Experimental Study of Concise Representations of Concepts and Dependencies

Aleksey Buzmakov[1], Egor Dudyrev[2], Sergei O. Kuznetsov[2], Tatiana Makhalova[3] and Amedeo Napoli[3,1,*]

[1]*HSE University, Perm, Russia*

[2]*HSE University, Moscow, Russia*

[3]*Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France*

Abstract

In this paper we are interested in studying concise representations of concepts and dependencies, i.e., implications and association rules. Such representations are based on equivalence classes and their elements, i.e., minimal generators, minimum generators including keys and passkeys, proper premises, and pseudo-intents. All these sets of attributes are significant and well-studied from the computational point of view, while their statistical properties remain to be investigated. This is the purpose of this paper to study these singular attribute sets and in parallel to study how to evaluate the complexity of a dataset from an FCA point of view. In the paper we analyze the empirical distributions and the sizes of these particular attribute sets. In addition we propose several measures of data complexity relying on these attribute sets in considering real-world and related randomized datasets.

Keywords

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

1. Introduction

In this paper we are interested in measuring "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 and 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 contain 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.

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 [8] 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 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 (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 $\{\{b\}, \{e\}, \{c, d\}, \{a, b, c\}\}$ 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	

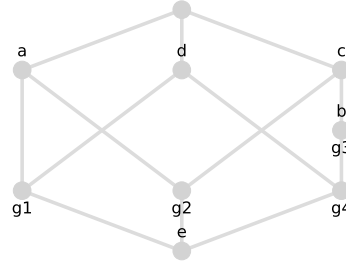Table 1: The adapted context of geometrical figures [8].

Figure 1: The corresponding lattice of geometrical figures.

Definition 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, 9]) or "direct canonical base" [10, 11]. 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 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 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 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. $M_d = \{\text{intent, pseudo-intent, proper premise, key, passkey}\}$, 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.
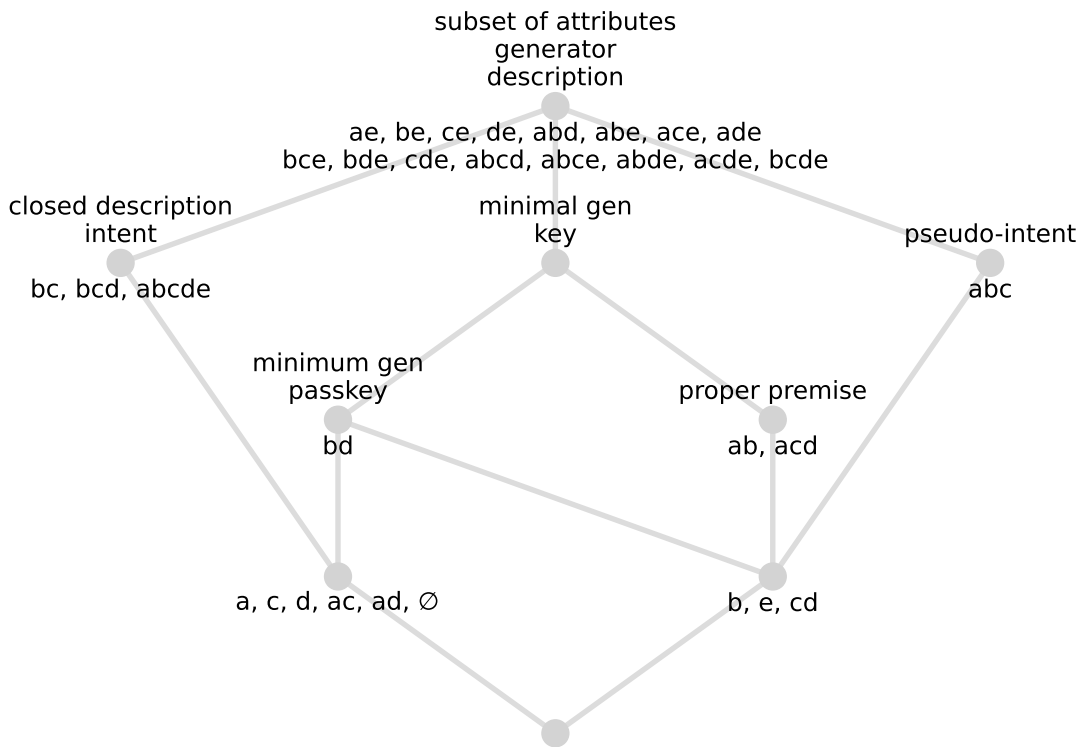

Figure 2: The concept lattice of "characteristic attribute sets" of the context introduced in Table 1.

