Concept frustration: Aligning human concepts and machine representations

Enrico Parisini
King's College London, UK
PharosAI, UK

Christopher J. Soelistyo
The Francis Crick Institute, UK

Ahab Isaac
University College London, UK

Alessandro Barp
University College London, UK
The Alan Turing Institute, UK

Christopher R.S. Banerji
King's College London, UK
The Alan Turing Institute, UK
christopher.banerji@kcl.ac.uk

Abstract

Aligning human-interpretable concepts with the internal representations of modern machine learning systems remains a central challenge for interpretable AI. We introduce a geometric framework to compare supervised human concepts with unsupervised representations derived from foundation models. We formalise concept frustration: a mismatch that arises when an unobserved concept induces relationships between known concepts that cannot be made consistent within the existing ontology. We develop task-aligned similarity measures that detect this phenomenon, and show that frustration is identifiable in task-aligned geometry while conventional Euclidean comparisons fail. Under a linear–Gaussian generative model, we derive a closed-form expression for Bayes-optimal concept-based classifier accuracy, decomposing predictive signal into known and unknown contributions and identifying where frustration impacts performance. Experiments on synthetic data and real language and vision tasks demonstrate task-aligned concept frustration in foundation model representations. Across 845 frustrated concept triples in vision tasks, the maximally frustrating unsupervised direction recovers omitted concepts better than a label-permuted control. These results provide a principled framework for diagnosing representational inconsistency and localising candidate omitted concepts in interpretable AI systems.

1 Introduction

Human-level interpretability in artificial intelligence (AI) is critical for high-consequence domains such as medicine, scientific discovery and criminal justice [1–3]. In such settings, errors can have catastrophic consequences and accountability, as well as emerging regulation, requires not only accurate predictions but transparent reasoning [4, 5].

There are two prominent strategies for securing transparency. (1) Post-hoc feature attribution, where we train a black box model, then perform techniques such as SHAP [6], LIME [7] or Integrated Gradients [8, 9] to identify task-relevant elements of the input. Similarly, mechanistic interpretability decomposes feature representations of trained models using methods such as sparse autoencoders

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

(SAEs) [10–12]. (2) Interpretability-by-design, where we optimise aspects of the model architecture, training regime or input set, for interpretability pre-hoc. This includes the concept bottleneck model (CBM), where we constrain the model to learn a set of supervised concepts that are pre-defined by human domain experts and therefore guaranteed to be interpretable [13].

Both strategies have limitations. Post-hoc explanations need not provide faithful or comprehensive accounts of model behaviour[1], and SAE derived features often exhibit polysemanticity and feature superposition, degrading interpretability [10–12]. In CBMs, the bottleneck constraint that ensures interpretability also limits its model expressivity, undermining performance when task-relevant features are excluded from the supervised concepts [14, 15].

There are at least two solutions to this problem in CBMs. The first is to allow predictive information to flow through an auxiliary pathway that bypasses the concept bottleneck [16–20]. These "concept side-channel models" increase expressivity but at the expense of interpretability. The second is to optimise the concept set - or *ontology* - for the task at hand. However, constructing an adequate ontology first requires a reliable diagnostic for cases when the current ontology is inadequate. Recent work has addressed this by comparing supervised CBM concepts with unsupervised SAE features through geometric alignment in Euclidean space [21]. While foundational, such approaches rely on restrictive assumptions: that supervised concepts are statistically independent, that alignment should be assessed in a task-agnostic geometry and that particular parameter constraints (such as non-negativity of concept codes) are preserved. These assumptions are often violated in practice.

