Data complexity: An FCA-based approach

Alexey Buzmakov, Egor Dudyrev, Sergei O Kuznetsov, Tatiana Makhalova, Amedeo Napoli

Data complexity: An FCA-based approach

Alexey Buzmakov[a], Egor Dudyrev[b,c], Sergei O. Kuznetsov[b], Tatiana Makhalova[c], Amedeo Napoli[c]

[a] *HSE University, Perm, 614070, Russia*
[b] *HSE University, Moscow, 109028, Russia*
[c] *Université de Lorraine, CNRS, Inria, LORIA, Nancy, 54000, France*

Abstract

In this paper we propose various complexity measures of a dataset in terms of Formal Concept Analysis (FCA). On the one hand, we follow the lines of the research of the "closure structure" and the "closure index" based on minimum generators of intents (closed itemsets). On the other hand, we would like to capture statistical properties of a dataset, not just extremal characteristics, such as the size of a passkey. In the following we introduce an alternative approach where we try to measure the complexity of a dataset in terms of five main elements that can be computed in a concept lattice, namely intents (closed sets of attributes), pseudo-intents, proper premises, keys (minimal generators), and passkeys (minimum generators). We study the distribution of these different elements in various datasets, both real and synthetic. We also investigate the relations that these five elements have with one another, and the relations with implications and association rules.

Keywords: Formal Concept Analysis, data complexity, attribute sets, equivalence classes, closed sets, generators, keys

1. Introduction

In this paper we are interested in measuring and computing"complexity" of a dataset in terms of Formal Concept Analysis (FCA [1]). On the one hand, we follow the lines of [2] where the "closure structure" and the "closure index" are introduced and based on the so-called passkeys, i.e., minimum generators in an equivalence class of itemsets. On the other hand, we would like to capture statistical properties of a dataset, not just extremal characteristics such as the size of a passkey. In the following we introduce an alternative

approach where we try to measure the complexity of a dataset in terms of five main elements that can be computed in a concept lattice, namely intents (closed sets), pseudo-intents, proper premises, keys (minimal generators), and passkeys (minimum generators). We follow a more practical point of view and we study the distribution of these different elements in various datasets. We also investigate the relations that these five elements have with one another, and the relations with implications and association rules.

For example, the number of intents gives the size of the lattice, while the number of pseudo-intents gives the size of the Duquenne-Guigues basis [3], and thus the size of the minimal implication basis representing the whole lattice. The size of the covering relation of the concept lattice gives the size of the "base" of association rules. Moreover, passkeys are indicators related to the closure structure and the closure index indicates the number of levels in the structure. The closure structure represents a dataset, so that closed itemsets are assigned to the level of the structure given by the size of their passkeys. The complexity of the dataset can be read along the number of levels of the dataset and the distribution of itemsets w.r.t. frequency at each level. The most interesting are the "lower" levels, i.e., the levels with the lowest closure index, as they usually include itemsets with high frequency, contrasting the higher levels which contain itemsets with a quite low frequency. Indeed, short minimum keys or passkeys correspond to implications in the related equivalence class with minimal left-hand side (LHS) and maximal right-hand side (RHS), which are the most informative implications [4, 5].

In this paper we discuss alternative ways of defining the "complexity" of a dataset and how it can be measured in the related concept lattice that can be computed from this dataset. For doing so, we introduce two main indicators, namely (i) the probability that two concepts C_1 and C_2 are comparable, (ii) given two intents A and B, the probability that the union of these two intents is again an intent. The first indicator "measures" how close is the lattice to a chain, and the second indicator "measures" how close the lattice is to a distributive one [6, 7]. Indeed, a distributive lattice may be considered as less complex than an arbitrary lattice, since, given two intents A and B, their meet $A \cap B$ and their join $A \cup B$ are also intents. Moreover, in a distributive lattice, all pseudo-intents are of size 1, meaning that every implication in the Duquenne-Guigues base has a premise of size 1. Following the same line, given a set of n attributes, the Boolean lattice $\wp(n)$ is the largest lattice that one can build from a context of size $n \times n$, but $\wp(n)$ can also be considered

as a simple lattice, since it can be represented by the set of its n atoms. In addition, the Duquenne-Guigues implication base is empty, so there are no nontrivial implications in this lattice. Finally, a Boolean lattice is also distributive, thus it is simple in terms of the join of intents.

This paper presents an original and practical study about the complexity of a dataset through an analysis of specific elements in the related concept lattice, namely intents, pseudo-intents, proper premises, keys, and passkeys. Direct links are drawn with implications and association rules, making also a bridge between the present study in the framework of FCA, and approaches more related to data mining, actually pattern mining and association rule discovery. Indeed, the covering relation of the concept lattice makes a concise representation of the set of association rules of the context [4, 5], so that every element of the covering relation, i.e., a pair of neighboring concepts or edge of the concept lattice, stays for an association rule, and reciprocally, every association rule can be given by a set of such edges. Frequency distribution of confidence of the edges can be considered as an important feature of the lattice as a collection of association rules.

