

Boolean Analogical Proportions - Axiomatics and Algorithmic Complexity Issues

Henri Prade^{1,2}(✉) and Gilles Richard¹

¹ IRIT, Toulouse University, Toulouse, France
{prade,richard}@irit.fr

² QCIS, University of Technology, Sydney, Australia

Abstract. Analogical proportions, i.e., statements of the form a is to b as c is to d , are supposed to obey 3 axioms expressing reflexivity, symmetry, and stability under central permutation. These axioms are not enough to determine a single Boolean model, if a minimality condition is not added. After an algebraic discussion of this minimal model and of related expressions, another justification of this model is given in terms of Kolmogorov complexity. It is shown that the 6 Boolean patterns that make an analogical proportion true have a minimal complexity with respect to an expression reflecting the intended meaning of the proportion.

1 Introduction

Despite the fact that conclusions obtained by analogical reasoning do no guarantee to be valid from a classical logic viewpoint, this kind of reasoning is considered as a valuable and often creative way to solve real life problems. Analogical proportions, i.e., relations between four items of the form a is to b as c is to d , constitute a key notion formalizing analogical inference and relying on the following principle: if such proportions hold on a noticeable subset of known features used for describing the four items, the proportion may still hold on other features as well, which may help guessing the unknown values of d on these other features from their values on a , b , and c [11, 19]. It is only quite recently that a logical modeling of these proportions has been proposed [12, 13], following several attempts at formalizing them in other settings [7, 10]. This logical modeling makes clear that the analogical proportion holds if and only if a differs from b as c differs from to d and vice-versa.

The paper investigates two new justifications of the Boolean expression of an analogical proportion. First, starting from the core axioms supposed to be satisfied by an analogical proportion, and agreed by everybody for a long time, this paper exhibits the Boolean models compatible with these axioms. There are several ones, but the smallest model is the standard Boolean expression of an analogical proportion previously proposed. This smallest model is characterized by six possible Boolean patterns (among sixteen candidates). In the second part of the paper, we try to evaluate their cognitive significance in terms of algorithmic complexity (i.e. Kolmogorov complexity) and show that they are also minimal

among all Boolean patterns with respect to an algorithmic complexity-based definition of analogy. Indeed algorithmic complexity measures a kind of universal information content of a Boolean string. Despite its inherent uncomputability, there exist powerful tool for computing good approximations. Kolmogorov complexity has been proved to be of great value in diverse applications: for example, in distance measures [1] and classification methods, plagiarism detection, network intrusion detection [5], and in numerous other applications [9].

The paper is organized as follows. In Sect. 2, we provide a background on the definition of an analogical proportion and its basic properties in a Boolean setting. In Sect. 3, we start from the characteristic axioms of analogical proportions, and we investigate the different compatible Boolean models. In Sect. 4, we briefly review the main definition and theorems of Kolmogorov complexity. As we have the main tools, we are in a position to give a Kolmogorov complexity-based definition of analogy in Sect. 5. Section 6 is devoted to a set of experiments that empirically validate our definition. Finally, we conclude in Sect. 7.

2 Background on Boolean Analogical Proportion

At the time of Aristotle, the idea of analogical proportion originated from the notion of numerical proportion. In that respect the *arithmetic* proportion between 4 integers a, b, c, d , which holds if $a - b = c - d$, is a good prototype of the idea of analogical proportion, since we can read it as “ a differs from b as c differs from d ”, which perfectly fits with “ a is to b as c is to d ”, denoted by $a : b :: c : d$. When considering Boolean interpretation where $a, b, c, d \in \{0, 1\}$, it is tempting to carry on with the same definition as $\{0, 1\} \subset \mathbb{R}$, with the inevitable drawback that difference is not an internal operator in $\mathbb{B} = \{0, 1\}$. Nevertheless, if we draw the truth table (16 lines) corresponding to this definition, we get Table 1 highlighting that only 6 among 16 lines are valid proportions.

