Discovering Latent Causes and Memory Modification: A Computational Approach Using Symmetry and Geometry

Arif Dönmez[*] arif.doenmez@iuf-duesseldorf.de
IUF - Leibniz Research Institute for Environmental Medicine
DNTOX GmbH

Editors: Sophia Sanborn, Christian Shewmake, Simone Azeglio, Nina Miolane

Abstract

We learn from our experiences, even though they are never exactly the same. This implies that we need to assess their similarity to apply what we have learned from one experience to another. It is proposed that we "cluster" our experiences based on (hidden) *latent causes* that we infer. It is also suggested that surprises, which occur when our predictions are incorrect, help us categorize our experiences into distinct groups. In this paper, we develop a computational theory that emulates these processes based on two basic concepts from intuitive physics and Gestalt psychology using symmetry and geometry. We apply our approach to simple tasks that involve inductive reasoning. Remarkably, the output of our computational approach aligns closely with human responses.

Keywords: computational cognitive science, symmetry, geometry, algebra, gestalt psychology, algorithm, latent causes, memory modification, unsupervised learning, categorization, artificial intelligence

1. Introduction

We learn from our experiences, even though they are never exactly the same. This implies that we need to assess their similarity to apply what we have learned from one experience to another. Yael Niv and her colleagues have proposed that we "cluster" our experiences based on hidden *(latent) causes* that we infer (Gershman and Niv, 2013; Gershman et al., 2015). They have also suggested that surprises, which occur when our predictions are incorrect, help us categorize our experiences into distinct groups (Gershman et al., 2017).

Imitating human intelligence is our primary objective. We strive to replicate intelligent behavior in machines, including their weak points, such as bias. In Lake et al. (2017) a set of core ingredients for building more human-like learning and thinking machines are proposed. In this paper, we will use the first set of ingredients which focuses on developmental "start-up software" to develop a computational theory that emulates these processes. Central to our theory is the idea that the hidden (latent) causes can be deduced by *linear symmetries* within (sensory) data. The linearity assumption is mainly motivated by the "simplicity principle" (Chater and Vitányi, 2003; Feldman, 2003): the idea that humans tend to simple explanations. We employ statistical tests to determine whether current (sensory) data aligns with previously inferred groups. If no previously inferred latent cause adequately predicts the current sensory data, we adjust the symmetries and form new memories.

The first main ingredient we are using in this paper, comes from intuitive/naive physics (Wellman and Gelman, 1992; Lake et al., 2017), and the simplest way you can think of

[*] www.arifdoenmez.github.io

the concept is that Figure 1 remains unchanged when the positions of the red dots are swapped. The second and last ingredient we are using is the *law of continuity* ("Gesetz der durchgehenden Linie") from Gestalt psychology (Koffka, 1935; Köhler, 1967) (Figure 2). Both ingredients combined result in the computational framework of *linear representations* resp. *linear actions* of the *symmetric group* (Lemma 11).

Figure 1: Figure remains the same when dots are swapped.

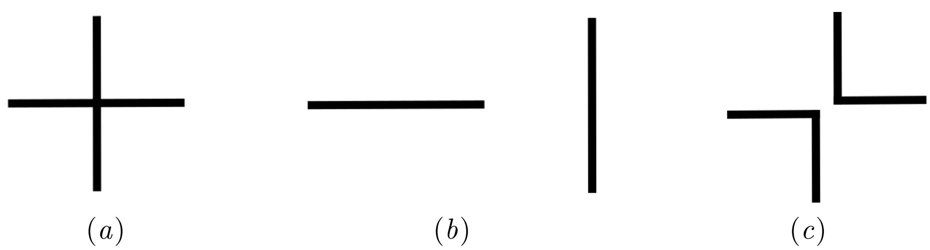

Figure 2: Law of continuity: Lines are always seen as following the simplest path. If two lines cross (Figure 2(*a*)), we do not assume that the course of the lines makes a bend at this point (Figure 2(*c*)), but we see two straight, continuous lines (Figure 2(*b*)).

Simulations demonstrate that the output of our computational approach closely matches human responses in simple tasks that require inductive reasoning. For the simulations, we use symbolic computation algorithms by Derksen and Kemper (2015); Sturmfels (2008) implemented within the computer algebra ecosystem `OSCAR` Symbolic Tools (2023) in the `Julia` programming language (Bezanson et al., 2017).

1.1. Organization

We introduce all necessary notation and the modern abstract definition of symmetry as "invariance under a specified group of transformations" in Section 2. In Section 3 and Section 4 we introduce our computational theory with linear and non-linear pattern recognition examples. We demonstrate in Section 5 how our approach replicates human-like reasoning and generalization. In Section 6, we discuss related work and contributions of the considerations in this paper. We try to summarize our approach in the form of a strategy blueprint in Appendix B, and we put the code used in this paper in Appendix C.

2. Definitions and Notation

2.1. The definition of a group

First, we need to introduce the notion of a monoid.