For studying practically this complexity, we have conducted a series of experiments where we measure the distribution of the different elements for real-world datasets and then for related randomized datasets. Actually these randomized datasets are based on corresponding real-world datasets where either the distribution of crosses in columns is randomized or the whole set of crosses is randomized while keeping the density of the dataset. We can observe that randomized datasets are usually more complex in terms of our indicators than real-world datasets. This means that, in general, the set of "interesting elements" in the lattice is smaller in real-world datasets.

This paper is an extended and revised version of a paper [8] presented at "Concept Lattices and Their Applications" (CLA 2022) Conference located in Tallinn. This version extends the preceding in various directions and proposes:

- a focus on dataset complexity based on FCA,
- alternative definitions of the linearity and distributivity indices, and a new index namely the nonlinear distributivity index,
- a complexity study of the computing of indices,
- an experimental study of the relations that indices have one with the other,

- more experiments and as well as a revised and more concise description of these experiments and their results.

The paper is organized as follows. In the second section we introduce the theoretical background and necessary definitions. Then the next section presents a range of experiments involving real-world and randomized datasets. Finally, the results of experiments are discussed and then we propose a conclusion.

2. Theoretical Background

2.1. Classes of Characteristic Attribute Sets

Here we recall basic FCA definitions related to concepts, dependencies, and their minimal representations. After that we illustrate the definitions with a toy example. Let us consider a formal context $K = (G, M, I)$ and prime operators:

$$A' = \{m \in M \mid \forall g \in A : gIm\}, \quad A \subseteq G \tag{1}$$
$$B' = \{g \in G \mid \forall m \in B : gIm\}, \quad B \subseteq M \tag{2}$$

We illustrate the next definitions using an adaptation of the "four geometrical figures and their properties" context [9] which presented in Table 1. The set of objects $G = \{g_1, g_2, g_3, g_4\}$ corresponds to {equilateral triangle, rectangle triangle, rectangle, square}) and the set of attributes $M = \{a, b, c, d, e\}$ corresponds to {has 3 vertices, has 4 vertices, has a direct angle, equilateral, e} ("e" is empty and introduced for the needs of our examples). The related concept lattice is shown in Figure 1.

Definition 2.1 (Intent or closed description)**.** A subset of attributes $B \subseteq M$ is an intent or is closed iff $B'' = B$.

In the running example (Table 1), $B = \{b, c\} = B''$ is an intent and is the maximal subset of attributes describing the subset of objects $B' = \{g_3, g_4\}$.

Definition 2.2 (Pseudo-intent)**.** A subset of attributes $P \subseteq M$ is a pseudo-intent iff:

1. $P \neq P''$
2. $Q'' \subset P$ for every pseudo-intent $Q \subset P$

Pseudo-intents are premises of implications of the cardinality-minimal implication basis called "Duquenne-Guigues basis" [3] (DG-basis, also known as "canonical basis" or "stembase" [1]). In the current example (Table 1), the set of pseudo-intents is $\big\{\{b\}, \{e\}, \{c,d\}, \{a,b,c\}\big\}$ since: (i) $\{b\}, \{e\}, \{c,d\}$ are minimal non-closed subsets of attributes, and (ii) $\{a,b,c\}$ is both non-closed and contains the closure $\{b,c\}$ of the pseudo-intent $\{b\}$.

	a	b	c	d	e
g_1	x			x	
g_2	x		x		
g_3		x	x		
g_4		x	x	x	

Table 1: The adapted context of geometrical figures [9].

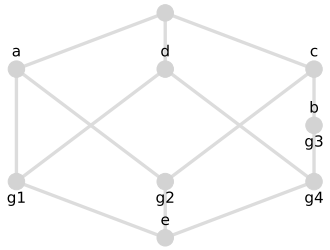

Figure 1: The corresponding lattice of geometrical figures.

Definition 2.3 (Proper premise)**.** A set of attributes $A \subseteq M$ is a proper premise iff:

$$A \cup \bigcup_{n \in A} (A \setminus \{n\})'' \neq A''$$

In the running example (Table 1), $Q = \{a,b\}$ is a proper premise since the union of Q with the closures of its subsets does not result in the closure of Q, i.e., $\{a,b\} \cup \{a\}'' \cup \{b\}'' = \{a,b\} \cup \{a\} \cup \{b,c\} = \{a,b,c\} \neq \{a,b,c,d,e\}$.

Proper premises are premises of the so-called "proper-premise base" (PP-base, see [1, 10]) or "direct canonical base" [11, 12]. The PP-base is a "direct" or "iteration-free base of implications", meaning that we can obtain all possible implications with a single application of Armstrong rules to implications in PP-base.

Definition 2.4 (Generator)**.** A set of attributes $D \subseteq M$ is a generator iff $\exists B \subseteq M : D'' = B$.

In this paper, every subset of attributes is a generator of a concept intent. A generator is called non-trivial if it is not closed. In the current example (Table 1), $D = \{a,b,d\}$ is a generator of $B = \{a,b,c,d,e\}$ since B is an intent, $D \subseteq B$, and $D'' = B$.

Definition 2.5 (Minimal generator, key)**.** A set of attributes $D \subseteq M$ is a key or a minimal generator of D'' iff $\nexists m \in D : (D \setminus \{m\})'' = D''$.