Table 1. Boolean valuations for $a : b :: c : d$

a	b	c	d	$a : b :: c : d$	a	b	c	d	$a : b :: c : d$
0	0	0	0	1	1	0	0	0	0
0	0	0	1	0	1	0	0	1	0
0	0	1	0	0	1	0	1	0	1
0	0	1	1	1	1	0	1	1	0
0	1	0	0	0	1	1	0	0	1
0	1	0	1	1	1	1	0	1	0
0	1	1	0	0	1	1	1	0	0
0	1	1	1	0	1	1	1	1	1

Boolean Definition. Looking for a purely logical definition of $a : b :: c : d$, we need to make use of the comparative indicators [14, 15] that are naturally associated with a pair of variable (a, b) :

- $a \wedge b$ and $\neg a \wedge \neg b$: they are *positive similarity* and *negative similarity* indicators respectively; $a \wedge b$ (resp. $\neg a \wedge \neg b$) is true iff only both a and b are true (resp. false);
- $a \wedge \neg b$ and $\neg a \wedge b$: they are *dissimilarity* indicators ; $a \wedge \neg b$ (resp. $\neg a \wedge b$) is true iff only a (resp. b) is true and b (resp. a) is false.

Then analogical proportion is defined by the two *logically equivalent* expressions [13]:

$$a : b :: c : d = (a \wedge \neg b \equiv c \wedge \neg d) \wedge (\neg a \wedge b \equiv \neg c \wedge d) \quad (1)$$

$$a : b :: c : d = ((a \wedge d) \equiv (b \wedge c)) \wedge ((\neg a \wedge \neg d) \equiv (\neg b \wedge \neg c)) \quad (2)$$

Expression (1) reads “ a differs from b as c differs from d and b differs from a as d differs from c ”. This definition is equivalent to the previous one (it yields Table 1) with the advantage of being an internal definition inside \mathbb{B} . Expression (2) may be viewed as the logical counterpart of the well-known property of arithmetical proportions $a - b = c - d \Leftrightarrow a + d = b + c$. “ a is to b as c is to d ” can now be read “what a and d have in common, b and c have it also (both positively and negatively)”, which, however, is a less straightforward reading of the idea of analogy than the one associated with (1). As can be checked on Table 1, analogical proportions are *independent with respect to the positive or negative encoding* of properties: $(a : b :: c : d) \equiv (\neg a : \neg b :: \neg c : \neg d)$.

For representing objects one generally needs *vectors* of Boolean values, rather than single Boolean values, each component being the value of a binary attribute. The previous definition directly extends to Boolean vectors in \mathbb{B}^n of the form $\vec{a} = (a_1, \dots, a_n)$ as follows: $\vec{a} : \vec{b} :: \vec{c} : \vec{d}$ iff $\forall i \in [1, n], a_i : b_i :: c_i : d_i$.

Equation and Creativity. The equation $a : b :: c : x$ has a unique solution $x = c \equiv (a \equiv b)$ provided that $(a \equiv b) \vee (a \equiv c)$ holds. Indeed neither $0 : 1 :: 1 : 0$ nor $1 : 0 :: 0 : 1$ holds true. This process can be extended componenwise to vectors. In that case, for instance, the following equation $010 : 100 : 011 : x$ has for unique solution the vector $(1, 0, 1)$ which is not among the 3 previous vectors $(0, 1, 0)$, $(1, 0, 0)$, $(0, 1, 1)$. Then analogical proportions for vectors are *creative* (an informal quality usually associated with the idea of analogy) as they may involve 4 distinct vectors.

A Previous View of Analogical Proportion. In [6], S. Klein suggests that an analogical proportion would hold as soon as a, b, c are completed by d taken as $d = c \equiv (a \equiv b)$. This amounts to define it as $A_K(a, b, c, d) \triangleq (a \equiv b) \equiv (c \equiv d)$. Then $0 : 1 :: 1 : 0$ and $1 : 0 :: 0 : 1$ become valid analogical proportions and leads to the model denoted Kl in the following section. The validity of such patterns may be advocated on the basis of a functional view of analogy where $a : f(a) :: b : f(b)$ sounds indeed valid, taking the negation in \mathbb{B} for f . But, this is debatable since $A_K(a, b, c, d) \Leftrightarrow A_K(b, a, c, d)$ (which does not fit with intuition). It turns out that $a : b :: c : d \Rightarrow A_K(a, b, c, d)$.

Lower Approximations of Analogical Proportion. While $A_K(a, b, c, d)$ is an upper approximation of $a : b :: c : d$ true for 8 patterns, one may look for

lower approximations that are true for 4 patterns only (taking into account code independency). There are 3 such approximations, given below, followed by the patterns they validate¹:

$$(a \equiv b) \wedge (c \equiv d) \begin{array}{|c|c|c|c|} \hline 1 & 1 & 1 & 1 \\ \hline 0 & 0 & 0 & 0 \\ \hline 1 & 1 & 0 & 0 \\ \hline 0 & 0 & 1 & 1 \\ \hline \end{array}; (a \equiv c) \wedge (b \equiv d) \begin{array}{|c|c|c|c|} \hline 1 & 1 & 1 & 1 \\ \hline 0 & 0 & 0 & 0 \\ \hline 1 & 0 & 1 & 0 \\ \hline 0 & 1 & 0 & 1 \\ \hline \end{array}; (a \neq d) \wedge (b \neq c) \begin{array}{|c|c|c|c|} \hline 1 & 1 & 0 & 0 \\ \hline 0 & 0 & 1 & 1 \\ \hline 1 & 0 & 1 & 0 \\ \hline 0 & 1 & 0 & 1 \\ \hline \end{array}.$$

Note that only the last one remains creative.

