fast unfolding of communities in large networks. J. Stat. Mech: Theory Exp. 2008 (10), P10008.
Bradley, D.C., McCauley, A.D., Stillings, L.M., 2017. Mineral-deposit model for lithium-cesium-tantalum pegmatites. U.S. Geological Survey Scientific Investigations Report 2010–5070–O, Reston, VA, 48 p. doi: 10.3133/sir20105070O.
Brandes, U., Delling, D., Gaertler, M., Gorke, R., Hoefer, M., Nikoloski, Z., Wagner, D., 2007. On modularity clustering. IEEE Trans. Knowl. Data Eng. 20 (2), 172–188.
Brouwer, A.E., Haemers, W.H., 2012. Spectra of Graphs. Springer, New York, p. 245.
Chen, M., Xiao, F., 2023. Projection pursuit random forest for mineral prospectivity mapping. Math. Geosci. 55 (7), 963–987.
Clauset, A., Newman, M.E., Moore, C., 2004. Finding community structure in very large networks. Phys. Rev. E 70, (6) 066111.
Cvetković, D.M., Doob, M., Sachs, H., 1995. Spectra of Graphs. Johann Ambrosius Barth Verlag, Heidelberg-Leipzig, p. 447.
Diestel, R., 2017. Graph Theory (5th Edition). Springer, Berlin, p. 446.
Farkas, I., Derényi, I., Jeong, H., Neda, Z., Oltvai, Z.N., Ravasz, E., Schubert, A., Barabási, A.L., Vicsek, T., 2002. Networks in life: scaling properties and eigenvalue spectra. Physica A 314 (1–4), 25–34.
Fekete, J.D., 2009. Visualizing networks using adjacency matrices: Progresses and challenges. In: Proceedings of the 11th IEEE International Conference on Computer-Aided Design and Computer Graphics, Huangshan, China, pp. 636-638. Doi: 10.1109/CADCG.2009.5246813.
Field, M., Stiefenhofer, J., Robey, J., Kurszlaukis, S., 2008. Kimberlite-hosted diamond deposits of southern Africa: a review. Ore Geol. Rev. 34 (1–2), 33–75.
Freeman, L.C., 2002. Centrality in social networks: conceptual clarification. In: Scott, J. (Ed.), Social Network: Critical Concepts in Sociology. Routledge, New York, pp. 238–263.
Girvan, M., Newman, M.E., 2002. Community structure in social and biological networks. Proc. Nat. Acad. Sci. 99 (12), 7821–7826.

Golden, J.J., Downs, R.T., Hazen, R.M., Pires, A.J., Ralph, J., 2019. Mineral evolution database: data-driven age assignment, how does a mineral get an age? In GSA Annual Meeting, Phoenix, Arizona, USA. https://doi.org/10.1130/abs/2019AM-334056.

Hazen, R.M., 2014. Data-driven abductive discovery in mineralogy. Am. Mineral. 99 (11–12), 2165–2170.

Hazen, R.M., Downs, R.T., Elesish, A., Fox, P., Gagné, O., Golden, J.J., Grew, E.S., Hummer, D.R., Hystad, G., Krivovichev, S.V., Li, C., Liu, C., Ma, X., Morrison, S.M., Pan, F., Pires, A.J., Prab-hu, A., Ralph, J., Runyon, S.E., Zhong, H., 2019. Data-driven discovery in mineralogy: recent advances in data resources, analysis, and visualization. Engineering 5, 397–405.

Hazen, R.M., Morrison, S., Williams, J., Prabhu, A., Eleish, A., Fox, P., 2021. Mineral Informatics: Analysis and Visualization of Minerals through Time and Space. AGU Fall Meeting 2021, New Orleans, LA, IN13A-01.

Hazen, R.M., Morrison, S.M., 2022. On the paragenetic modes of minerals: a mineral evolution perspective. Am. Mineral. 107 (7), 1262–1287.

Hey, T., Tansley, S., Tolle, K., 2009. The Fourth Paradigm: Data-Intensive Scientific Discovery. Microsoft Corporation, Redmond, WA, p. 252.

Jahoda, P., Drozdovskiy, I., Payler, S.J., Turchi, L., Bessone, L., Sauro, F., 2021. Machine learning for recognizing minerals from multispectral data. Analyst 146 (1), 184–195.

Jowitt, S.M., Mudd, G.M., Weng, Z., 2013. Hidden mineral deposits in Cu-dominated porphyry-skarn systems: how resource reporting can occlude important mineralization types within mining camps. Econ. Geol. 108 (5), 1185–1193.

Karl, N.A., Mauk, J.L., Reyes, T.A., Scott, P.C., 2019. Lithium Deposits in the United States. U.S. Geological Survey Data Release. Reston, VA. 10.5066/P9ZKRWQF.

Keskinen, R., Hillier, S., Liski, E., Nuutinen, V., Nyambura, M., Tiljander, M., 2022. Mineral composition and its relations to readily available element concentrations in cultivated soils of Finland. Acta Agriculturae Scandinavica, Section B—Soil & Plant. Science 72 (1), 751–760.

Lafuente, B., Downs, R.T., Yang, H., Stone, N., 2015. The power of databases: the RRUFF project. In: Armbruster, T., Danisi, R.M. (Eds.), Highlights in Mineralogical Crystallography. De Gruyter, Berlin and Boston, pp. 1–30.

Ma, X., Hummer, D., Golden, J.J., Fox, P.A., Hazen, R.M., Morrison, S.M., Downs, R.T., Madhikarmi, B.L., Wang, C., Meyer, M.B., 2017. Using visual exploratory data analysis to facilitate collaboration and hypothesis generation in cross-disciplinary research. ISPRS Int. J. Geo Inf. 6 (11), 368. https://doi.org/10.3390/ijgi6110368.