In the following we will use "key" rather than "minimal generator. A key is inclusion minimal in the equivalence class of subsets of attributes having the same closure [4, 5]. In the current example (Table 1), $D = \{a, c, d\}$ is a key since none of its subsets $\{a, c\}, \{a, d\}, \{c, d\}$ generates the intent $D'' = \{a, b, c, d, e\}$. Every proper premise is a key, however the converse does not hold in general.

Definition 2.6 (Minimum generator, passkey)**.** A set of attributes $D \subseteq M$ is a passkey or a minimum generator iff D is a minimal generator of D'' and D has the minimal size among all minimal generators of D''.

In the following we will use "passkey" rather than "minimum generator. A passkey is cardinality-minimal in the equivalence class of subsets of attributes having the same closure. It should be noticed that there can be several minimum generators and one is chosen as a passkey, but the minimal size is unique. In [2] the maximal size of a passkey of a given context was studied as an index of the context complexity. In the current example (Table 1), $D = \{b, d\}$ is a passkey of the intent $\{b, c, d\}$ since there is no other generator of smaller cardinality generating D''. Meanwhile $D = \{a, c, d\}$ is not a passkey of $D'' = \{a, b, c, d, e\}$ since the subset $E = \{e\}$ has a smaller size and the same closure, i.e., $E'' = D''$.

Finally, for illustrating all these definitions, we form the context $(2^M, M_d, I_d)$ of all classes of "characteristic attribute sets" of M as they are introduced above. $M_d = \{\text{intent}, \text{pseudo-intent}, \text{proper premise}, \text{key}, \text{passkey}\}$, while I_d states that a given subset of attributes in 2^M is a characteristic attribute set in M_d. The concept lattice of this context is shown in Figure 2.

2.2. Towards Measuring Data Complexity

"Data complexity" can mean many different things depending on the particular data analysis problem under study. For example, data can be claimed to be complex when data processing takes a very long time, and this could be termed as "computational complexity" of data. Alternatively, data can be considered as complex when data are hard to analyze and to interpret, and this could be termed as "interpretability complexity" of data. For example, it can be hard to apply FCA to very large datasets as the size of the resulting concept lattice may be too too large, while in many situations

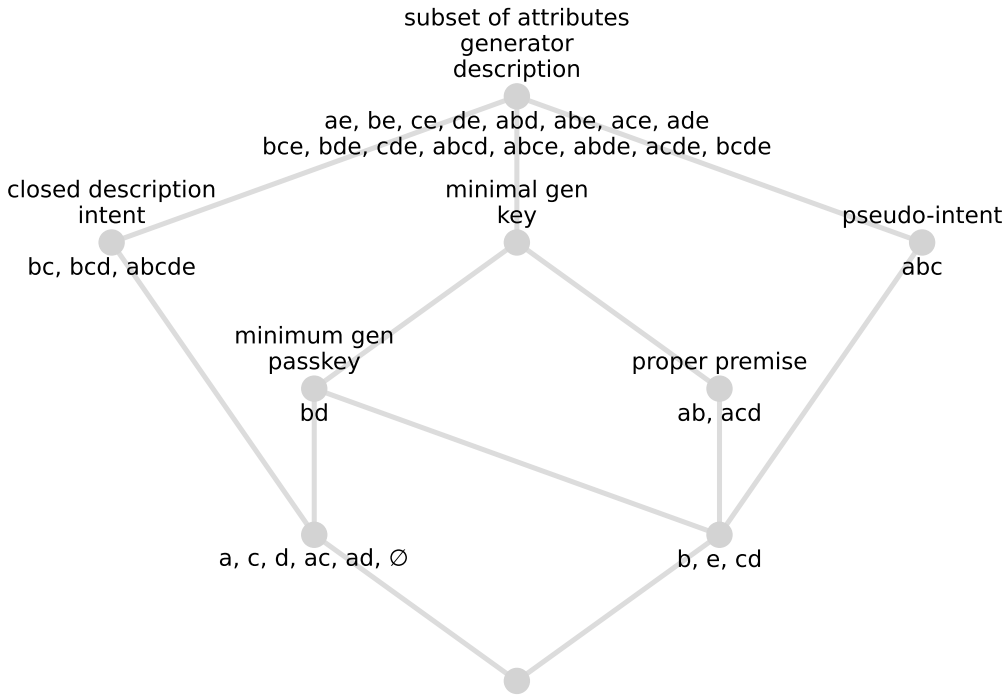

Figure 2: The concept lattice of "characteristic attribute sets" of the context introduced in Table 1.

the results of machine learning algorithms are not explainable as expected and give rise to to XAI research lines. Accordingly, it is quite hard to define data complexity in general terms.

If we consider the dimension of interpretability, then the size of the "patterns" to interpret and their number are definitely important elements to take into account. In the following, the expression "pattern" refers to "interesting subsets of attributes" in a broad sense. In an ideal case, one prefers a small number of (interesting) patterns to facilitate interpretation. Indeed, a small number of rules with a few attributes in the premises and in the conclusions is simpler to interpret than hundreds of rules with more than ten attributes in the premises and conclusions. Thus, it is natural to study how the number of patterns is distributed w.r.t. their size. In most of the cases, large numbers of patterns are associated with computational complexity. Then controlling the size and the number of patterns is also a way to control computational complexity.