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 or machine learning algorithms, such as clustering, classification, or regression. 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 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 [12], 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. However, it should be noticed that the Boolean lattice may be considered as complex from the point of view of interpretability, but very simple from the point of view of implication base, which is empty in such a lattice.

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

Definition 7. *Let us consider a concept lattice $\mathscr{L}$, $\mathscr{L}^* = \mathscr{L} \setminus \{\top, \bot\}$, where $|\mathscr{L}|$ denotes the number of the concepts in $\mathscr{L}$, $\top$ and $\bot$ being the top and bottom elements of $\mathscr{L}$. The linearity index $\mathrm{LIN}(\mathscr{L})$ is defined as:*

$$\mathrm{LIN}(\mathscr{L}) = \begin{cases} \frac{2}{|\mathscr{L}^*| \cdot (|\mathscr{L}^*|-1)} \sum_{\{c_i, c_j\} \subset \mathscr{L}^*} \mathbb{1}(c_i < c_j \vee c_i > c_j) \; if \, |\mathscr{L}| > 3, \\ 1, \; otherwise \end{cases} \tag{3}$$

where $\{c_i, c_j\}$ is an unordered pair of concepts, and $\mathbb{1}$ is the indicator function taking the value 1 when the related constraint is true.

The linearity index is maximal for a chain, i.e., the lattice related to a linear order. 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].

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. One of the simplest structure for the implication basis can be found in distributive lattices, where pseudo-intents are all of size 1 [6]. Accordingly, the "distributivity index" measures how a lattice is close to a distributive one.

Definition 8. *Given a lattice $\mathscr{L}$, the distributivity index* $\mathrm{DIST}(\mathscr{L})$ *is defined as*

$$\mathrm{DIST}(\mathscr{L}) = \frac{2}{|\mathscr{L}| \cdot (|\mathscr{L}| - 1)} \sum_{\{k_i, k_j\}_{i \neq j} \subset Intents(\mathscr{L})} \mathbb{1}(k_i \cup k_j \in Intents(\mathscr{L})), \tag{4}$$

where $Intents(\mathscr{L})$ is the set of concept intents in $\mathscr{L}$.

The distributivity index is maximal for distributive lattices, and this includes chain lattices which are distributive lattices [6, 7]. 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 this makes the DG-basis hard to interpret.

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". Actually, randomized data cannot be 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. A natural way is making randomized data similar in a certain sense to the real-world data they are compared to. Firstly, when considering reference real-world data, it seems natural to keep the same number of objects and attributes as they are in the real-world data. Moreover, the "density" of the context, i.e., the number of crosses, significantly affects the size and the structure of the related concept lattice. Thus, to ensure that randomized data are "similar" to the real-world data it is also natural to keep the density of data. This gives us the first randomization strategy, i.e., for any real-world dataset we can generate a randomized dataset with the same number of objects and attributes, and with the same density. Then, the crosses in the original context will be randomly spread along the new context in ensuring that the density is the same as in the original real-world data.

For example, let us consider the context given in Table 2, where there are 8 objects, 6 attributes, and 35 crosses. Thus, any context with 8 objects, 6 attributes, and 35 crosses, can be considered as a randomization of this original context. In our first randomization strategy, we suppose that the probability of generating any such randomized context is equally distributed.

descriptions	generator	intent	key	passkey	pseudo intent	proper premise
67	x	x	x	x		
45	x	x				
41	x		x	x	x	x
125	x		x	x		x
1	x		x		x	x
25	x		x			x
33	x				x	
1048239	x					

Table 2

The context corresponding to the lattice of descriptions for Bob Ross dataset (https://datahub.io/five-thirty-eight/bob-ross).

The randomized formal contexts for such strategy were studied in [13]. The authors have shown that the correlation between the number of concepts and the number of pseudo-intents has a non-random structure, suggesting that fixing density is not enough to generate randomized data which are similar to the real-world ones. Accordingly, we also studied a randomization strategy that keeps the number of objects as follows. A randomized context is generated attribute by attribute. The number of crosses in every column remains the same as in the corresponding "real-world attribute" but the crosses are randomly assigned. This can be viewed as a permutation of the crosses within every column in the randomized context, every column being permuted independently of the others. Such a procedure corresponds to the "null hypothesis" in statistical terms of independence between attributes.