The question addressed now is “Could an axiomatic view of analogical proportions offer a kind of intrinsic justification that only the 6 patterns obeying (1)–(2) are acceptable?”

3 Analogy and Its Lattice of Boolean Models

Analogy, viewed as a quaternary relation R , is supposed to obey 3 axioms (e.g., [7,10]):

1. $\forall a, b, R(a, b, a, b)$ (reflexivity);
2. $\forall a, b, c, d, R(a, b, c, d) \rightarrow R(c, d, a, b)$ (*symmetry*);
3. $\forall a, b, c, d, R(a, b, c, d) \rightarrow R(a, c, b, d)$ (*central permutation*).

These axioms are clearly inspired by numerical proportions. From them, some basic properties can be deduced by proper applications of symmetry and central permutation:

- $\forall a, b, R(a, a, b, b)$ (identity);
- $\forall a, b, c, d, R(a, b, c, d) \rightarrow R(b, a, d, c)$ (*inside pair reversing*);
- $\forall a, b, c, d, R(a, b, c, d) \rightarrow R(d, b, c, a)$ (*extreme permutation*).

In fact, another (less standard) axiom expected from a *natural* analogy is:

$$\forall a, b, R(a, a, b, x) \implies x = b \text{ (unicity)}$$

All these properties fit with our intuition of what may be an analogical proportion. In this paper, we focus on $\mathbb{B} = \{0, 1\}$ as interpretation domain. In that case, R should be interpreted as a subset of \mathbb{B}^4 : removing the emptyset leaves $2^{16} - 1$ candidate models. It is straightforward to get a basic model. By applying reflexivity, we see that 0101, 1010 should belong to the relation and 0000, 1111 as well since we may have $a = b$, and central permutation then leads to add 0011 and 1100. Thus, we get the model $\Omega_0 = \{0000, 1111, 0101, 1010, 0011, 1100\}$, which is

¹ There are 3 companion approximations that involve the two additional patterns of A_K :

$$(a \equiv d) \wedge (b \equiv c) \begin{array}{|c|c|c|c|} \hline 1 & 1 & 1 & 1 \\ \hline 0 & 0 & 0 & 0 \\ \hline 1 & 0 & 0 & 1 \\ \hline 0 & 1 & 1 & 0 \\ \hline \end{array}; (a \neq b) \wedge (c \neq d) \begin{array}{|c|c|c|c|} \hline 1 & 0 & 0 & 1 \\ \hline 0 & 1 & 1 & 0 \\ \hline 1 & 0 & 1 & 0 \\ \hline 0 & 1 & 0 & 1 \\ \hline \end{array}; (a \neq c) \wedge (b \neq d) \begin{array}{|c|c|c|c|} \hline 1 & 1 & 0 & 0 \\ \hline 0 & 0 & 1 & 1 \\ \hline 1 & 0 & 0 & 1 \\ \hline 0 & 1 & 1 & 0 \\ \hline \end{array}.$$

stable under symmetry. Ω_0 is the *smallest* model for analogical proportion over \mathbb{B}^2 . However, one may ask about other models, and we can show the following:

Property 1. *There are exactly 8 models of analogy (satisfying the 3 first axioms) over \mathbb{B} . There are exactly 2 models of analogy (satisfying the 3 first axioms plus unicity).*

Proof. Any model should include Ω_0 . Let us note that a bigger model should necessarily have an even cardinality due to the following facts:

- To be bigger than Ω_0 , it should contain a string s containing both 0 and 1.
- Thanks to symmetry or central permutation axioms, it should contain the symmetric $cdab$ of $s = abcd$ and the central permutation $acbd$ of s : necessarily, one of these 2 strings is different from s (otherwise, we get $a = b = c = d$).