It should also be mentioned that the number of patterns is related to the so-called "VC-dimension" of a context [13], i.e., the maximal size of a Boolean sublattice generated from the context. Accordingly, in this study about data complexity, we decided to count the number of concepts, pseudo-intents, proper premises, keys, and passkeys, in order to understand and evaluate the complexity of data. For all these "pattern types", we also study the distribution of pattern sizes.

Additionally, we decided to measure the "lattice complexity", i.e., the complexity of the corresponding concept lattice, with two new measures related to what could be termed the "linearity" of the lattice. Indeed, the simplest lattice structure that can be imagined is a chain, while the counterpart is represented by the Boolean lattice, i.e., the lattice with the largest amount of connections and concepts, as any combination of attributes is admissible as a closed set. However, it should be noticed that the Boolean lattice may be considered as complex if one has to interpret all combinations of attributes, while, by contrast, it can be considered as simple from the point of view of the implication base, which is empty in such a lattice.

Then, a first way to measure the closeness of a concept lattice to a chain is the "linearity index" which is defined below as the probability that two random concepts are comparable in the lattice.

Definition 2.7. Given a lattice of intents $\mathcal{L}$, the linearity index of $\mathcal{L}$ is defined as the fraction of comparable pairs of intents of the lattice $\mathcal{L}$ w.r.t. the number of all pairs of elements:

$$\texttt{LIN}(\mathcal{L}) = \frac{\left|\left\{(B_1, B_2) \in \mathcal{L}^2 \mid B_1 \subseteq B_2 \text{ or } B_2 \subseteq B_1\right\}\right|}{\left|\mathcal{L}\right|^2}. \tag{3}$$

Let us introduce the following notations for making precise the practical computation of the linearity index:

(i) Let us consider a concept lattice $\mathcal{L}$ with $N = |\mathcal{L}|$ concepts;

(ii) Let us suppose that we have an ordering of the concepts in $\mathcal{L}$;

(iii) Let us define $\mathbb{1}(c_i \leq c_j) = 1$ if $i = j$ or $c_i < c_j$ and 0 otherwise, where $\leq$ is the order between concepts in $\mathcal{L}$ and $\mathbb{1}$ is the indicator function taking the value 1 when the related constraint is true.

Based on these notations, definition 2.7 can be rewritten in the following way which shows how to compute the linearity index $\texttt{LIN}(\mathcal{L})$:

$$\texttt{LIN}(\mathcal{L}) = \begin{cases} \frac{2}{N\cdot(N-1)} \sum\limits_{(1\leq i\leq j\leq N)} \mathbb{1}(c_i \leq c_j) & \text{if } N > 1, \\ 1 & \text{if } N = 1 \end{cases} \tag{4}$$

The linearity index is maximal for a chain, i.e., the lattice related to a linear order (see Figure 3). It is minimal for the lattice related to a nominal scale which is also the lattice related to a bijection. One example of such a lattice is given by the so-called M3 lattice which includes a top and a bottom element, and three incomparable elements. In particular, when a lattice includes a sublattice such as M3 it is not distributive [6, 7].

However, this index does not directly measure how well the lattice is interpretable. One of the main interpretability properties is the size of some particular sets, such as the size and the structure of the implication basis. Then the simplest structure supporting the implication basis can be found in distributive lattices, where pseudo-intents are all of size 1. Accordingly, the "distributivity index" measures how a lattice is close to a distributive one. For that we check the probability of building an intent when joining two other intents.

Definition 2.8. Given a lattice of intents $\mathcal{L}$, the distributivity index $\texttt{DIST}(\mathcal{L})$ is defined as the fraction of pairs of closed descriptions from lattice $\mathcal{L}$, s.t. their union also lies in lattice $\mathcal{L}$:

$$\texttt{DIST}(\mathcal{L}) = \frac{\left|\left\{(B_1, B_2) \in \mathcal{L}^2 \mid B_1 \cup B_2 \in \mathcal{L}\right\}\right|}{|\mathcal{L}^2|}. \tag{5}$$

Below we make precise a practical way of computing the distributivity index:

(iv) Let us define $\mathbb{1}(k_i \cup k_j) = 1$ if the union of the intents k_i and k_j is an intent of $\mathcal{L}$,and 0 otherwise.

Based on the above notations, definition 2.8 can be rewritten in the following way which shows how to compute the distributivity index $\texttt{DIST}(\mathcal{L})$:

$$\texttt{DIST}(\mathcal{L}) = \begin{cases} \frac{2}{N\cdot(N-1)} \sum\limits_{(1\leq i\leq j\leq N)} \mathbb{1}(k_i \cup k_j) & \text{if } N > 1, \\ 1 & \text{if } N = 1. \end{cases} \tag{6}$$

The distributivity index is maximal for distributive lattices, and this includes chain lattices which are distributive lattices [6, 7] (see Figure 3). Again it is minimal for lattices of nominal scales which are not distributive. Although, it may sound strange to consider the lattices of nominal scales as complex, they are not simple from the viewpoint of implications. For example, any pair of attributes from the M3 lattice –introduced above– can form the premise of an implication with a non-empty conclusion. This indeed introduces many implications in the basis and makes the DG-basis hard to interpret.