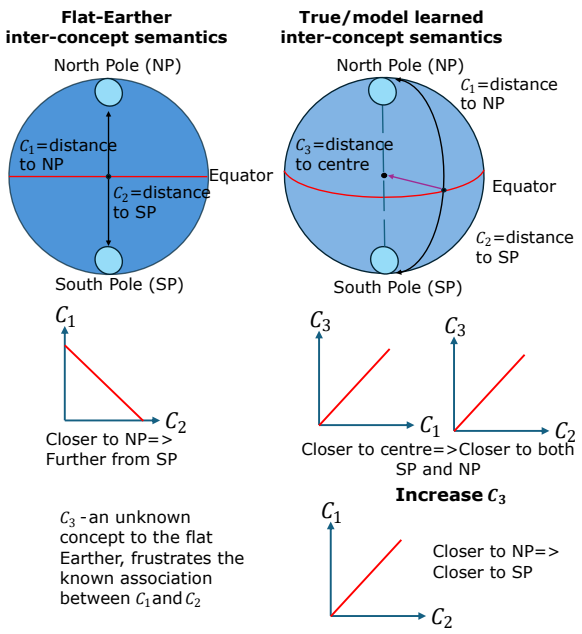

Figure 1: **Concept frustration.** Consider a flatworlder training a model to predict a position on Earth from a complex signal (such as from a satellite), in an interpretable manner. They suggest two known concepts as C_1=distance from the North Pole and C_2=distance from the South pole. They expect these to be anti-correlated. Their trained model is highly accurate of position, but reasons using an unknown concept C_3, which frustrates their known inter-concept semantics. Increasing C_3 forces C_1 and C_2 to be positively correlated. Once C_3 is understood as distance to the curved Earth's core, frustration makes sense and the understanding of the system is improved.

Here we introduce a diagnostic framework for incomplete concept ontologies that addresses limitations of prior work. We identify a phenomenon we term *concept frustration*, in which information that is predictive of the task but absent from the supervised concept set induces contradictions between known concepts, revealing that the ontology is incomplete. We show that detecting frustration requires a task-aligned geometry, rather than a Euclidean geometry, and that frustration degrades both performance and interpretability in CBMs. In real-world language and vision tasks, incorporating a previously unobserved frustrating concept reorganises the geometry of existing concept representations, reducing their task-aligned frustration. Extending the vision analysis to 845 frustrated concept triples, we find that reductions in scalar frustration are not specific to adding the true omitted concept. However, the maximally frustrating SAE direction recovers the omitted concept better than a matched label-permuted control, supporting its use to identify candidate omitted semantic structure.

Together, these results provide a pathway for diagnosing task-aligned representational inconsistency and identifying candidate missing conceptual dimensions to guide interpretation and ontology refinement.

2 Background and motivation

2.1 Flatworld: a motivating example

To illustrate frustration, consider a thought experiment: a flatworlder navigates the Earth using two trusted concepts: distance to the North Pole and distance to the South Pole. They expect these concepts to oppose each other, so that moving closer to one Pole moves them farther from the other. However, due to the Earth's curvature, a traveller who moves closer to the Earth's centre, say by diving in an ocean, can move closer to both Poles simultaneously (Figure 1). This introduces an unknown concept, depth, that is not part of the flatworlder's ontology but distorts how known concepts relate. We refer to such a hidden, task-relevant concept as *frustrating*: without it, the flatworlder cannot form a coherent picture of their world.

2.2 Background and models

Let X denote a space of input data. An arbitrary feature extractor (e.g., a foundation model) maps inputs to an activation space $f : X \to A \subset \mathbb{R}^r$. For $\mathbf{x} \in X$, we write $\mathbf{a} = f(\mathbf{x}) \in A$ for its activation vector. We treat A as a dense representation of features relevant for predicting downstream tasks. Let Y denote an outcome variable and $C = \bigcup_{j=1}^{k} C_j$ a set of k concepts. We assume that Y and C are associated, such that $p(Y|C) \neq p(Y)$. Similarly we assume that A and C are associated, i.e., $p(C|A) \neq p(C)$, which naturally implies an association between A and Y such that $p(Y|A) \neq p(Y)$. We partition C into k_{known} known concepts, $C_{\text{known}} = \bigcup_{j=1}^{k_{\text{known}}} C_j$, and $k - k_{\text{known}}$ unknown concepts, $C_{\text{unknown}} = \bigcup_{j=k_{\text{known}}+1}^{k} C_j$. For each $\mathbf{a} \in A$, we assume C_{known} is observed, while C_{unknown} is not.

A common goal of interpretable modelling is to predict Y from A via intermediate concepts C, enabling human-understandable reasoning. We consider three modelling strategies. Further architectural and training details are provided in Appendix A.