Ma, X., Ralph, J., Zhang, J., Que, X., Prabhu, A., Morrison, S.M., Hazen, R.M., Wyborn, L., Lehnert, K., 2024. OpenMindat: open and FAIR mineralogy data from the Mindat database. Geosci. Data J. 11 (1), 94–104. https://doi.org/10.1002/gdj3.204.

Ma, X., 2023. Data Science for Geoscience: Recent Progress and Future Trends from the Perspective of a Data Life Cycle. In: Ma, X., Mookerjee, M., Hsu, L., Hills, D. (Eds.), Recent Advancement in Geoinformatics and Data Science. Geological Society of America Special Paper V. 558, Boulder, CO, pp. 57-69.

Morrison, S.M., Liu, C., Eleish, A., Prabhu, A., Li, C., Ralph, J., Downs, R.T., Golden, J.J., Fox, P., Hummer, D.R., Meyer, M.B., 2017. Network analysis of mineralogical systems. Am. Mineral. 102 (8), 1588–1596.

Morrison, S.M., Prabhu, A., Eleish, A., Hazen, R.M., Golden, J.J., Downs, R.T., Perry, S., Burns, P.C., Ralph, J., Fox, P., 2023. Predicting new mineral occurrences and planetary analog environments via mineral association analysis. PNAS Nexus 2 (5), pgad110.

Okoe, M., Jianu, R., Kobourov, S., 2019. Node-link or adjacency matrices: old question, new insights. IEEE Trans. Vis. Comput. Graph. 25 (10), 2940–2952. https://doi.org/10.1109/TVCG.2018.2865940.

Palla, G., Derényi, I., Farkas, I., Vicsek, T., 2005. Uncovering the overlapping community structure of complex networks in nature and society. Nature 435 (7043), 814–818.

Pollard, P.J., Taylor, R.G., Peters, L., 2005. Ages of intrusion, alteration, and mineralization at the Grasberg Cu-Au deposit, Papua, Indonesia. Econ. Geol. 100 (5), 1005–1020.

Pons, P., Latapy, M., 2006. Computing communities in large networks using random walks. J. Graph Algorithms Appl. 10 (2), 191–218.

Prabhu, A., Morrison, S.M., Fox, P., Ma, X., Wong, M.L., Williams, J., McGuinness, K.N., Krivovichev, S., Lehnert, K.A., Ralph, J.P., Lafuente, B., 2023. What is mineral informatics? Am. Mineral. 108 (7), 1242–1257. https://doi.org/10.2138/am-2022-8613.

Raghavan, U.N., Albert, R., Kumara, S., 2007. Near linear time algorithm to detect community structures in large-scale networks. Phys. Rev. E 76, (3) 036106.

Ralph, J., Ma, X., Prabhu, A., Martynov, P., 2022. Building OpenMindat for FAIR mineralogical data access. EarthCube 2022 Annual Meeting, San Diego, CA. Poster.

Rayzman, V.L., Aturin, A.V., Pevzner, I.Z., Sizyakov, V.M., Ni, L.P., Filipovich, I.K., 2003. Extracting silica and alumina from low-grade bauxite. J. Metals 55, 47–50.

Reichardt, J., Bornholdt, S., 2006. Statistical mechanics of community detection. Phys. Rev. E 74, (1) 016110.

Rosvall, M., Bergstrom, C.T., 2008. Maps of random walks on complex networks reveal community structure. Proc. Nat. Acad. Sci. 105 (4), 1118–1123.

Sadeghi, B., 2021. Concentration-concentration fractal modelling: a novel insight for correlation between variables in response to changes in the underlying controlling geological-geochemical processes. Ore Geol. Rev. 128, 103875.

Tukey, J.W., 1977. Exploratory Data Analysis. Addison-Wesley, Reading, PA, p. 688.

Wang, C., Hazen, R.M., Cheng, Q., Stephenson, M.H., Zhou, C., Fox, P., Shen, S., Oberhansli, R., Hou, Z., Ma, X., Feng, Z., Fan, J., Ma, C., Hu, X., Luo, B., Wang, J., 2021. The deep-time digital Earth program: data-driven discovery in the geosciences. Natl. Sci. Rev. 8 (9), nwab027. https://doi.org/10.1093/nsr/nwab027.

Wang, B., Ma, K., Wu, L., Qiu, Q., Xie, Z., Tao, L., 2022. Visual analytics and information extraction of geological content for text-based mineral exploration reports. Ore Geol. Rev. 144, 104818.

Xiao, F., Chen, J.G., 2012. Fractal projection pursuit classification model applied to geochemical survey data. Comput. & Geosci. 45, 75–81.

Yousefi, M., Carranza, E.J.M., Kreuzer, O.P., Nykänen, V., Hronsky, J.M., Mihalasky, M. J., 2021. Data analysis methods for prospectivity modelling as applied to mineral exploration targeting: state-of-the-art and outlook. J. Geochem. Explor. 229, 106839.

Zhang, J., Que, X., Madhikarmi, B., Hazen, R.M., Ralph, J., Prabhu, A., Morrison, S.M., Ma, X., 2024. Using a 3D heat map to explore the diverse correlations among elements and mineral species. Applied Computing & Geosciences 21, 100154.

Zuo, R., Wang, J., Xiong, Y., Wang, Z., 2021. The processing methods of geochemical exploration data: past, present, and future. Appl. Geochem. 132, 105072.