Note that if two closed descriptions $A, B \in \mathcal{L}$ are comparable (e.g. $A \subseteq B$) then their union (in our example $A \cup B = B$) is also a closed description from the lattice $\mathcal{L}$. Therefore, all pairs of descriptions accounted in linearity index are also accounted in distributivity index. To make the two indices less correlated we introduce the nonlinear distributivity index.

Definition 2.9. Given a lattice of closed descriptions $\mathcal{L}$, the nonlinear distributivity index $\texttt{NL.DIST}(\mathcal{L})$ is defined as the fraction of pairs of closed descriptions from lattice $\mathcal{L}$, such that they are not comparable, yet their union lies in lattice $\mathcal{L}$:

$$\begin{aligned}\texttt{NL.DIST}(\mathcal{L}) &= \texttt{DIST}(\mathcal{L}) - \texttt{LIN}(\mathcal{L}) \\ &= \frac{\left|\left\{(B_1, B_2) \in \mathcal{L}^2 \mid B_1 \cup B_2 \in \mathcal{L}, B_1 \not\subseteq B_2, B_2 \not\subseteq B_1\right\}\right|}{|\mathcal{L}^2|}. \end{aligned} \quad (7)$$

Note, that the complexity of computing distributivity by the standard definition over triples, when one has an oracle giving $\vee$ and $\wedge$ of the lattice in O(1) time or an $L \times L$-table with operation results is $\mathrm{O}(L^3)$, where L is the size of the lattice (i.e., the number of concepts). This is not realistic in our setting, since we do not have a powerful lattice hardware and software with O(1) time for computing lattice operations. So to check distributivity by comparing, say, intents of concepts in the left side and right side of the definition of distributivity, one can compute $A \cap B$ in O(1) or $\mathrm{O}(M)$ time for providing extent of $(A, B) \wedge (C, D)$, but to compute $(A, B)v \vee (C, D)$ one needs to compute closure $(A \cup B)''$ (intersecting extents is not enough, since we agreed above to compare by intents). So, the total complexity of checking distributivity by standard definition will be $\mathrm{O}(L^3 \times |G| \times |M|)$, where $\mathrm{O}(|G| \times |M|)$ is the time for computing closure $(\cdot)''$. So, we get the same factor $\mathrm{O}(|G| \times |M|)$, as in the case of computing our distributivity indices

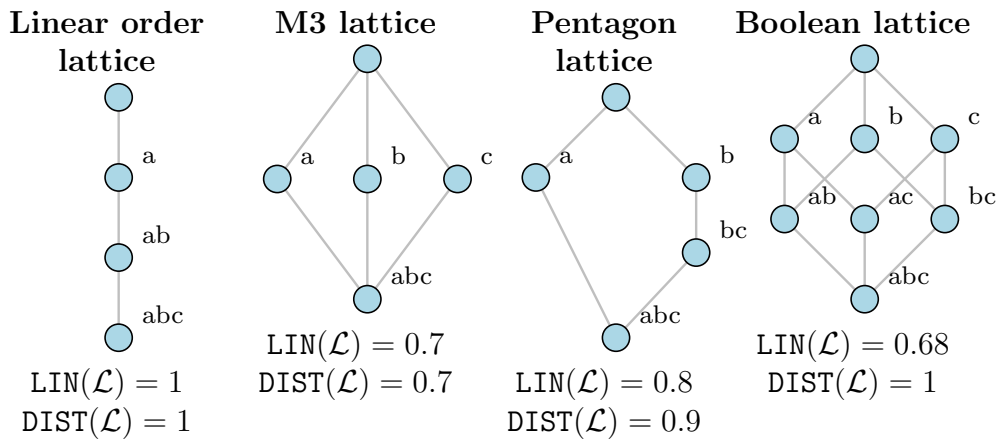

Figure 3: Four examples of lattices and the respective values of the LIN and DIST indices.

(5)-(7), and we can say that computing distributivity indices according to our definitions (5)-(7) in $\mathrm{O}(L^2 \times |G| \times |M|)$ time can be much faster in practice than computing distributivity index in $\mathrm{O}(L^3 \times |G| \times |M|)$ time.

Finally, introduce one other way to compute data complexity based on the lattice diagram.

Definition 2.10. Given lattice $\mathcal{L}$, edge density of lattice $\mathcal{L}$ is defined as the number of edges in the graph representation of the lattice w.r.t. the maximal number of edges $|\mathcal{L}|(\mathcal{L}-1)$.

2.3. Synthetic Complex Data

In order to study some ways of measuring data complexity, we need to compare the behavior of different complexity indices for "simple" and "complex" data. However, beforehand we cannot know which dataset is complex. Accordingly, we will generate synthetic complex datasets and compare them with real-world datasets. One way of generating complex data is "randomization". This idea is justified, e.g., by the theory of Kolmogorov complexity. Actually, randomized data cannot be well-interpreted since any possible result is an artifact of the method. For randomized data we know beforehand that there cannot exist any rule or concept that have some meaning. Thus, randomized data are good candidate data for being considered as "complex".