Black-box model. We approximate $p(Y|A)$ using a model trained on $(\mathbf{a}_i, y_i)_{i=1}^{N_{\text{train}}} \subset (A, Y)$. In our implementation, we use a single hidden layer of width h with ReLU activation and consider Y as a binary task (extension to non-binary is straightforward). The hidden layer outputs logits $l(\mathbf{a})$, from which the predicted task label is obtained via a sigmoid transformation, $\hat{y} = \sigma(l(\mathbf{a}))$. This approach can achieve high predictive accuracy, but offers no interpretability.

Sparse autoencoder (SAE). We extract k_{SAE} unsupervised latent features from A, using a model trained on $(\mathbf{a}_i)_{i=1}^{N_{\text{train}}} \subset A$. An encoder $s : \mathbb{R}^r \to \mathbb{R}^{k_{\text{SAE}}}$ maps activations to the latent feature space $s(\mathbf{a}) = \text{ReLU}(\mathbf{W}_{\text{SAE}}\mathbf{a} + \mathbf{b}_{\text{SAE}}) \in \mathbb{R}^{k_{\text{SAE}}}$, where $\mathbf{W}_{\text{SAE}} \in \mathbb{R}^{k_{\text{SAE}} \times r}$ is a learned matrix of weights and $\mathbf{b}_{\text{SAE}} \in \mathbb{R}^{k_{\text{SAE}}}$ is a learned vector of biases, and a linear decoder reconstructs $\hat{\mathbf{a}} = s(\mathbf{a})\mathbf{D}$, where $\mathbf{D} \in \mathbb{R}^{k_{\text{SAE}} \times r}$ is a dictionary whose rows define atoms in activation space. The SAE decomposes A into unsupervised features and may loosely estimate $p(C|A)$, however these features are not conditioned on relevance to the task Y.

Concept bottleneck model (CBM). Known concepts are predicted from activations and subsequently used to predict the outcome. We consider here joint training from $(\mathbf{a}_i, \mathbf{c}_{\text{known},i}, y_i)_{i=1}^{N_{\text{train}}} \subset (A, C_{\text{known}}, Y)$, and assume A is a sufficiently processed representation that concepts are linearly deducible. The CBM thus learns a linear concept mapping $\hat{\mathbf{c}}_{\text{known}}(\mathbf{a}) = \mathbf{Q}\mathbf{a}$, with $\mathbf{Q} \in \mathbb{R}^{k_{\text{known}} \times r}$, together with a linear task head that maps the predicted concepts to a binary outcome via a sigmoid transformation, yielding $\hat{y}(\mathbf{a})$. The CBM estimates $p(C_{\text{known}}|A)$ and $p(Y|\hat{C}_{\text{known}})$. When the task depends strongly on C_{unknown}, predictive accuracy may degrade relative to the black-box model. However, the CBM provides direct interpretability by expressing predictions as linear functions of supervised concepts.

3 A task-aligned geometry to detect frustrating unsupervised concepts

3.1 Task-aware information geometry of activation space

The three models provide three outputs: (1) Black box model: An estimate of $p(Y|A)$. (2) SAE: A matrix $\mathbf{D} \in \mathbb{R}^{k_{\text{SAE}} \times r}$, describing a dictionary of k_{SAE} *unsupervised* concepts extracted from A,

represented as r-dimensional vectors. (3) CBM: A matrix $\mathbf{Q} \in \mathbb{R}^{k_{known} \times r}$ describing a dictionary of k_{known} *supervised* concepts extracted from A, represented as r-dimensional vectors.

We wish to compare the known concepts in $\mathbf{Q}$ to the unsupervised concepts in $\mathbf{D}$. The simplest way is using Euclidean geometry over $\mathbb{R}^r$, defining a standard inner product as $\langle \mathbf{q}_i, \mathbf{d}_j \rangle_E = \mathbf{q}_i^\top \mathbf{d}_j$ where $\mathbf{q}_i$, $\mathbf{d}_j$ are the ith and j^{th} rows of $\mathbf{Q}$ and $\mathbf{D}$ respectively, this measures alignment of the SAE and CBM derived concepts in Euclidean space. However, this geometry is not anchored to the predictive task. Many SAE features may be irrelevant to Y, and Euclidean alignment need not reflect task-relevant relationships. Consequently, Euclidean similarity does not guarantee semantically meaningful concept comparisons with respect to the outcome.