Definition 1 *A monoid is a set M equipped with a law of composition*

$$? \cdot ? : M \times M \to M, (x, y) \mapsto x \cdot y,$$

and a neutral element $1 \in M$, such that associativity, that is

$$(x \cdot y) \cdot z \;=\; x \cdot (y \cdot z),$$

and unitality, that is $1 \cdot x \;=\; x$, are satisfied for all $x, y, z \in M$.

Next, we can define a group:

Definition 2 *A group is a monoid $(G, \cdot, 1)$ in which every element has an inverse, that is for every $x \in G$, there is $y \in G$ such that the equation $y \cdot x \;=\; 1$ is satisfied.*

Example 1 *Let X denote a set and $\mathrm{Sym}(X) := \{f : X \to X \,:\, f \text{ is bijective}\}$. By using the composition of maps as the law of composition we end up with the symmetric group of X $(\mathrm{Sym}(X), \circ, \mathrm{id})$. For $X = \{1, \ldots, n\}$ ($n \in \mathbb{N}$), we write $\mathrm{Sym}_n := \mathrm{Sym}(X)$.*

Example 2 *Let V denote a vector space over a field $\mathbb{K}$ and*

$$\mathrm{GL}_{\mathbb{K}}(V) := \{f : V \to V \,:\, f \text{ is } \mathbb{K}\text{-linear and bijective}\}.$$

By using the composition of maps as the law of composition again we end up with a group $(\mathrm{GL}_{\mathbb{K}}(V), \circ, \mathrm{id})$. For $V = \mathbb{K}^n$ ($n \in \mathbb{N}$), by fixing a linear basis, we can identity

$$\mathrm{GL}_{\mathbb{K}}(V) \;=\; \{n \times n \text{ matrices } A \text{ with coefficients in } \mathbb{K} \text{ and } \det A \neq 0\},$$

the law of composition getting the matrix multiplication and the neutral element in the identity matrix. This is the $n \times n$ general linear group, we write $\mathrm{GL}_n(\mathbb{K})$ for this.

2.2. Group actions

Definition 3 *Let $(G, \cdot, 1)$ be a group, and X a set. A action of G on X is a law of composition $G \times X \to X, (g, x) \mapsto gx$, such that associativity, that is $(g \cdot g')x \;=\; g(g'x)$, and unitality, that is $1x \;=\; x$, are satisfied for all $g, g' \in G, x \in X$.*

Definition 4 *Let $(G, \cdot, 1)$ be a group, and V a vector space over a field $\mathbb{K}$. A action of G on V is called linear action iff for all $g \in G$ the induced map $V \to V, v \mapsto gv$ is $\mathbb{K}$-linear.*

Definition 5 *Let G acting on a set X. The orbit of $x \in X$ is defined as*

$$Gx \;:=\; \{y \in X \,:\, \exists\, g \in G \text{ such that } y = gx\} \;\subseteq\; X.$$

Remark 6 *The orbits for a group action are equivalence classes for the relation $x \sim x'$ iff $x' \in Gx$. In particular, X is a union of disjoint orbits.*

2.3. Abstract symmetry

In this paper, we are interested in the applications of group actions on sample spaces. We use the basic notions and results of algebraic geometry as in Shafarevich and Reid (1994). We will work with the following assumptions throughout the paper:

- The sample space X is an affine algebraic variety over an algebraically closed field $\mathbb{K}$.
- The group G is an affine algebraic group over $\mathbb{K}$ that acts regularly on X, meaning that the action preserves the algebraic structure of X. In short: X is a G-variety.

We have a canonical linear action of G on the coordinate ring $\mathbb{K}[X] = \{f : X \to \mathbb{K} : f \text{ is regular}\}$ by $(g.f)(x) := f(g^{-1}x)$, $g \in G$, $f \in \mathbb{K}[X]$, $x \in X$ (Kraft and Wiedemann, 1985).

Definition 7 *The invariant ring is defined as*

$$\mathbb{K}[X]^G \ := \ \{f \in \mathbb{K}[X] \ : \ g.f = f \textit{ for all } g \in G\} \ \subseteq \ \mathbb{K}[X].$$

Finally, we can give a rigorous definition of *symmetry*:

Definition 8 *A symmetry is a non-constant function $\chi \in \mathbb{K}[X]^G$. Symmetries $\{\chi_1, \ldots, \chi_n\}$ are called fundamental symmetries iff they are algebraic independent and generate $\mathbb{K}[X]^G$. In the linear case, that is X is a vector space and the action is linear, we say a symmetry is linear iff it is a homogeneous polynomial of degree 1.*

Remark 9 *We always have the inclusion $\iota : \mathbb{K}[X]^G \hookrightarrow \mathbb{K}[X], f \mapsto f$. If additionally G is finite, we have the* Reynolds *operator*

$$r : \mathbb{K}[X] \to \mathbb{K}[X]^G, f \mapsto \frac{1}{|G|} \sum_{g \in G} g.f$$

Since $r \circ \iota = \mathrm{id}_{\mathbb{K}[X]^G}$ is fulfilled, r is surjective.