Although such randomization strategy considers objects and attributes differently, it corresponds to typical cases in data analysis. Indeed, in typical datasets, objects stand for observations that are described by different attributes. The attributes correspond to any hypothesis in the data domain. Then, analysts are usually interested in discovering some relations between attributes, and the hypothesis of attribute independence is a natural null hypothesis in such a setting.

For example, let as consider again the context in Table 2, and that the numbers of objects and attributes remain the same in the randomized context. Then a randomized context following the second strategy is any context having 8 crosses for the first attribute, 2 crosses for the second attribute, 5 crosses for the third attribute, etc. Accordingly, when we are randomizing data from a given real-world dataset, we should have in mind that many randomized datasets can be generated w.r.t. the same randomization strategy. Thus, for the sake of objectivity, it is not enough to study only one random dataset for a given real-world dataset, but it is necessary to generate several randomized datasets and then to estimate the distribution of a characteristic under study within all randomized datasets.

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 preliminary study we selected 4 small real-world datasets in order to support an efficient computing of all necessary elements of the lattice. Efficiency matters here because we involve randomization and computations are repeated hundreds of times for one dataset. The study includes the following datasets: "Live in water[1]", "Tea ladies[2]" "Lattice of lattice properties[3]", and "Bob Ross episodes[4]". For the sake of efficiency the fourth dataset was restricted to only first the 20 attributes.

The datasets are analyzed in two different ways. Firstly, the characteristic attribute sets are computed, e.g., concepts, keys, passkeys, pseudo-intents, and proper premises, and then the relations existing between these elements are discussed. Secondly, we study the complexity of a dataset in studying its characteristic attribute sets and their distributions w.r.t. the size of these attribute sets. We also compared all the two numerical indicators, namely the linearity and the distributivity indices, for real-world and randomized datasets.

3.2. Characteristic Attribute Sets in a Lattice

In this section we study the relations between the different characteristic attribute sets. The experiment pipeline reads as follows.

- Given a context $K = (G, M, I)$, we compute all possible "attribute descriptions", i.e., subsets of attributes in 2^M, and we check whether a description verifies some characteristic such as being an intent, a key, a passkey....
- Given $AD = \{gen, intent, key, passkey, pseudo\text{-}intent, proper\text{-}premise\}$, we construct the context $(2^M, AD, I_{AD})$ where I_{AD} is the relation indicating that a given set of attributes has a characteristic in AD.
- Finally, we construct the "lattice of descriptions" based on the context $(2^M, AD, I_{AD})$. This lattice shows the relations existing between all generators −subsets of attributes− in a given dataset.

The "description context" for the "Bob Ross" dataset is given in Table 2 and the corresponding lattice is shown in Figure 3. From the lattice we can check that any two classes of descriptions may intersect if this is not forbidden by their definition (e.g., a description cannot be both an intent and a pseudo-intent). Although such a lattice is computed for a particular dataset, this is the general lattice structure which is obtained most of the time. In some small datasets it may happen that some characteristic attribute sets are missing. For example, in the "Live in water" lattice, the properties of being a key, being a passkey, and being a proper premise, all coincide and collapse into one single node.

It is also very interesting to analyze the proportions of the sizes of classes of descriptions. For example, in the "Bob Ross" context restricted to 20 attributes, there are 2^{20} possible descriptions,