So we have to look for models of cardinality 8, 10, 12, 14 and 16. Obviously \mathbb{B}^4 of cardinality 16 is a model, the biggest one. Due to the axioms, we have to add to Ω_0 subsets of \mathbb{B}^4 that are stable w.r.t. symmetry and central permutation. We have exactly:

- one subset with 2 elements: $S_2 = \{1001, 0110\}$
- two subsets with 4 elements: (i) $S_3 = \{1110, 1101, 1011, 0111\}$; (ii) $S_4 = \{0001, 0010, 0100, 1000\}$.

Since every model has to be built by adding to Ω_0 one of the previous subsets, we get the following models for analogy in \mathbb{B} :

- (1) 1 model with 6 elements: Ω_0 (the smallest one)
- (2) 1 model with 8 elements: $Kl = \Omega_0 \cup S_2 = \{0000, 1111, 0101, 1010, 0011, 1100, 0110, 1001\}$

As previously explained, this model is due to Klein [6].

- (3) 2 model with 10 elements:
 - $M_3 = \Omega_0 \cup S_3 = \{0000, 1111, 0101, 1010, 0011, 1100, 1110, 1101, 1011, 0111\}$,
 - $M_4 = \Omega_0 \cup S_4 = \{0000, 1111, 0101, 1010, 0011, 1100, 0001, 0010, 0100, 1000\}$
- (4) 2 models with 12 elements:
 - $M_5 = M_3 \cup S_2 = \{0000, 1111, 0101, 1010, 0011, 1100, 1110, 1101, 1011, 0111, 0110, 1001\}$,
 - $M_6 = M_4 \cup S_2 = \{0000, 1111, 0101, 1010, 0011, 1100, 0001, 0010, 0100, 1000, 0110, 1001\}$,
- (5) 1 model with 14 elements:
 - $M_7 = M_3 \cup S_4 = M_4 \cup S_3 = \Omega_0 \cup S_3 \cup S_4 = \{0000, 1111, 0101, 1010, 0011, 1100, 1110, 1101, 1011, 0111, 0100, 1000, 0110, 1001\}$,
- (6) 1 model with exactly 16 elements: $\Omega = \Omega_0 \cup S_2 \cup S_3 \cup S_4 = \mathbb{B}$.

Finally Ω_0 and Kl satisfy unicity but M_3 (containing 1100 and 1101) and M_4 (containing 0000 and 0001) do not satisfy. This achieves the proof. \square

² Note that lower approximations of analogical proportions miss at least one axiom.

The set of models is a lattice with bottom element Ω_0 and top element \mathbb{B} , see Fig. 1. As can be seen, 8 models fit with the axioms in the Boolean case, including the 6-patterns model Ω_0 and the 8-patterns model Kl due to Klein. However, it is natural to privilege the smallest model, the minimal one that just accounts for the axioms and nothing more.

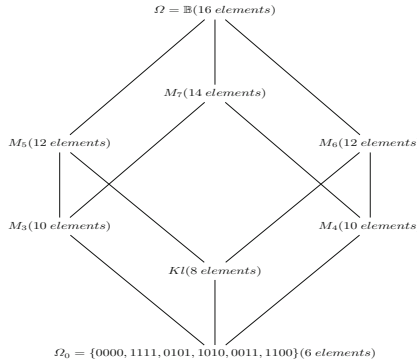


Fig. 1. The lattice of Boolean models of analogy

We now investigate if another justification in favor of the minimal model Ω_0 can be obtained by minimizing an expression reflecting the information content of an analogical proportion in terms of Kolmogorov complexity. We now review the fundamentals of Kolmogorov complexity theory, also known as Algorithmic Complexity Theory.

4 Kolmogorov Complexity: A Brief Review

Kolmogorov complexity is not a new concept and the theory has been designed many years ago: see for instance [9] for an in depth study. This theory has not to be confused with Shannon information theory [16] despite the fact that they share some links.

The Starting Point. We need the help of a universal Turing machine denoted U . Then p denotes a program running on U . Two situations can happen: (i) either p does not stop for the input x , or (ii) p stops for the input x and outputs a finite string y . In that case, we write $U(p, x) = y$. The Kolmogorov complexity [9] of y w.r.t. x is then defined as:

$$K_U(y/x) = \min\{|p|, U(p, x) = y\}.$$

$K_U(y/x)$ is the size of the shortest program able to reconstruct y with the help of x . The Kolmogorov complexity [9] of y is just obtained with the empty string ϵ :

$$K_U(y) = \min\{|p|, U(p, \epsilon) = y\}.$$

Given a string s , $K_U(s)$ is an integer which, in some sense, is a measure of the information content of s : instead of sending s to somebody, we can send p from which s can be recovered as soon as this somebody has the machine U . K_U enjoys a lot of properties among which a kind of universality: this complexity is independent of the underlying Turing machine as we have the invariance theorem [9]:

Theorem 1. *If U_1 and U_2 are two universal Turing machines, there exists a constant $c_{U_1U_2}$ such that for all string s : $|K_{U_1}(s) - K_{U_2}(s)| < c_{U_1U_2}$, where $K_{U_1}(s)$ and $K_{U_2}(s)$ denote the algorithmic complexity of s w.r.t. U_1 and U_2 respectively.*

This theorem guarantees that complexity values may only diverge by a constant c (e.g. the length of a compiler or a translation program) and for huge complexity strings, we can denote K without specifying the Turing machine U . It can also be shown that [9]:

Theorem 2. $\forall x, y, K(xy) = K(x) + K(y/x) + \mathcal{O}(1)$.

Unfortunately K has been proved as a non-computable function [9]. But in fact, K or an upper bound of K can be estimated in diverse ways that we investigate now.

Complexity Estimation. The first well known option available to estimate K is via lossless compression algorithm. For instance *gzip* approximates better than *gzip*, and the *PAQ* family is still better than *gzip2*. Due to the invariance theorem, when the size of s is huge, using compression will provide a relatively stable approximation as the constant c in the theorem can be considered as negligible. It is obviously not the case when the size of s becomes small. When s is short, compression is not a valid option. On another side, the constant c can prevent for providing stable approximations of $K(s)$. Luckily, the works of [3, 4, 17] give means of providing sensible values for the complexity of short strings (i.e. less than 10bits). This job has been done by the Algorithmic Nature Group (<https://algorithmicnature.org/>). They have developed a tool OACC (<http://www.complexitycalculator.com/>) allowing to estimate the complexity of short strings. The authors derived their approach from a theorem from Levin [4, 8] establishing the exact connection between $m(s)$ and $K(s)$, where $m(s)$ is a semi-measure known as the Universal Distribution defined as follows [18]: $m(s) = \sum_{p:U(p,\epsilon)=s} 2^{-|p|}$.

Theorem 3. *There exists a constant c depending only of the underlying Turing machine such that: $\forall s, |-\log_2(m(s)) - K(s)| < c$.*

Rewriting the formula as $K(s) = -\log_2(m(s)) + \mathcal{O}(1)$, shows that estimating K could also be done via estimating $m(s)$. Estimating $m(s)$ becomes realistic when s is short as we have to estimate the probability for s to be the output of a short program. Considering simple Turing machines as described in [17], over a Boolean alphabet $\{0, 1\}$ and a finite number n of states $\{1, \dots, n\}$ plus a

special Halt state denoted 0, there are exactly $(4n + 2)^{2n}$ such Turing machines. Using clever optimizations [17], running these machines for $n = 4$ and $n = 5$ becomes realistic and provides an estimation of $m(s)$ and ultimately of $K(s)$. In the following, we denote $K'(s)$ this OACC estimation of $K(s)$.

Short Chains Complexity Estimation. Some properties are expected from a complexity calculator machinery to be in accordance with a cognitive process:

1. There is no way to distinguish strings of length 1 and it is absolutely clear that $K(0) = K(1)$ should hold whatever the considered universal Turing machine.
2. An important point is to be able to distinguish the 4 strings of length 2: 00, 11, 10, 01 and we expect the following properties: $K(00) = K(11) < K(01) = K(10)$;
3. In terms of n bits strings, we expect $0 \dots 0$ and $1 \dots 1$ to be the simplest ones and to have the same complexity.

Observing the tables in [4], it appears that the properties above are satisfied, namely:

- Whatever the number of states of a 2-symbols Turing machine, $K'(0) = K'(1)$.
- Whatever the number of states of a 2-symbols Turing machine, $K'(00) = K'(11) = a$, $K'(01) = K'(10) = b$ and $a < b$.
- Whatever the number of states of a 2-symbols Turing machine, and for strings of length less than or equal to 10 (short strings) $K'(0 \dots 0) = K'(1 \dots 1) = a$ and a is the minimum value among the set of values.

Then the estimation of K via K' coming from the OACC estimator is a suitable candidate for our purpose. But before going further, we have first to check that OACC validate the above conditions. As we can check by examining Table 2 and column 4 of the final table in Sect. 6, these basic cognitive evidences are confirmed with the OACC tool. So we can start from OACC to check the properties required to validate the analogical hypothesis that we propose in the next section.