Symmetry is a concept that can be defined within a broader framework. We have rigorously discussed this in Dönmez (2023).

3. A toy example

In this section, we illustrate the main idea with a toy example. We show how we can apply group actions and symmetries to discover latent causes in the data.

Assume we give you the numerical series $\mathbf{0}, \mathbf{1}, \mathbf{2}, \ldots$ and you have to continue this. For sure, there are infinitely many possibilities to continue this series. But the first continuation you have in mind for certain is $\mathbf{0}, \mathbf{1}, \mathbf{2}, \mathit{3}, \mathit{4}, \mathit{5}, \ldots$.

If we assume that there should exist a relationship between two immediately following terms in the sequence, then it makes sense to start grouping the numerical series into 2 dimensional vectors:

$$\begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 2 \end{pmatrix}.$$

Therefore, our first candidate for the sample space is $X = \mathbb{K}^2$ with $\mathbb{K} = \mathbb{C}$. To estimate the next number x for the given numeric series $0, 1, 2$, we ask the question, for which numbers $x \in \mathbb{C}$, the vector

$$\begin{pmatrix} 2 \\ x \end{pmatrix} \text{ aligns with the previously inferred cluster } U = \left\{ \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 2 \end{pmatrix} \right\} \subseteq X.$$

The first key idea is that the clusters inferred from previous experiences are deduced by (the closures of) the *orbits* of some group action. In particular, we need a G-variety. Theorem 12 and Theorem 13 give evidence that we can look for some general linear group G and a linear action.

So, we are looking for a group G and

$$\text{a linear action on } V := X \text{ such that } \quad U \subseteq G \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad \text{holds.}$$

A canonical candidate for the group is

$$G := \{ f : V \to V \ : \ f \text{ is a bij. linear function, } f(U) \subseteq U \} \subseteq \mathrm{GL}_{\mathbb{K}}(V)$$

with the law of composition being the composition of maps and the action is just the evaluation, that is $f.v := f(v)$ $(f \in G, v \in V)$. Because the vectors in U form a $\mathbb{K}$-basis of V we can easily compute:

$$G = \left\{ \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} -2 & 1 \\ -3 & 2 \end{pmatrix} \right\} \ \subseteq \ \mathrm{GL}_2(\mathbb{K}).$$

Remark 10 *The matrices are identified as linear maps via the canonical basis.*

In this example $|G| = 2, \mathbb{K}[V] = \mathbb{K}[X_1, X_2]$ and by using the Reynolds operator (Remark 9) we can easily calculate the symmetry

$$\begin{aligned} \chi(X_1, X_2) \ := \ 2 \cdot r(X_1) \ &= \ X_1 \ + \ (-2X_1 + X_2) \\ &= \ X_2 - X_1 \ \in \ \mathbb{K}[V]^G. \end{aligned}$$

In this way, we discover the latent cause deduced by the symmetry and we can pursue the given numerical series: From the previous experiences

$$\chi(0, 1) \ = \ 1, \ \ \chi(1, 2) \ = \ 1.$$

we conclude that $\chi(2, x) = 1$ has to be fulfilled. Therefore, $x = 3$.

4. Pattern recognition - Continuing numerical series

In this section, we extend the approach applied to the toy example by demonstrating how we can handle unexpected information, such as a prediction error, to adjust the clustering of the data.

4.1. Non-linear example

We would like to generalize the procedure by looking at the following example. Assume we ask to continue the numerical series: $0, 1, 4, \ldots$. We proceed in the same way as in the previous section and end up in the following setting (Appendix C):

$$U = \left\{ \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 4 \end{pmatrix} \right\}, \quad G = \left\{ \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} -4 & 1 \\ -15 & 4 \end{pmatrix} \right\} \subseteq \mathrm{GL}_2(\mathbb{K}),\ V = X = \mathbb{K}^2.$$

The *fundamental linear symmetry* (unique up to a scalar $\neq 0$) is (Appendix C)

$$\chi(X_1, X_2) \;=\; X_2 - 3X_1.$$

Concluded from the fundamental linear symmetry our approach will predict *13* as the next term. Assume we can give the algorithm the feedback:

"Your prediction *13* is not correct, the next term should be 9!"

We see that 9 does not align with the previously inferred cluster. In this case, we need to adjust symmetries. Combined we have the following start of the series $\mathbf{0}, \mathbf{1}, \mathbf{4}, \mathbf{9}, \ldots$ and you have to continue this. The first continuation you have in mind for certain is

$$\mathbf{0}, \mathbf{1}, \mathbf{4}, \mathbf{9}, \mathit{16}, \mathit{25}, \mathit{36}, \mathit{49}, \ldots.$$

An ad-hoc symmetry for this continuation in two variables (for adjacent terms) can be expressed as $\chi(X_1, X_2) = \sqrt{X_2} - \sqrt{X_1}$. However, this expression is not polynomial and therefore does not satisfy the definition of a symmetry (Definition 8). In other words, the proposed framework only permits polynomial terms. Lemma 11 gives further evidence that we need to extend the dimension of the setting.