[1]https://upriss.github.io/fca/examples.html
[2]https://upriss.github.io/fca/examples.html
[3]https://upriss.github.io/fca/exampLes.html
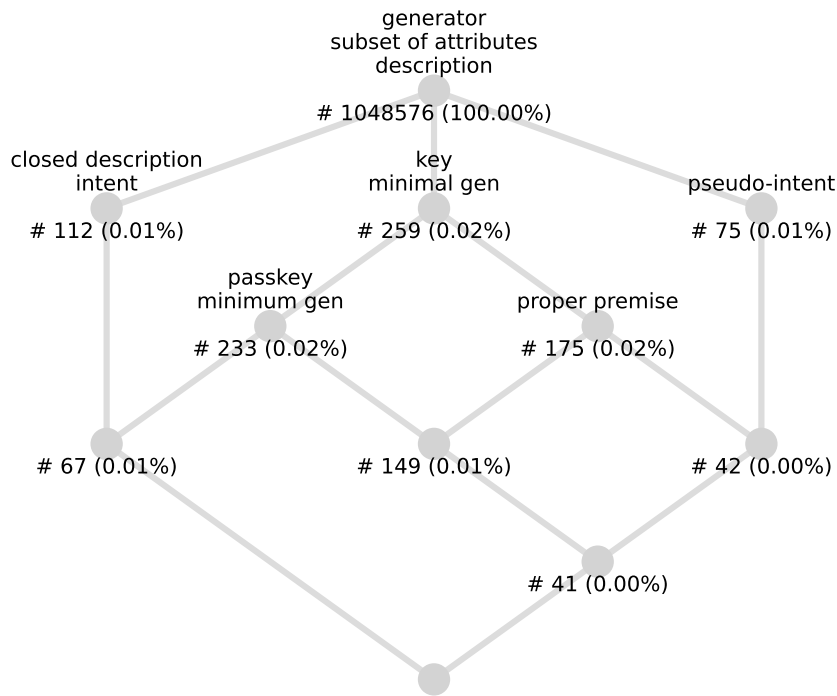[4]https://datahub.io/five-thirty-eight/bob-ross

Figure 3: The lattice of "attribute descriptions" for the "Bob Ross" dataset and their distributions.

but there are only 112 of them which are closed, and only 259 of them which are minimal generators. Thus, the large majority of the descriptions are "useless" in the sense that they do not correspond to any of the characteristic attribute subsets introduced above. In the next subsection we consider the distributions of these characteristic attribute subsets.

3.3. Data Complexity

For analyzing data complexity, we start by comparing the numbers of elements in real-world data and in randomized data. In the Figures 4 and 5, the distributions of the different characteristic elements for "Bob Ross" dataset are shown. Along the horizontal axis, the sizes of the elements are shown, i.e., the number of attributes in the intent, pseudo-intent, key, etc. Along the vertical axis the number of elements of the corresponding sizes are shown.

Red crosses show the values corresponding for real-world data and the boxplots visualize the values found in random data. There were 100 randomizations and thus boxplots are based on these 100 values. A box corresponds to the 50% middle values among 100 values. In addition, it should be noticed that these two figures differ in the randomization strategy. Figure 4 corresponds to density-based randomization while Figure 5 shows randomization based on column permutations.

From both figures we can observe that randomized data contain significantly larger numbers

of elements than the real-world data. Moreover, the sizes of the elements for randomized data are larger than the sizes of real-world data. Similar figures can be built for "Tea Ladies" and "Lattice of lattice properties" datasets. However, it is not possible to distinguish the real-world dataset and the randomized data for the "Live in Water" dataset[5]. This can be explained by the fact that either the dataset does not contain deep dependencies, or that the dataset is too small, i.e., the randomized dataset cannot be substantially different from the original one.

Let us now study how the linearity index and the distributivity index measure the complexity of a dataset. Figures 7 and 6 show the values of the linearity and distributivity indices correspondingly w.r.t. different randomizations. From these figures we can see that datasets built from density-based randomization are more different from real-world datasets than the randomized datasets built from column-wise permutations. We also notice that the values of the linearity and distributivity indices show a substantial dependence w.r.t. density of the corresponding context. Indeed, if we examine the randomized datasets, we can see that the distributions of the linearity and distributivity indices are different. This can be explained either by the context density and by the context size. Thus, we cannot have any reference values for these indices that would split between "complex" and not "simple" data. However, comparing the values of the index to the distribution of these indices allow one to decide on complexity of the data. Finally, in all datasets but "Live in water", both linearity and distributivity indices have higher values for real-world datasets than for randomized datasets. This shows again that real-world datasets are more structured than their randomized counterparts.

4. Conclusion

In this paper we have introduced and studied "concise representations" of datasets given by related contexts and concept lattices, and characteristic attributes sets based on equivalence classes, i.e., intents, keys (minimal generators), passkeys (minimum generators), proper premises, and pseudo-intents. We have also defined two new indices for measuring the complexity of a dataset, namely 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 the latter, 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.

The future work will be to improve this study in several directions, by studying more deeply the role of both indices, the linearity index and the distributivity index, by analyzing more larger datasets, and more importantly by analyzing the complexity from the point of view of the generated implications and association rules. This paper is another step in the analysis of the complexity of datasets in the framework of FCA, and we believe that FCA can bring a substantial support for analyzing data complexity in general.