Table 2. Complexity of 1 bit and 2 bits chains with OACC

	$K(x)$
0	3.5473880692782100
1	3.5473880692782100

x_1x_2	$K(x_1x_2)$
00	5.4141012345247104 = a
01	5.4141040197301500 = b
10	5.4141040197301500 = b
11	5.4141012345247104 = a

5 An Algorithmic Complexity View of Analogical Proportions

As described in our introduction, several attempts have been done to formalize analogy or analogical reasoning with mitigate success. In this paper, as it has been the case in the works of [2], we adopt a machine learning viewpoint. Our aim is to integrate analogical reasoning in the global landscape of predicting values from observable examples.

When stated in a machine learning perspective, the problem of analogical inference is as follows: for a given x_3 , predict x_4 such that the *target* pair (x_3, x_4) is in the *same* relation that another given *source* pair (x_1, x_2) considered as an example. The pair (x_3, x_4) is the target pair which is partially known. In the case of classification where the 2nd element in a pair is the label, it amounts to predict the label of x_3 having only one classified example (x_1, x_2) at hand.

A functional view amounts to considering a hidden function f such that $x_2 = f(x_1)$ and we have to guess $x_4 = f(x_3)$. This functional view is the one developed in [2]: the problem of analogical inference strictly fits with a regression problem but with only one example. Ruling out any statistical models, this approach needs a brand new formalization that the authors extract from algorithmic complexity theory. Instead of trying to find regularities among a large set of observations (statistical approach), they consider the very meaning of each of the 3 observables $x_1, x_2 = f(x_1)$ and x_3 . We start from this philosophy, but we depart from it as below:

- We focus on the Boolean case where the 3 objects under consideration are Boolean vectors. So we do not have to care about the change between the source domain representation and the target domain representation: these 2 domains are identical. The cost of this representation change is null in terms of algorithmic complexity.
- To be in line with the machine learning minimal assumption that there exists some unknown probability distribution P from which the data are drawn, we do not consider that x_2 is a (hidden) function of x_1 . We just have a probability of observing x_2 having already observed x_1 which is more general than associating a fixed x_2 with every given x_1 . It could be the case that for another x'_2 we still have $x_1 : x'_2 :: x_3 : x_4$.

As a consequence, we start from the following intuitions:

1. For $x_1 : x_2 :: x_3 : x_4$ to be accepted as a valid analogy, it is clear that the way we go from x_1 to x_2 should not be very different from the way we go from x_3 to x_4 (but it has not to be a functional link). We suggest to measure this expected proximity with the difference $|K(x_2/x_1) - K(x_4/x_3)|$. Considering $K(x_2/x_1)$ as the *difficulty* to build x_2 from x_1 , the previous expression $|K(x_2/x_1) - K(x_4/x_3)|$, when small, tells us that it is not more difficult to build x_4 from x_3 than to build x_2 from x_1 , and vice versa. This is what we call the atomic view of analogy. But this is obviously not enough.

2. In fact, the previous formula does not tell anything about the link between the pair (x_1, x_2) and the pair (x_3, x_4) . For $x_1 : x_2 :: x_3 : x_4$ to be accepted as a valid analogy, the *difficulty* to apprehend the string x_1x_2 from the string x_3x_4 should be close to the difficulty to apprehend x_3x_4 from x_1x_2 . We suggest to measure this expected proximity with the difference $|K(x_1x_2/x_3x_4) - K(x_3x_4/x_1x_2)|$. This difference is obviously symmetric and is linked to the symmetry of an analogy.
3. Above all, the global picture has to be “simple” i.e. telling that $x_1 : x_2 :: x_3 : x_4$ is a valid analogy should not be too disturbing, at least from a cognitive viewpoint. This means that the occurrence of the string $x_1x_2x_3x_4$ in this order should be highly plausible. We suggest to measure this plausibility with $K(x_1x_2x_3x_4)$ which is the size of the shortest program producing the binary string $x_1x_2x_3x_4$ from a universal Turing machine.

Following the ideas of [2], we use the sum as aggregator operator and denote $k(x_1x_2x_3x_4)$ the following formula measuring, in some sense, the quality of an analogy:

$$|K(x_2/x_1) - K(x_4/x_3)| + |K(x_1x_2/x_3x_4) - K(x_3x_4/x_1x_2)| + K(x_1x_2x_3x_4)$$

This leads us to postulate that the “best” x_4 we are looking for to build a valid analogy $x_1 : x_2 :: x_3 : x_4$ is the one minimizing this expression. So, we have: $x_4 = \operatorname{argmin}_u k(x_1x_2x_3u)$. Let us see if we can, at least from an empirical viewpoint, validate this model.