Inspired by the constructive proof of Theorem 13 given in Kraft and Wiedemann (1985), we extend the sample space from the vector space of homogeneous polynomials of degree 1 (2-dimensional) to the vector space of homogeneous polynomials of degree 2 (3-dimensional), that is we augment due the projection $(x^2, xy, y^2) \mapsto (x^2, y^2)$. We embed the grouped data into the new sample space:

$$U = \left\{ \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 4 \end{pmatrix}, \begin{pmatrix} 4 \\ 9 \end{pmatrix} \right\} \;\rightsquigarrow\; U = \left\{ \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 2 \\ 4 \end{pmatrix}, \begin{pmatrix} 4 \\ 6 \\ 9 \end{pmatrix} \right\}.$$

We proceed in the same way as in the previous section and end up in the following setting (Appendix C):

$$G = \left\{ \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} -9 & \frac{9}{2} & 1 \\ -12 & 5 & 2 \\ -14 & \frac{7}{2} & 4 \end{pmatrix}, \begin{pmatrix} -11 & \frac{15}{2} & 0 \\ -16 & 11 & 0 \\ -20 & \frac{25}{2} & 1 \end{pmatrix}, \right.$$

$$\left. \begin{pmatrix} 7 & \frac{-11}{2} & 1 \\ 12 & -10 & 2 \\ 18 & \frac{-33}{2} & 4 \end{pmatrix}, \begin{pmatrix} 13 & \frac{-29}{2} & 4 \\ 20 & -22 & 6 \\ 28 & \frac{-63}{2} & 9 \end{pmatrix}, \begin{pmatrix} 9 & -12 & 4 \\ 12 & -17 & 6 \\ 16 & -24 & 9 \end{pmatrix} \right\} \subseteq \mathrm{GL}_3(\mathbb{K}),$$

$V = X = \mathbb{K}^3$. The *fundamental linear symmetry* (unique up to a scalar $\neq 0$) is (Appendix C)

$$\chi(X_1, X_2, X_3) \;=\; X_1 - 2X_2 + X_3.$$

Concluded from the fundamental linear symmetry our approach will predict $\mathit{16}, \mathit{25}, \mathit{36}, \mathit{49}, \ldots$ as the next terms!

5. Boundaries of generalization

Gershman and Niv (2013) conducted a series of experiments in which participants estimated the number of colored circles that were displayed in a random spatial configuration on a computer screen. Participants were given 5 seconds to estimate the number of circles on the screen using a two-digit entry on the keyboard. If no response was entered within this time limit, a message indicated that the response was too slow and the trial was subsequently excluded from data analysis. The circles were displayed on the screen throughout the 5-second response interval. Upon entering a response, participants received feedback indicating the correct number of circles. Two colors were used to characterize each trial, and all circles on a given trial were displayed in the same color. Instructions made no mention of color to participants. They continued this experiment for about half an hour. The number of circles was drawn from a color-specific (Gaussian) distribution. When the distributions associated with each color overlapped substantially, participants' estimates were biased toward values intermediate between the two means. This suggests that subjects ignored the color of the circles and grouped different-colored stimuli into one perceptual category. However, when the distributions associated with each color overlapped minimally, the bias was reduced (Gershman and Niv, 2013), and this suggests that color was used as a cue for categorization. We implicitly adopt the rational analysis approach (Anderson, 1990) to this task and demonstrate how these qualitative patterns can arise from symmetries!

5.1. The sample space

Instead of circles, we assume that *dots/boxes* were displayed on the screen (Figure 3). This mild adjustment should not change the qualitative result of the experiments.

Then, a random scattering of boxes on each trial (Figure 3) can be identified as a matrix of a fixed dimension $n \times m$ ($n, m \in \mathbb{N}$) (deduced from the screen size) with entries $0, 1$. The center of a box is represented by the entry 1 within the matrix. Further, we flat the matrix to a vector v of dimension $N = n \cdot m$. Then, the number of boxes on each trial $v = (v_1, \ldots, v_N) \in \mathbb{K}^N$ is just given by the sum $\sum_{i=1}^{N} v_i$. To capture the information on the color used for the boxes, we need one further dimension. In other words, the sample space is the vector space $V = X := \mathbb{K} \times \mathbb{K}^N$, and each trial is represented as a vector $v = (v_0, v_1, \ldots, v_N)$, where v_0 encrypts the color used for the boxes in this trial, and $(v_1, \ldots, v_N)$ captures the random spatial configuration of the boxes.

5.2. Our computational approach

Similar to the previous examples in Section 3 and Section 4, we require clustering of the sample space as an initial step. In the examples presented in Section 3 and Section 4, the clustering was implicitly defined by U.