A more natural geometry is that of the statistical manifold $(A, p(y|\mathbf{a}))$, equipped with the Fisher information metric $F_A(\mathbf{a}) = \mathbb{E}_{y \sim p(y|\mathbf{a})} \left[\nabla_\mathbf{a} \log p(y|\mathbf{a}) \nabla_\mathbf{a} \log p(y|\mathbf{a})^\top \right]$. Under our single-layer black-box model, this metric admits a closed-form expression.

Theorem 1 (Closed-form Fisher metric for one-hidden-layer binary model)**.** *Consider the black-box model defined above with sigmoid output. The Fisher information metric on activation space satisfies* $F_A(\mathbf{a}) = p(\mathbf{a})(1 - p(\mathbf{a}))\, g(\mathbf{a}) g(\mathbf{a})^\top$, *where* $p(\mathbf{a}) \equiv p(y = 1|\mathbf{a}) = \sigma(l(\mathbf{a}))$ *and* $g(\mathbf{a}) = \nabla_\mathbf{a} l(\mathbf{a})$.

The proof and explicit form of $g(\mathbf{a})$ are provided in Appendix B. For fixed $\mathbf{a}$, $F_A(\mathbf{a})$ is a rank-one positive semi-definite matrix, as it is the outer product of $g(\mathbf{a})$ with itself. Thus the pointwise Fisher metric captures sensitivity only along a single direction in activation space. Moreover, whenever $p(\mathbf{a}) \in \{0, 1\}$ the prefactor vanishes and $F_A(\mathbf{a}) = 0$. Consequently, the Fisher geometry is expected to be more informative near the decision boundary.

We therefore define a global, task-aligned quadratic form by averaging over activations near the decision boundary. Let $A_{\text{ave}} := \{\mathbf{a} \in A_{\text{train}} : p(\mathbf{a}) \in [p_{\text{low}}, p_{\text{high}}]\}$. We define $\bar{F}_A = \frac{1}{|A_{\text{ave}}|} \sum_{\mathbf{a} \in A_{\text{ave}}} p(\mathbf{a})(1 - p(\mathbf{a}))\, g(\mathbf{a}) g(\mathbf{a})^\top$.

Equipped with $\bar{F}_A$, we define the task-aligned inner product $\langle \mathbf{q}_i, \mathbf{d}_j \rangle_{F_A} = \mathbf{q}_i^\top \bar{F}_A \mathbf{d}_j$. This inner product will detect alignment between SAE and CBM derived concepts in a geometry aligned to task prediction and can thus be expected uncover more task-relevant inter-concept semantics than a Euclidean geometry.

3.2 Defining and measuring concept frustration

We now formalise a framework to compare supervised and unsupervised concept directions under $\bar{F}_A$. We define two concept similarity matrices. First, the task aligned similarity between known and unsupervised concepts $\mathbf{S}^{F_A}$ with elements: $S_{ij}^{F_A} = \frac{\mathbf{q}_i^\top \bar{F}_A \mathbf{d}_j}{\|\mathbf{q}_i\|_{\bar{F}_A} \|\mathbf{d}_j\|_{\bar{F}_A}}$, where $\|\mathbf{v}\|_{\bar{F}_A} := \sqrt{\mathbf{v}^\top \bar{F}_A \mathbf{v}}$ and $S_{ij}^{F_A} \in [-1, 1]$ measures alignment between $\mathbf{q}_i$ and $\mathbf{d}_j$ in the task-aligned geometry. Second, the task-aligned similarities between supervised concepts $\mathbf{Z}^{F_A}$ with elements: $Z_{rl}^{F_A} = \frac{\mathbf{q}_r^\top \bar{F}_A \mathbf{q}_l}{\|\mathbf{q}_r\|_{\bar{F}_A} \|\mathbf{q}_l\|_{\bar{F}_A}}$, where $Z_{rl}^{F_A} \in [-1, 1]$ quantifies the task-aligned relation between known concepts $\mathbf{q}_r$ and $\mathbf{q}_l$. For comparison, we define task-agnostic Euclidean similarities $S_{ij}^{E} = \frac{\mathbf{q}_i^\top \mathbf{d}_j}{\|\mathbf{q}_i\|_2 \|\mathbf{d}_j\|_2}$, with $\mathbf{Z}^E$ defined analogously for supervised concepts. We formally define concept frustration and related quantities:

Definition 1 (Concept triplet frustration)**.** *Let $r \neq l$ index two known concepts and let j index an unsupervised direction. The triplet $(\mathbf{q}_r, \mathbf{q}_l, \mathbf{d}_j)$ exhibits concept frustration if* $\operatorname{sign}\big(Z_{rl}^{F_A}\big) \neq \operatorname{sign}\big(S_{rj}^{F_A} S_{lj}^{F_A}\big)$.