Now we discuss which randomization strategy should be used for generating such data. Assume we are given a formal context (G, M, I) and we want

to compare it to the random context $(\tilde{G}, \tilde{M}, \tilde{I})$. In order to make the two contexts comparable, we equate their respective sets of objects $G = \tilde{G}$, sets of attributes $M = \tilde{M}$, and the cardinalities of relations $|I| = |\tilde{I}|$. Thus, the only difference between the contexts lies in the fact that relation I is taken from the real-word and relation $\tilde{I}$ is a random subset of pairs of objects G and attributes M.

We also want to see the gradual change of the context complexity indices with the increasing randomization while keeping the number of connections the same. To achieve this we generate relation $\tilde{I}$ as a random subset of pairs $G \times M \setminus I$ or size $|I|$. Then we assign random indices from 1 to $|I|$ to each element of $I = \{(g,m)_i\}_{i=1}^{|I|}$ and to each element from $\tilde{I} = \{\widetilde{(g,m)}_i\}_{i=1}^{|I|}$. Finally, to evaluate the complexity of a context with α percent randomization, we construct a new relation I_α consisting of α percent of pairs from I and $100 - \alpha$ percent of pairs from $\tilde{I}$: $I_\alpha = \{(g,m)_i\}_{i=1}^{\alpha|I|} \cup \{\widetilde{(g,m)}_i\}_{i=\alpha|I|}^{|I|}$. Therefore, relation I_0 with $\alpha = 0$ percent randomization would be equal to the real-world relation I, relation I_{100} with $\alpha = 100$ percent randomization would be equal to the random relation $\tilde{I}$, and relation I_{50} would be one-half random and one-half real-world. Note that for every α cardinality $|I_\alpha|$ equals to cardinality $|I|$, as randomized relation $\tilde{I}$ is a subset of $G \times M \setminus I$.

In the next section we study different ways of measuring the complexity of a dataset and we observe that the complexity of randomized datasets is generally higher than the complexity of the corresponding real-world dataset.

3. Experiments

3.1. Datasets

For this study we selected 14 real-world from LUCS-KDD repository[1].

We use only a half of the datasets from the repository as the omitted ones take too long time to compute when randomized. Table 2 provides the information about the contexts used in the experiments.

For each real world context we construct one hundred random contexts. Then, for each pair of real-world relation I and its random counterpart $\tilde{I}$ we evaluate the gradual change of complexity indices on mixed relations

[1]Coenen, F. (2003), The LUCS-KDD Discretised/normalised ARM and CARM Data Library, `http://www.csc.liv.ac.uk/~frans/KDD/Software/LUCS_KDD_DN/`, Department of Computer Science, The University of Liverpool, UK.

id	context	# rows $\lvert G\rvert$	# columns $\lvert M\rvert$	# connections $\lvert I\rvert$	density $\frac{\lvert I\rvert}{\lvert G\times M\rvert}$
1	zoo	101	43	1717	0.40
2	iris	150	20	750	0.25
3	wine	178	69	2492	0.20
4	glass	214	49	2140	0.20
5	heart	303	53	4236	0.26
6	ecoli	336	35	2688	0.23
7	dematology	366	50	4750	0.26
8	breast	699	21	6974	0.48
9	pima	768	39	6912	0.23
10	anneal	898	74	12847	0.19
11	ticTacToe	958	30	9580	0.33
12	flare	1389	40	15279	0.28
13	led7	3200	25	25600	0.32
14	pageBlocks	5473	47	60203	0.23

Table 2: The description of contexts used in the experiments

I_α where α takes values from the set $\{0, 2, 4, \ldots, 20, 24, \ldots, 40, 50, \ldots, 100\}$. The reason for choosing such logarithmic-like spacing of α values will become clear in the following figures.

3.2. Data Complexity

Clustering complexity indices. In the previous sections of the paper we discussed various indices to evaluate the complexity of the data. We have also noticed that some of them can be highly correlated: for example, linearity and distributivity indices. Thus, in this paragraph, we examine the empirical correlation between the complexity indices.

The heatmap on Figure 4 presents the Spearman rank correlation coefficients between various data complexity indices. The data underlying the correlations were obtained on all 14 contexts for all 22 values α. An exception is made for contexts (G, M, I_α) whose corresponding lattices of closed intents contain only one intent, namely M. Each index representing the mean size of one of characteristic attribute sets was normalized by the number of attributes $|M|$ of the respective context. The heatmap highlights the high correlation between the indices. This fact allows us to group the indices into