Here, first, we need to learn some clustering from the feedback indicating the correct number of boxes. This is the reason, if you like, why the experiment goes on for so long, namely as mentioned about half an hour. At the beginning of the experiment, the subject learns from the feedback. Subsequently, latent structure learning commences and is reflected in the estimates. Our computational approach replicates this procedure. We start

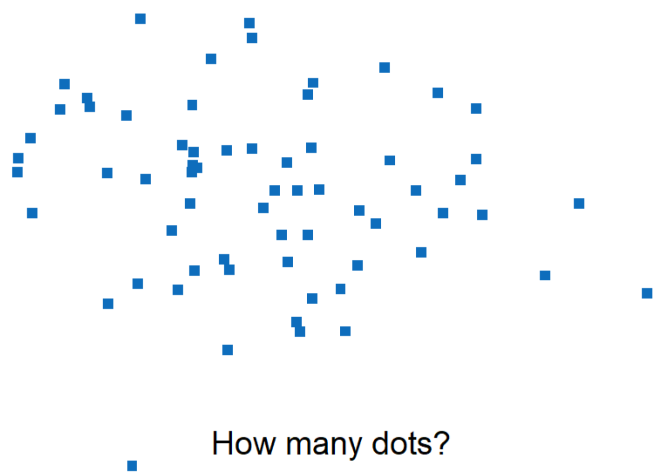

Figure 3: Example trial. Assuming participants were shown a random scattering of dots/boxes (instead of circles) on each trial and asked to estimate the number of dots. All the dots on each trial were of the same color, and the number of dots was randomly drawn from a Gaussian distribution specific to that color.

a simulation following the procedure as described in (Gershman and Niv, 2013, Experiment 1 - Procedure) and use the kernel density estimation for an initial clustering (Appendix C).

5.2.1. Minimal overlap

In Figure 4 you can see the kernel density plot when the distributions associated with each color overlap minimally. Clearly, *two* clusters can be derived from this. The key idea is that the clusters inferred from previous experiences are deduced by *orbits* of some group action. We proceed in the same way as in the previous sections and end up with a subgroup $G \subseteq \mathrm{Sym}_N \subset \mathrm{GL}_N(\mathbb{K})$, and the linear action $\sigma.(v_0, v_1, \ldots, v_N) = (v_0, v_{\sigma(1)}, \ldots, v_{\sigma(N)})$, $\sigma \in G$, $v = (v_0, v_1, \ldots, v_N) \in V$. By using the Reynolds operator (Remark 9) we can easily calculate the linear symmetry $\chi(X_0, X_1, \ldots, X_N) := r(X_0) = X_0$. Another linear symmetry is $\phi(X_0, X_1, \ldots, X_N) = \sum_{i=1}^{N} X_i$, that is the polynomial describing the number of boxes. Each symmetry can adduce the "cluster". Since the symmetry χ is less Kolmogorov complex (Li et al., 2008) than ϕ, we prefer to use χ instead of ϕ. If you look closely, you can see that the symmetry χ indicates the use of color as a cue for categorization. This is exactly what the authors Gershman and Niv (2013) of the study expected from the participants, and it is what happened. We rediscover computationally the (hidden) latent cause "color" *naturally* from symmetries in data!

5.2.2. Substantially overlap

When the distributions associated with each color overlap substantially, kernel density estimation results in *one* cluster (Figure 4). Statistics cannot distinguish between groups with too similar mean. We proceed in the same way as in the previous sections and end up with only *one* linear symmetry $\phi(X_0, X_1, \ldots, X_N) = \sum_{i=1}^{N} X_i$, which is the polynomial describing the number of boxes. This symmetry indicates the "cluster". However, since $N \gg 0$ and humans have limited computational abilities to evaluate ϕ within 5 seconds, ϕ cannot be used. Therefore, in such a case, our computational approach will rely on its experience for estimation by sampling from the learned distribution. If the estimation is done by sampling from the learned distribution, statistical tests will not detect a significant difference from the correct answer in the further course of this experiment. In this way, the estimate is further encouraged by sampling the learned distribution and the estimate converges to the overall mean. This result occurred also in the experiment with humans (Gershman and Niv, 2013).

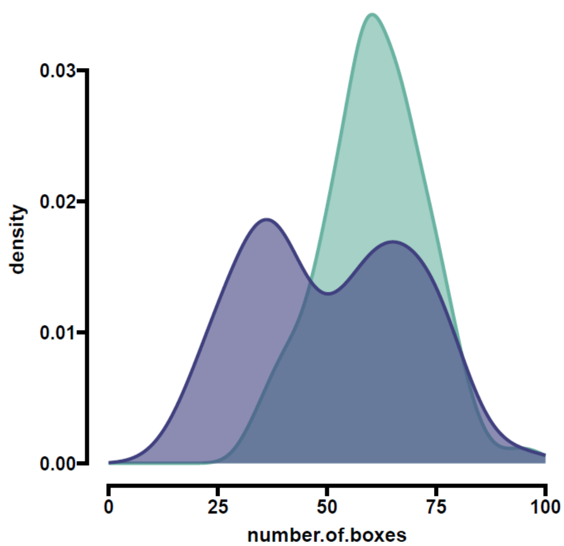

Figure 4: Kernel density estimation plot deduced from the simulation of 160 trials for *Baseline* vs. *Low mean alternative* and *Baseline* vs. *High mean alternative* (Gershman and Niv, 2013, Experiment 1).