This definition mirrors frustration in spin systems [22], where spins are discrete random variables with covariances determined by pairwise couplings J_{ij}. A triangle of sites $\{r, l, j\}$ is frustrated when the three pairwise couplings J_{rl}, J_{lj}, J_{jr} between those sites cannot be satisfied simultaneously. This occurs when $\operatorname{sign}(J_{rl} J_{lj} J_{jr}) = -1$, which is equivalent to the definition of concept frustration.

Definition 2 (Maximally frustrating direction)**.** *For each known concept pair (r, l), define the maximally frustrating unsupervised direction $j^*(r, l)$ via* $j^*(r, l) = \arg\max_{j : \operatorname{sign}(Z_{rl}^{F_A}) \neq \operatorname{sign}(S_{rj}^{F_A} S_{lj}^{F_A})} \big|S_{rj}^{F_A} S_{lj}^{F_A}\big|$. *If no such j exists, the known concept pair is unfrustrated.*

Definition 3 (Pairwise frustration)**.** *We define the pairwise frustration for a known concept pair* (r, l) *as the magnitude of the maximally frustrating direction:*

$$\mathrm{Frust}^{F_A}(r,l) = \begin{cases} \left|S^{F_A}_{rj^*} S^{F_A}_{lj^*}\right| & \text{if } j^*(r,l) \text{ exists,} \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

Definition 4 (Global frustration)**.** *We define the global frustration as the average pairwise frustration across all known concept pairs:* $\gamma^{F_A} = \frac{2}{k_{\text{known}}(k_{\text{known}}-1)} \sum_{l<r} \mathrm{Frust}^{F_A}(r,l)$*. Thus* $\gamma^{F_A} \in [0,1]$ *measures the average magnitude of task-aligned geometric contradiction between supervised concepts mediated by unsupervised directions.*

We note that via Euclidean similarities $\mathbf{S}^E$ and $\mathbf{Z}^E$, Definitions 1-4 can be restated in Euclidean geometry yielding the Euclidean global frustration γ^E, for comparison.

Computing γ^{F_A} is inexpensive: forming $\bar{F}_A$ costs $O(|A_{\text{ave}}|\, r^2)$, the similarity matrices $\mathbf{S}^{F_A}$ and $\mathbf{Z}^{F_A}$ cost $O(r^2(k_{\text{SAE}} + k_{\text{known}}))$, and scanning all $\binom{k_{\text{known}}}{2} k_{\text{SAE}}$ candidate triplets for maximally frustrating directions costs $O(k^2_{\text{known}} k_{\text{SAE}})$, so the procedure is linear in the dictionary size.

3.3 Detecting concept frustration in the flatworld

As a motivating example, we return to our flatworlder thought experiment. Consider three concepts C_1, C_2, C_3 predictive of an outcome Y, where only (C_1, C_2) are supervised. Let C_1 and C_2 denote surface-parallel distance to the North and South poles respectively. For a traveler constrained to the Earth's surface, moving closer to the North pole necessarily moves them farther from the South pole, so the two distances behave as opposing concepts, i.e., $\mathrm{Cov}(C_1, C_2) < 0$.

We now allow the traveler to descend below the surface. In a spherical (round-world) geometry of radius 1, depth changes the surface geometry. Defining depth of a point as its inward distance from the surface, a traveler at depth d lies on a sphere of radius $1-d$, and the supervised concepts are defined as geodesic distances on that sphere to the North and South poles (Figure 2A). Increasing depth therefore shortens both geodesic distances simultaneously. We denote by C_3 depth below the surface and assume this concept is unsupervised. On a round Earth this induces $\mathrm{Cov}(C_1, C_3) < 0$, and $\mathrm{Cov}(C_2, C_3) < 0$. Together with $\mathrm{Cov}(C_1, C_2) < 0$, this sign pattern corresponds to the triplet-frustration structure defined in the previous section.