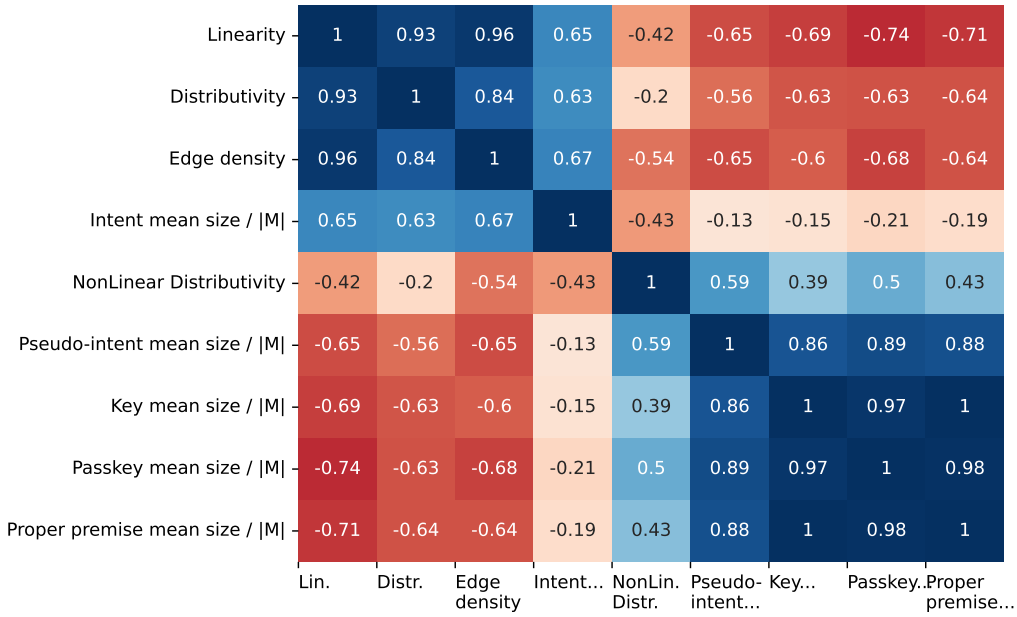

Figure 4: Correlations between various complexity indices

highly correlated clusters and to study only one representative index from each of the clusters.

Specifically, we group the indices into five clusters:

- {linearity, distributivity, edge density}
  (representative index: linearity);
- {nonlinear distributivity};
- {intent mean size (normalized by $|M|$)};
- {key mean size, passkey mean size, proper premise mean size} (all normalized by $|M|$)
  (representative index: key mean size (normalized by $|M|$));
- {pseudo-intent mean size (normalized by $|M|$)}.

It is interesting to notice that the last three proposed clusters follow the idea of a lattice in Figure 2. That is, all characteristic attribute sets can be divided into three parts: intents (the left branch of the figure), keys and their

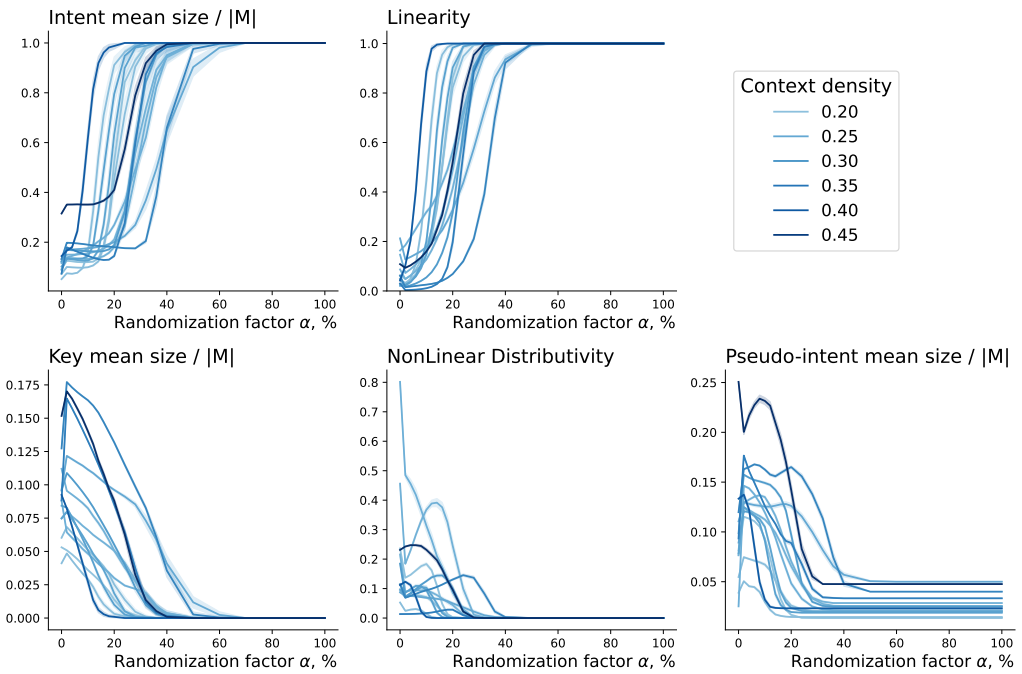

Figure 5: Change of complexity indices with the growth of randomization factor for each of the contexts. The plots only reflect the concepts with at least 10% support.

derivatives (the center of the figure), and pseudo-intents (the right branch of the figure).

Complexity indices with increasing randomization. Now let us study in details how different complexity indices react to data randomization.

Figure 5 describes how each complexity index cluster (defined above) reacts to data randomization and, thus, to increasing data complexity. Each subplot in the figure shows the behaviour of the representative index of each complexity index cluster. Each line in a subplot follows the average value of a complexity index over one hundred trials for one of the 14 formal contexts. The color intensity of each line depends on the density of a context.