6. Discussion

6.1. Related work

Our approach tries to solve nonlinear problems with linear symmetries in some augmented linear space. Therefore, it is related to *kernel machines* (Aizerman, 1964; Shawe-Taylor and Cristianini, 2004). But differ fundamentally in their intentions and permitted opportunities.

Our approach does not aim to find *decision boundaries by hyperplanes* in some linear space (*feature space*). We look for linear equations (symmetries) that infer the latent structures with statistical certainty (that is non-deterministically). Our approach also allows the intersection of several hyperplanes for this purpose.

6.2. Contributions, Limitations

In this work, we presented a computational approach whose output closely matches human responses in simple tasks that require inductive reasoning and, in particular, generalization. We wish to emphasize two main points about this work. Firstly, we follow the advice of Lake et al. (2017) and start with very basic concepts from intuitive physics and psychology. We then translate these concepts into a computational framework via (linear) representations of groups. The notion of abstract symmetry then allows inductive reasoning.

Concepts from intuitive physics and psychology are captured by the abstract mathematical structure of a group and its representation! In this paper, we started with two very basic concepts (Section 1) that are captured by linear representations of the symmetric group Sym_n ($n \in \mathbb{N}$) (Lemma 11). Here we will be confronted by the main limitation $|\mathrm{Sym}_n| = n!$. The size of the group grows very fast! However, this limitation can be overcome by starting with more additional structural concepts. The intuition behind this is that more complex concepts simplify the groups. But then the symmetries become more opaque. And here we are already at the next limit of this work. New concepts mean new groups and representations, that is we cannot simply reuse Lemma 11. This in turn is linked to memory modification (sample space augmentation), which will also require adaptation.

Finally, we would like to note here where exactly the geometry took place in this frame. The main idea behind noncommutative geometry lies in the duality between algebra and geometry. In the commutative case, we can switch to the algebra side by looking at the coordinate ring. This is exactly what we are doing in this paper. We do not dive into estimations about the topology and geometry of the sample space, but instead, we switch to the algebra side by focusing on symmetries (Definition 8), the coordinate ring of some categorical/geometric quotient (Kraft and Wiedemann, 1985). Furthermore, the linear symmetries are related to the tangent space of the orbit closures.

6.3. Future work

Integrating more principles from intuitive physics and Gestalt psychology to address more complex tasks with the presented approach is the logical next step for this line of research. In addition, a detailed description of the relationship between principles and group representations to facilitate the introduction of new principles.

Acknowledgments

The author gratefully thanks the Leibniz-Gemeinschaft for its postdoc career development program, and especially the IUF – Leibniz Research Institute for Environmental Medicine for its support. The author was funded by the European Union's Horizon 2020 Research and Innovation Program, under the Grant Agreement number: 825759 of the ENDpoiNTs project.

References

A Aizerman. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and remote control*, 25:821–837, 1964.

John Robert Anderson. *The adaptive character of thought*. Psychology Press, 1990.

Jeff Bezanson, Alan Edelman, Stefan Karpinski, and Viral B Shah. Julia: A fresh approach to numerical computing. *SIAM Review*, 59(1):65–98, 2017. doi: 10.1137/141000671. URL https://epubs.siam.org/doi/10.1137/141000671.

Nick Chater and Paul Vitányi. Simplicity: a unifying principle in cognitive science? *Trends in cognitive sciences*, 7(1):19–22, 2003.

Harm Derksen and Gregor Kemper. *Computational invariant theory*. Springer, 2015.

Arif Dönmez. On the ambiguity in classification. In *NeurIPS Workshop on Symmetry and Geometry in Neural Representations*, pages 158–170. PMLR, 2023.

Jacob Feldman. The simplicity principle in human concept learning. *Current directions in psychological science*, 12(6):227–232, 2003.

Samuel J Gershman and Yael Niv. Perceptual estimation obeys occam's razor. *Frontiers in psychology*, 4:623, 2013.

Samuel J Gershman, Kenneth A Norman, and Yael Niv. Discovering latent causes in reinforcement learning. *Current Opinion in Behavioral Sciences*, 5:43–50, 2015.

Samuel J Gershman, Marie-H Monfils, Kenneth A Norman, and Yael Niv. The computational nature of memory modification. *Elife*, 6:e23763, 2017.

K Koffka. Principles of gestalt psychology. 1935.

Wolfgang Köhler. Gestalt psychology. *Psychologische Forschung*, 31(1):XVIII–XXX, 1967.