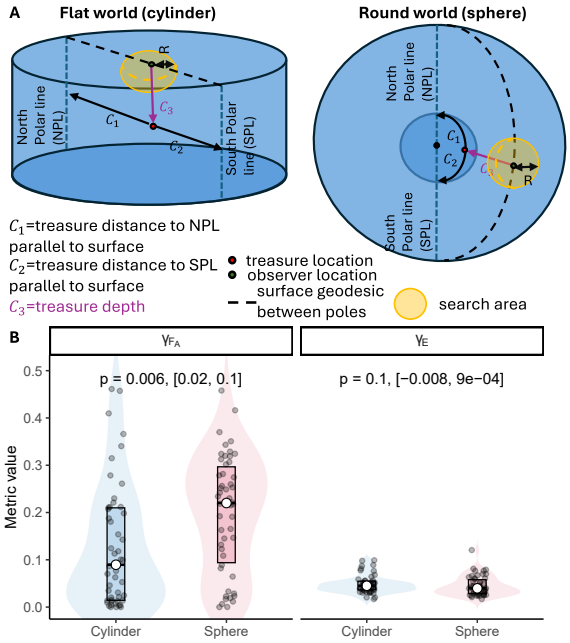

Figure 2: **Treasure hunter task and frustration.** (A) A signal encodes treasure location; the task is to detect whether it lies within a fixed radius. Known concepts C_1, C_2 capture distance to polar lines, while unknown C_3 represents depth. A flat Earth yields no frustration; a spherical Earth induces frustration. (B) Box/violin plots of γ_{F_A} and γ_E across 50 simulations. In-panel values show paired Wilcoxon p-values and 95% confidence intervals. γ_{F_A} detects frustration, while γ_E does not.

In contrast, under a cylindrical (flatworld) geometry, depth corresponds to translation orthogonal to a flat disk. The supervised concepts are defined as planar distances within the disk at fixed depth. In this case, depth does not alter surface geometry and is independent of the planar distances, so $\mathrm{Cov}(C_1, C_3) = 0$ and $\mathrm{Cov}(C_2, C_3) = 0$, thus no covariance-based concept frustration is expected. This motivates a simple "treasure hunter" task in which we determine whether a treasure is within a radius of our location from a signal encoding its position, using an interpretable model with known concepts (C_1, C_2). Formally, an activation $\mathbf{a}$ encodes a 3D location $\mathbf{p}$ and

the goal is to predict whether $\mathbf{p}$ lies within distance R of a fixed reference point $\mathbf{e} = (1, 0, 0)$: $Y(\mathbf{p}) = \mathbb{1}\{\|\mathbf{p} - \mathbf{e}\|_2 < R\}$.

We compare two geometries: cylindrical and spherical. In both cases we define the same supervised concepts abstractly as surface-parallel distances to the North and South poles, $C_1(\mathbf{p}) = \mathrm{dist}_{\mathrm{surface}}(\mathbf{p}, \mathrm{North})$, and $C_2(\mathbf{p}) = \mathrm{dist}_{\mathrm{surface}}(\mathbf{p}, \mathrm{South})$, with depth C_3 treated as unknown. We consider a linear-Gaussian mapping from concepts to activations to generate the signal $\mathbf{a}$. Details are provided in Appendix C.

Across 50 simulations per geometry, we trained the black-box model, SAE, and CBM and computed the frustration metrics defined above. Figure 2B shows that the task-aligned frustration score γ^{F_A} is significantly higher in the spherical setting than in the cylindrical (paired Wilcoxon $p = 5.4 \times 10^{-3}$, Cohen's $d_z = 0.43$, AUROC $= 0.68$), consistent with stronger task-aligned concept frustration in the round-world setting. In contrast, the Euclidean analogue γ^E does not distinguish the two settings (paired Wilcoxon $p = 0.118$, Cohen's $d_z = -0.21$, AUROC $= 0.42$). This indicates that frustration is not detectable from Euclidean alignment, but emerges when concepts are evaluated in a geometry encoding their contribution to task uncertainty. Further comparisons are provided in Table S1.