The figure shows that each complexity index tends to some constant value as the randomization factor approaches 100%. The reason for this is that for each context with high randomization factor value we observe a concept lattice containing one single element: the concept with empty extent and the maximal intent M. In fact, we expect that such lattice should contain large

amount of concepts covering tiny subsets of objects. However, to make the experiments run in a reasonable time, we filter out all concepts that cover less than 10% of objects (i.e. their minimal support is 10%).

The top-left subplot in the figure shows that the mean size of the intents grows w.r.t. an increasing randomization factor of the context. However, the growth is not linear, but more quadratic or even combinatorial. That is, small levels of randomization (up to 20%) do not affect the mean intent size, but high levels (starting from 60%) make the mean intent size rise up to $|M|$ (i.e. the lattice of closed descriptions contains only one maximal description). The bottom-left subplot of the figure shows the behaviour of the mean key size. Contrasting the mean intent size, the mean key size decreases w.r.t. an increasing randomization. In other words, small values of mean key sizes correspond to complex data, while larger keys correspond to more real-world contexts.

Linearity index in the top-center subplot shows a nonlinear dependency. It is close to zero for both real-world and random contexts. However, it rapidly increases for real-world contexts with a small level of added noise. The other index –NonLinear Distributivity– often resembles quadratic functions. That is, it slightly grows with a small randomization factor, but then tends to zero with bigger randomization factors. Finally, the mean size of pseudo-intents works similarly to mean keys sizes and nonlinear distributivity: they non-monotonically decrease with a growing randomization.

4. Conclusion

In this paper we have studied various definitions of data complexity based on FCA, like number and average size of intents, keys (minimal generators), passkeys (minimum generators), proper premises, and pseudo-intents. We have introduced new indices for measuring the complexity of a dataset, among which the linearity index for checking the direct dependencies between concepts or how a concept lattice is close to a chain, and the distributivity lattice which measures how close is a concept lattice to a distributive lattice. In a distributive lattice, all pseudo-intents are of length 1, leading to sets of simple implications. We have also proposed a series of experiments where we analyze real-world datasets and their randomized counterparts. As expected, the randomized datasets are more complex that the real-world ones.

Future work will be to improve this study in several directions, (i) studying more deeply the role of indices, especially the linearity index and the

distributivity index, and the relations inter indices, (ii) analyzing larger datasets, and more importantly (iii) analyzing the complexity from the point of view of the generated implications and association rules.

This paper proposes a meaningful step in the analysis of the dataset complexity in the framework of FCA. We indeed believe that FCA brings a significant support for analyzing data complexity in general.

References

[1] B. Ganter, R. Wille, Formal Concept Analysis, Springer, Berlin, 1999.

[2] T. Makhalova, A. Buzmakov, S. O. Kuznetsov, A. Napoli, Introducing the closure structure and the GDPM algorithm for mining and understanding a tabular datasets, International Journal of Approximate Reasoning 145 (2022) 75–90.

[3] J.-L. Guigues, V. Duquenne, Famile minimale d'implications informatives resultant d'un tableau de données binaire, Mathematique, Informatique et Sciences Humaines 95 (1986) 5–18.

[4] N. Pasquier, Y. Bastide, R. Taouil, L. Lakhal, Pruning Closed Itemset Lattices for Association Rules, International Journal of Information Systems 24 (1) (1999) 25–46.

[5] Y. Bastide, R. Taouil, N. Pasquier, G. Stumme, L. Lakhal, Mining frequent patterns with counting inference, SIGKDD Exploration Newsletter 2 (2) (2000) 66–75.

[6] B. A. Davey, H. A. Priestley, Introduction to Lattices and Order, Cambridge University Press, Cambridge, UK, 1990.

[7] G. Grätzer, General Lattice Theory (Second Edition), Birkäuzer, 2002.

[8] A. Buzmakov, E. Dudyrev, S. O. Kuznetsov, T. Makhalova, A. Napoli, Experimental Study of Concise Representations of Concepts and Dependencies, in: P. Cordero, O. Krídlo (Eds.), Proceedings of the Sixteenth International Conference on Concept Lattices and Their Applications (CLA), CEUR Workshop Proceedings 3308, CEUR-WS.org, 2022, pp. 117–132.

[9] S. O. Kuznetsov, S. A. Obiedkov, Comparing performance of algorithms for generating concept lattices, Journal of Experimental & Theoretical Artificial Intelligence 14 (2/3) (2002) 189–216.

[10] U. Ryssel, F. Distel, D. Borchmann, Fast algorithms for implication bases and attribute exploration using proper premises, Annals of Mathematics and Artificial Intelligence 70 (1-2) (2014) 25–53.

[11] K. Bertet, B. Monjardet, The multiple facets of the canonical direct unit implicational basis, Theoretical Computer Science 411 (22-24) (2010) 2155–2166.

[12] B. Ganter, S. A. Obiedkov, Conceptual Exploration, Springer, 2016.

[13] A. Albano, B. Chornomaz, Why concept lattices are large: extremal theory for generators, concepts, and VC-dimension, International Journal of General Systems 46 (5) (2017) 440–457.