Hanspeter Kraft and A Wiedemann. *Geometrische methoden in der invariantentheorie*. Springer, 1985.

Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. Building machines that learn and think like people. *Behavioral and brain sciences*, 40:e253, 2017.

Ming Li, Paul Vitányi, et al. *An introduction to Kolmogorov complexity and its applications*, volume 3. Springer, 2008.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2023. URL https://www.R-project.org/.

Igor Rostislavovich Shafarevich and Miles Reid. *Basic algebraic geometry*, volume 2. Springer, 1994.

John Shawe-Taylor and Nello Cristianini. *Kernel methods for pattern analysis*. Cambridge university press, 2004.

Bernd Sturmfels. *Algorithms in invariant theory*. Springer Science & Business Media, 2008.

OSCAR Symbolic Tools. *OSCAR – Open Source Computer Algebra Research system, Version 0.12.0-DEV*. The OSCAR project, 2023. URL https://www.oscar-system.org.

Henry M Wellman and Susan A Gelman. Cognitive development: Foundational theories of core domains. *Annual review of psychology*, 43(1):337–375, 1992.

Appendix A. Some basic facts

Let's start with some basic observations from linear algebra:

Lemma 11 *Let $V = \mathbb{K}^n$ ($n \in \mathbb{N}$), and $U = \{b_1, \ldots, b_n\} \subseteq V$ a $\mathbb{K}$-basis of V. Then the group*

$$G := \{f : V \to V \ : \ f \text{ is a bij. linear function}, \ f(U) \subseteq U\} \subseteq \mathrm{GL}_{\mathbb{K}}(V)$$

is the image of the following group homomorphism

$$\mathrm{Sym}_n \longrightarrow \mathrm{GL}_n(\mathbb{K}),\ \sigma \mapsto T^{-1} P_\sigma T,$$

where $T = (b_1 | \ldots | b_n) \in \mathrm{GL}_n(\mathbb{K})$, and P_σ denotes the permutation matrix associated to σ.

Proof Is a classical change of basis application. ■

Let us now move on to the geometric invariant theory:

Theorem 12 *Every affine algebraic group G is a Zariski-closed subgroup of the general linear group $\mathrm{GL}_n(\mathbb{K})$ for some $n \in \mathbb{N}$.*

Proof See Kraft and Wiedemann (1985). ■

Theorem 13 *Every G-variety Z is G-isomorphic to a G-stable closed subvariety of a vector space V, on which G acts linearly and rationally.*

Proof See Kraft and Wiedemann (1985). ■

Appendix B. Outline of a strategy blueprint

We summarize our approach from the previous sections in a strategy blueprint. There the function Π and the multimodal distribution $\Pi_*\mathbb{P}$ formalize the initial clustering of the sample space. In Section 5 Π is the "number of boxes" and the multimodal distribution $\Pi_*\mathbb{P}$ is the kernel density estimation (Figure 4). Please note that the proposed strategy is only a very rough sketch and does not need to terminate in this formulation. Crucial steps, such as augmenting the sample space to a vector space, are left intentionally vague. Different data augmentation strategies can be followed. For example, we can take the vector space of homogeneous polynomials of degree $N-1$ as the model vector space, we want to embend in. In this case, the strategy of expansion by increasing the degree of the homogeneous polynomials was successfully applied in Section 4. But other strategies are conceivable as well. We can extend to the multivariate case, that is Π with codomain $\mathbb{K}^m$ for some $m \in \mathbb{N}$ and multimodal distributions on $\mathbb{K}^m$.

Strategy blueprint: Discovering latent causes by symmetries

Given a *sample space* X, that is an affine variety over a field $\mathbb{K} \in \{\mathbb{R}, \mathbb{C}\}$, finite samples $\Omega \subseteq X$ and a function $\Pi : \Omega \to \mathbb{K}$.

1. Apply kernel density estimation on $\Pi(\Omega)$, and denote by $\Pi_*\mathbb{P}$ the estimated *multimodal distribution* on $\mathbb{K}$.

2. Let $\Omega = U_1 \cup \ldots \cup U_m$ denote the clustering induced by the multimodal distribution $\Pi_*\mathbb{P}$.

3. Set $N = \dim X$.

    4. Augment X to the vector space $V := \mathbb{K}^N$.
    5. Determine a subgroup $G \subseteq \mathrm{GL}_N(\mathbb{K})$ that aligns orbits induced by the linear action of G with the cluster as closely as possible.
    6. Determine the fundamental linear symmetries $\chi_1, \ldots, \chi_l \in \mathbb{K}[V]^G$.
    7. Apply kernel density estimation on $\chi_i(\Omega)$, and denote by $\chi_*^{(i)}\mathbb{P}$ the estimated *multimodal distribution* on $\mathbb{K}$ for all $i = 1, \ldots, l$.
    8. Determine the set of fundamental symmetries $FSI \subseteq \{1, \ldots, l\}$, for which the Kullback–Leibler divergence (Kullback and Leibler, 1951)

       $$D_{\mathrm{KL}}(\Pi_*\mathbb{P} : \chi_*^{(i)}\mathbb{P}) \text{ is within a predefined acceptable range}$$

       for all $i \in FSI$.
    9. If $FSI \neq \emptyset$,

       the linear symmetry that adduces the clustering is given by χ_i, $i \in FSI$ with the minimal Kolmogorov complexity,

       otherwise

       increment N and return to step 4.