6 Validation in the Boolean Setting

As we are not in a position to prove something at this stage, let us just investigate now the empirical evidence for our formula. One point to start with is to check if this formula holds in the very basic Boolean case. Considering x_1, x_2, x_3, x_4 as Boolean values, we have to check how the 6 cases of valid analogical proportions actually behave w.r.t. the formula $k(x_1x_2x_3x_4)$. Thus, we have to estimate formula $k(x_1x_2x_3x_4)$ for every $x_1x_2x_3x_4 \in \mathbb{B}^4$. The point is that our strings are very short: only 4 bits. So, as explained in Sect. 4, we have to rely on OACC instead of a compression estimation.

The Less Complex Analogical Chains. On top of that, we have to consider, not only pure Kolmogorov complexity K but also complexity w.r.t. a given string as in $K(x_3x_4/x_1x_2)$. Generally, it is quite clear that $K(xy) \leq K(x) + K(y/x)$: roughly speaking, we can build a program whose output is xy by concatenating a program whose output is x to a program taking x as input and providing y as output. It is more difficult to get a more precise bound. Thanks to Theorem 2: $K(xy) = K(x) + K(y/x) + \mathcal{O}(1)$, which shows that we can approximate $K(y/x)$ with $K(xy) - K(x)$. As we now have all the tools needed to approximate formula k , it remains to use OACC to compute the estimation. The following table reports the results of this computation:

Using OACC <http://www.complexitycalculator.com/>
 $k(abcd) = |K(b/a) - K(d/c)| + |K(cd/ab) - K(ab/cd)| + K(abcd)$

abcd	A K(b/a)	B K(d/c)	K(abcd)	C k(cd/ab)	D K(ab/cd)	A-B + C-D +K(abcd)
0000	1.8667131652	1.8667131652	11.2174683967	5.8033671621	5.8033671621	11.2174683967
1111	1.8667131652	1.8667131652	11.2174683967	5.8033671621	5.8033671621	11.2174683967
0101	1.8667159505	1.8667159505	11.7002793293	6.2861753096	6.2861753096	11.7002793293
1010	1.8667159505	1.8667159505	11.7002793293	6.2861753096	6.2861753096	11.7002793293
0001	1.8667131652	1.8667159505	11.5731249872	6.1590237527	6.3435937193	11.757697739
1110	1.8667131652	1.8667159505	11.5731249872	6.1590237527	6.3435937193	11.757697739
0111	1.8667159505	1.8667131652	11.5731249872	6.1590209675	6.3435965045	11.7577033094
1000	1.8667159505	1.8667131652	11.5731249872	6.1590209675	6.3435965045	11.7577033094
0011	1.8667131652	1.8667131652	11.8099819092	6.3958806747	6.3958806747	11.8099819092
1100	1.8667131652	1.8667131652	11.8099819092	6.3958806747	6.3958806747	11.8099819092
0100	1.8667159505	1.8667131652	11.757697739	6.3435937193	6.1590237527	11.9422704908
1011	1.8667159505	1.8667131652	11.757697739	6.3435937193	6.1590237527	11.9422704908
0010	1.8667131652	1.8667159505	11.757697739	6.3435965045	6.1590209675	11.9422760613
1101	1.8667131652	1.8667159505	11.757697739	6.3435965045	6.1590209675	11.9422760613
1001	1.8667159505	1.8667159505	12.0548412692	6.6407372495	6.6407372495	12.0548412692
0110	1.8667159505	1.8667159505	12.0548412692	6.6407372495	6.6407372495	12.0548412692

As can be seen for the 6 patterns of the model Ω_0 of analogical proportion, the unique solution of equation $a : b :: c : x$ always corresponds to a string $abcx$ that minimizes expression k wrt the other option $abc\bar{x}$ (where $\bar{x} = \neg x$), e.g. $k(1111) < k(1110)$. Besides 0101 is simpler than 0110 despite the fact that in the second case there is also an underlying function such that $x_2 = f(x_1)$ and $x_4 = f(x_3)$: the negation. Note that 0110 and 1001 exhibit the highest complexity as estimated by OACC. It eliminates Kl . As there is no known convergence result regarding K and that we cannot estimate the constant in the formula $K(s) = -\log_2(m(s)) + \mathcal{O}(1)$, these experiences should only be considered as adding a bit of credibility to the smallest model.