---

[5]All the figures that cannot be shown in this paper are visible in supplementary materials, see https://yadi.sk/i/8_5EEvY4zNi82g

Acknowledgments

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 (1999) 25–46.

[5] Y. Bastide, R. Taouil, N. Pasquier, G. Stumme, L. Lakhal, Mining frequent patterns with counting inference, SIGKDD Exploration Newsletter 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] S. O. Kuznetsov, S. A. Obiedkov, Comparing performance of algorithms for generating concept lattices, Journal of Experimental & Theoretical Artificial Intelligence 14 (2002) 189–216.

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

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

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

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

[13] D. Borchmann, T. Hanika, Some Experimental Results on Randomly Generating Formal Contexts, in: M. Huchard, S. O. Kuznetsov (Eds.), Proceedings of the 13th International Conference on Concept Lattices and Their Applications (CLA), volume 1624 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2016, pp. 57–69.
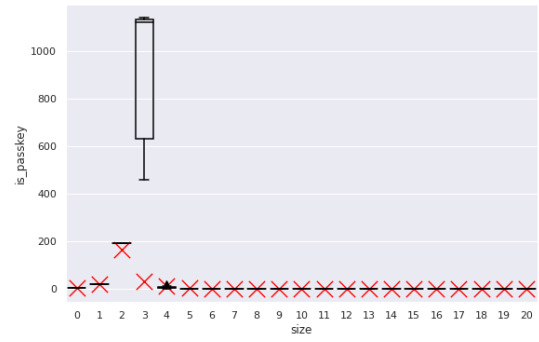
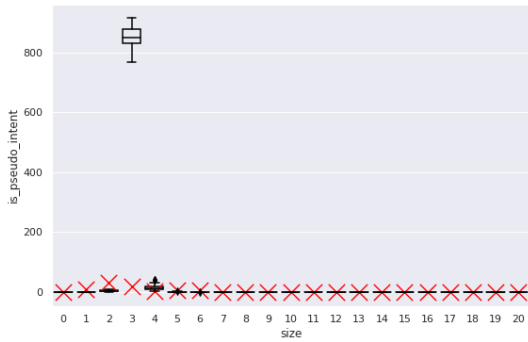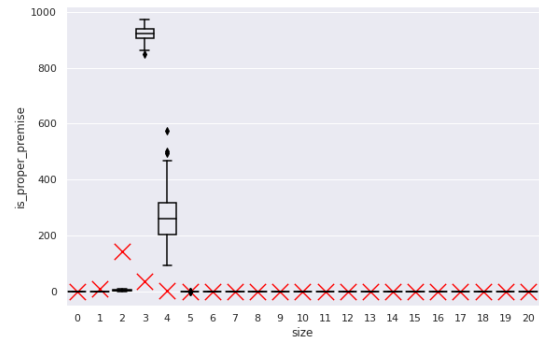## 5. Appendix: Figures Related to Experiments
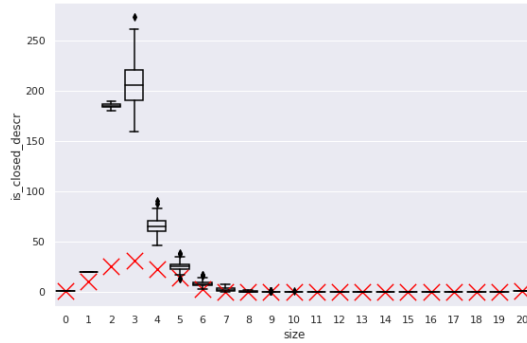
(a) Intents
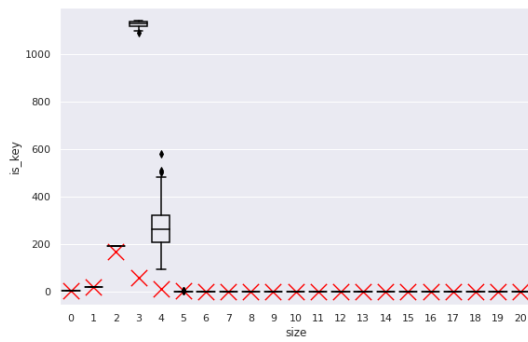
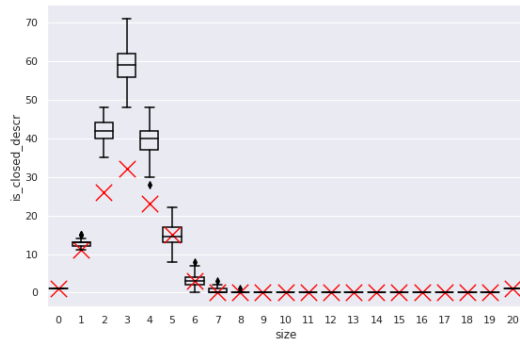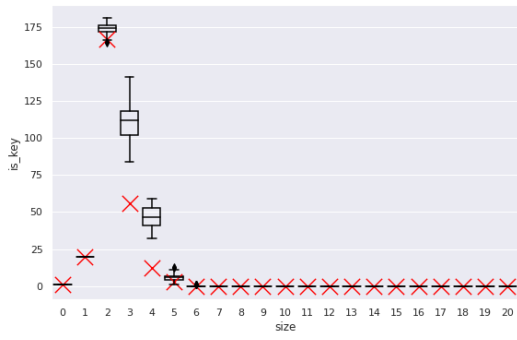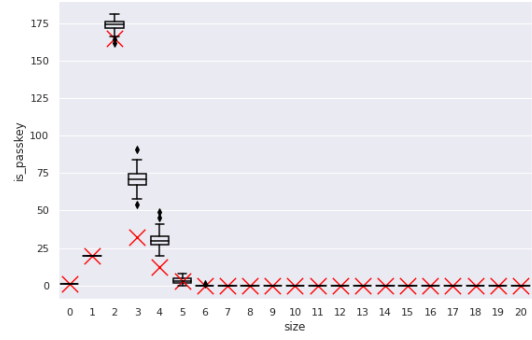
(b) Keys