Appendix C. Some code

Code used for Section 3:

```
using␣LinearAlgebra
using␣Permutations

v1␣=␣[0␣;␣1]
v2␣=␣[1␣;␣2]
V␣=␣hcat(v1,␣v2)

T␣=␣inv(V)

H␣=␣Set()
for␣p␣in␣PermGen(2)
␣␣push!(H,␣inv(T)*T[p.data,␣:])
end
H

using␣Oscar

S␣=␣matrix_space(QQ,␣2,␣2)
M1␣=␣S(BigInt[-2.0␣1.0;␣-3.0␣2.0])
M2␣=␣S(BigInt[1.0␣0.0;␣0.0␣1.0])
G␣=␣matrix_group(M1,␣M2)
IR␣=␣invariant_ring(G)
fundamental_invariants(IR)
```

Code for Section 4:
First iteration:

```
using␣LinearAlgebra
using␣Permutations

v1␣=␣[0␣;␣1]
v2␣=␣[1␣;␣4]
V␣=␣hcat(v1,␣v2)

T␣=␣inv(V)

H␣=␣Set()
for␣p␣in␣PermGen(2)
␣␣push!(H,␣inv(T)*T[p.data,␣:])
end
H

using␣Oscar
```

```
S = matrix_space(QQ, 2, 2)
MM1 = S(BigInt[1.0 0.0; 0.0 1.0])
MM2 = S(BigInt[-4.0 1.0; -15.0 4.0])
G = matrix_group(MM1, MM2)
IR = invariant_ring(G)
fundamental_invariants(IR)
```

Second iteration:

```
using LinearAlgebra
using Permutations

v1 = [0 ; 0 ; 1]
v2 = [1 ; 2 ; 4]
v3 = [4; 6 ; 9]
V = hcat(v1, v2, v3)

T = inv(V)

H = Set()
for p in PermGen(3)
  push!(H, inv(T)*T[p.data, :])
end
H

using Oscar

S = matrix_space(QQ, 3, 3)
MM1 = S([-9//1 9//2 1//1; -12//1 5//1 2//1; -14//1 7//2 4//1])
MM2 = S([-11//1 15//2 0//1; -16//1 11//1 0//1; -20//1 25//2 1//1])
MM3 = S([1//1 0//1 0//1; 0//1 1//1 0//1; 0//1 0//1 1//1])
MM4 = S([7//1 -11//2 1//1; 12//1 -10//1 2//1; 18//1 -33//2 4//1])
MM5 = S([13//1 -29//2 4//1; 20//1 -22//1 6//1; 28//1 -63//2 9//1])
MM6 = S([9//1 -12//1 4//1; 12//1 -17//1 6//1; 16//1 -24//1 9//1])
G = matrix_group(MM1, MM2, MM3, MM4, MM5, MM6)
IR = invariant_ring(G)
fundamental_invariants(IR)
```

For Section 5 using `R` (R Core Team, 2023):

```
library(ggplot2)
library(ggprism)
library(ggnewscale)

baseline <- rnorm(80, mean = 65, sd = 10)
```

```
alternative␣<-␣rnorm(80,␣mean␣=␣35,␣sd␣=␣10)

geordnet␣<-␣c(baseline,␣alternative)

dt␣<-␣sample(geordnet)
df␣<-␣data.frame("number.of.boxes"␣=␣dt)

alternative_a␣<-␣rnorm(80,␣mean␣=␣55,␣sd␣=␣10)
geordnet_a␣<-␣c(baseline,␣alternative_a)

dt_a␣<-␣sample(geordnet_a)
df_a␣<-␣data.frame("number.of.boxes"␣=␣dt_a)

gg␣<-␣ggplot(df_a,␣aes(x=number.of.boxes))␣+
␣␣geom_density(size␣=␣1.2,␣colour␣=␣"#69b3a2",␣fill␣=␣"#69b3a2",␣alpha=0.6)␣+
␣␣geom_density(data␣=␣df,␣mapping␣=␣aes(x=number.of.boxes),␣size␣=␣1.2,
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣colour␣=␣"#404080",␣fill␣=␣"#404080",␣alpha=0.6)␣+
␣␣theme_prism(base_size␣=␣16)␣+
␣␣scale_x_continuous(limits␣=␣c(0,␣100),␣guide␣=␣"prism_offset")␣+
␣␣scale_y_continuous(guide␣=␣"prism_offset")

plot(gg)
```