7 Conclusion

We have given a complete description of the Boolean models of analogy. To choose the most relevant one among the possible 8 models beyond the minimality argument, we have proposed a complexity-based definition for Boolean analogical proportion. Using a set of calculations with OACC, the tool developed by the Algorithmic Nature Group (<https://algorithmicnature.org/>), we have checked that the truth table of the Boolean analogy fits with the fact that the corresponding combinations minimize the given complexity formula. It remains to consider the formula in a more general setting than the Boolean one. This would in particular allow to establish a link between transfer learning and Kolmogorov complexity. Another point would be of interest: to be able to solve the minimization problem associated to the formula. Doing so would be to solve the analogical equation $a : b :: c : x$. This might be the basis of a constructive process.

References

1. Bennett, C.H., Gács, P., Li, M., Vitányi, P., Zurek, W.H.: Information distance (2010). CoRR abs/1006.3520
2. Cornuéjols, A.: Analogy as minimization of description length. In: Nakhaeizadeh, G., Taylor, C. (eds.) *Machine Learning and Statistics: The Interface*, pp. 321–336. Wiley, Chichester (1996)
3. Delahaye, J.P., Zenil, H.: On the Kolmogorov-Chaitin complexity for short sequences (2007). CoRR abs/0704.1043
4. Delahaye, J.P., Zenil, H.: Numerical evaluation of algorithmic complexity for short strings: a glance into the innermost structure of randomness. *Appl. Math. Comput.* **219**(1), 63–77 (2012)
5. Goel, S., Bush, S.F.: Kolmogorov complexity estimates for detection of viruses in biologically inspired security systems: a comparison with traditional approaches. *Complexity* **9**(2), 54–73 (2003)
6. Klein, S.: Whorf transforms and a computer model for propositional/appositional reasoning. In: *Proceedings of the Applied Mathematics colloquium*, University of Bielefeld, West Germany (1977)
7. Lepage, Y.: Analogy and formal languages. *Electr. Notes Theor. Comp. Sci.* **53**, 180–191 (2002). Moss, L.S., Oehrle, R.T. (eds.) *Proceedings of the Joint Meeting of the 6th Conference on Formal Grammar and the 7th Conference on Mathematics of Language*
8. Levin, L.: Laws of information conservation (non-growth) and aspects of the foundation of probability theory. *Probl. Inf. Transm.* **10**, 206–210 (1974)
9. Li, M., Vitányi, P.: *An Introduction to Kolmogorov Complexity and Its Applications*, 3rd edn. Springer, New York (2008)
10. Miclet, L., Delhay, A.: Relation d’analogie et distance sur un alphabet défini par des traits. Technical Report 1632, IRISA, July 2004
11. Miclet, L., Delhay, A.: Analogical dissimilarity: definition, algorithms and first experiments in machine learning. Technical Report RR-5694, INRIA, July 2005
12. Miclet, L., Prade, H.: Logical definition of analogical proportion and its fuzzy extensions. In: *Annual Meeting of the North American Fuzzy Information Processing Society (NAFIPS)*, New-York, pp. 1–6. IEEE (2008)
13. Miclet, L., Prade, H.: Handling analogical proportions in classical logic and fuzzy logics settings. In: Sossai, C., Chemello, G. (eds.) *ECSQARU 2009*. LNCS (LNAI), vol. 5590, pp. 638–650. Springer, Heidelberg (2009). doi:[10.1007/978-3-642-02906-6_55](https://doi.org/10.1007/978-3-642-02906-6_55)
14. Prade, H., Richard, G.: From analogical proportion to logical proportions. *Logica Universalis* **7**(4), 441–505 (2013)
15. Prade, H., Richard, G.: Homogenous and heterogeneous logical proportions. *IfCoLog J. Logics Appl.* **1**(1), 1–51 (2014)
16. Shannon, C.E.: A mathematical theory of communication. *Bell Syst. Tech. J.* **27**(3), 379–423 (1948)
17. Soler-Toscano, F., Zenil, H., Delahaye, J.P., Gauvrit, N.: Correspondence and independence of numerical evaluations of algorithmic information measures. *Computability* **2**, 125–140 (2013)
18. Solomonoff, R.J.: A formal theory of inductive inference. Part i and ii. *Inf. Control* **7**(1), 1–22 and 224–254 (1964)
19. Stroppa, N., Yvon, F.: Du quatrième de proportion comme principe inductif: une proposition et son application à l’apprentissage de la morphologie. *Traitement Automatique des Langues* **47**(2), 1–27 (2006)