(c) Passkeys

(d) Pseudo-intents

(e) Proper premises

Figure 4: The numbers of elements for the "Bob Ross" dataset w.r.t. context randomization based on density. Along the horizontal axes the sizes of the elements are shown, e.g., the number of attributes in the intents, pseudo-intents, keys, etc. Along the vertical axis the number of elements of the corresponding sizes are shown.
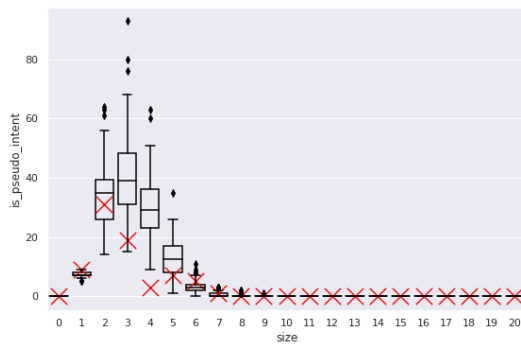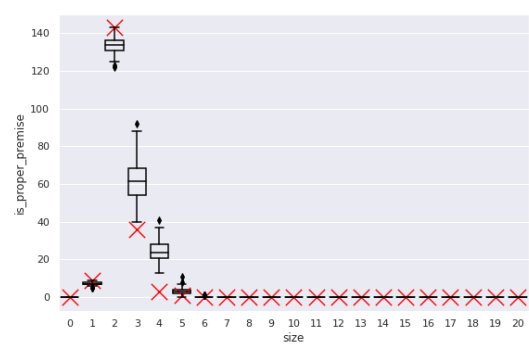
(a) Intents

(b) Keys

(c) Passkeys

(d) Pseudo-intents

(e) Proper premises

Figure 5: The numbers of elements for the "Bob Ross" dataset w.r.t. context randomization based on column-wise permutations.
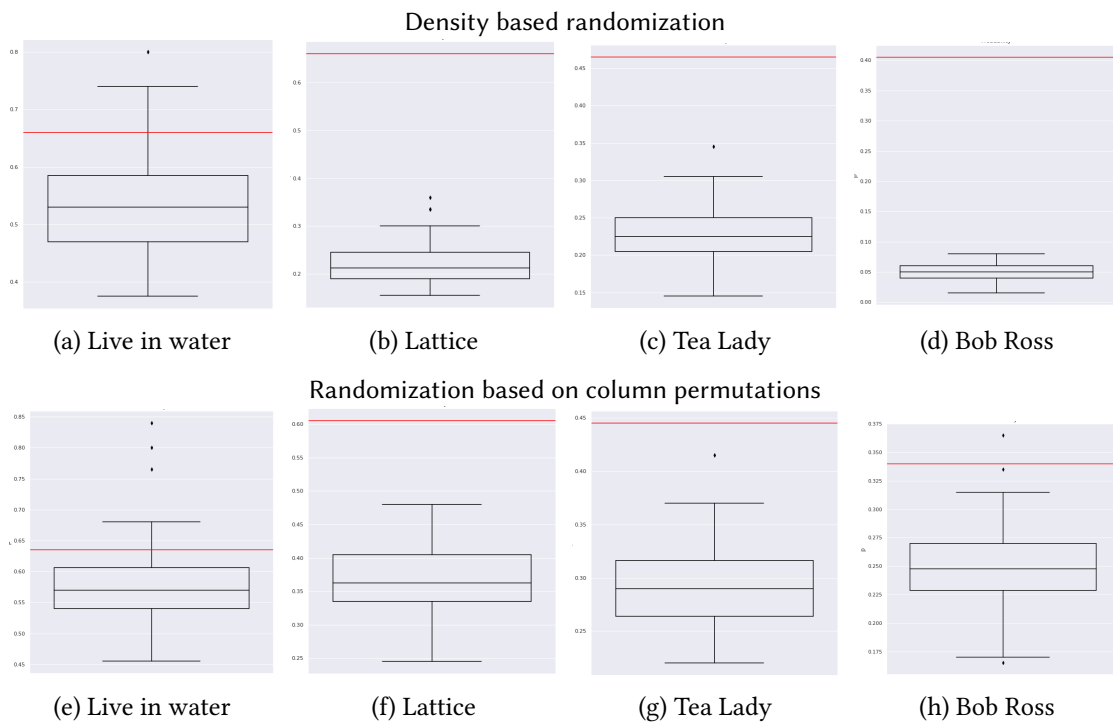
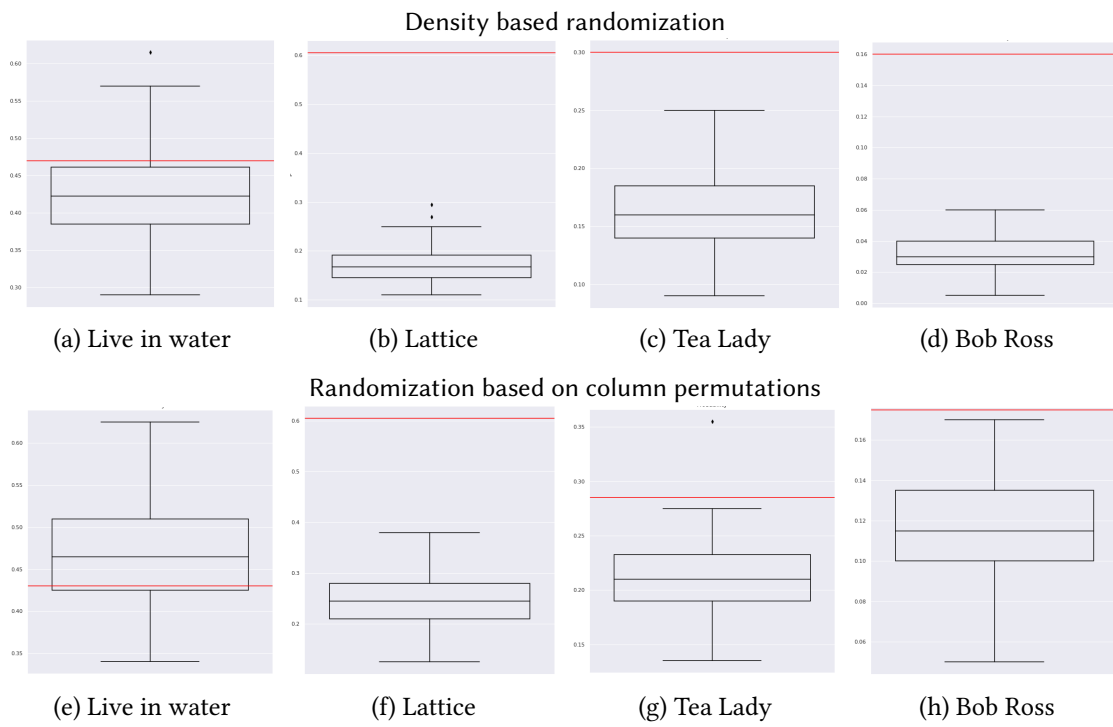**Figure 6:** The distributivity index for different datasets and different randomizations.

Figure 7: The linearity index for different datasets and different randomizations.