4 Results

4.1 A data model to control concept frustration

To study concept frustration under controlled conditions, we construct a synthetic data generator in which the covariance structure of supervised and unsupervised concepts can be manipulated explicitly while preserving the marginal structure of known concepts (see Appendix D for details).

We consider a concept vector $\mathbf{c} = (\mathbf{c}_k, \mathbf{c}_u) \in \mathbb{R}^k$, where $\mathbf{c}_k$ contains k_{known} supervised concepts and $\mathbf{c}_u$ contains k_{unknown} unsupervised concepts. Concepts are sampled from a zero-mean Gaussian distribution with covariance

$$\mathbf{B}(\alpha) = \begin{pmatrix} \mathbf{B}_{kk} & \mathbf{B}_{ku}(\alpha) \\ \mathbf{B}_{uk}(\alpha) & \mathbf{B}_{uu}(\alpha) \end{pmatrix},$$

where $\mathbf{B}_{kk}$ is the covariance among known concepts and $\mathbf{B}_{ku}(\alpha)$ the cross-covariance between known and unknown concepts. The parameter $\alpha \in [-1, 1]$ enforces three regimes. When $\alpha = 0$, known and unknown concepts are independent. For $\alpha < 0$, cross-block correlations reinforce the correlation structure within $\mathbf{B}_{kk}$, producing a non-frustrated regime. For $\alpha > 0$, they oppose it, generating concept frustration.

To ensure $\mathbf{B}(\alpha)$ is positive semi-definite, we take $\mathbf{B}_{uu}(\alpha)$ to be of the form $\mathbf{B}_{uu}(\alpha) = \mathbf{B}_{\mathrm{temp}} + \mathbf{B}_{uk}(\alpha)\mathbf{B}_{kk}^{-1}\mathbf{B}_{ku}(\alpha)$, with $\mathbf{B}_{\mathrm{temp}}$ an α-independent baseline covariance capturing variation intrinsic to the unsupervised concepts.

We define the task via $\tau = \boldsymbol{\psi}_k^\top \mathbf{c}_k + \boldsymbol{\psi}_u^\top \mathbf{c}_u + \eta$, where $\eta \sim \mathcal{N}(0, \sigma_y^2)$, and $\boldsymbol{\psi} = (\boldsymbol{\psi}_k, \boldsymbol{\psi}_u)$ specifies the task-relevant directions on known and unknown concepts, and σ_y^2 controls label noise. Binary labels are then generated as $y = \mathbb{1}\{\tau > 0\}$. A separate parameter ω controls how strongly the task depends on known versus unknown concepts through the construction of $\boldsymbol{\psi}$. Activations are generated from concepts through a linear-Gaussian map. This provides a controlled environment in which (i) the supervised concept ontology remains fixed across regimes, (ii) frustration strength is governed solely by α, and (iii) the task dependence on unknown structure is governed by ω.

4.2 Concept-optimal accuracy can be interpreted in terms of frustration

We now consider the concept-optimal Bayes classifier for the synthetic data defined in Section 4.1, that minimises the expected error of predicting the task, given knowledge of supervised concepts. We find that the accuracy of this classifier admits a closed form expression.

Theorem 2 (Concept-optimal Accuracy in the Linear–Gaussian Model)**.** *Consider the synthetic model defined in Section 4.1. Then the concept-optimal Bayes classifier using only the known concepts* $\mathbf{c}_k$*, which minimises the risk* $R(\hat{y}) = \mathbb{E}_{(\mathbf{c}_k, y)}\left[\mathbb{1}\{\hat{y}(\mathbf{c}_k) \neq y\}\right]$*, is* $\hat{y}(\mathbf{c}_k) = \mathbb{1}\{\boldsymbol{\phi}(\alpha)^\top \mathbf{c}_k > 0\}$*, and its classification accuracy is* $\mathrm{Acc}_{CBM}(\alpha) = \frac{1}{2} + \frac{1}{\pi}\arctan\left(\sqrt{\frac{\boldsymbol{\phi}(\alpha)^\top \mathbf{B}_{kk}\boldsymbol{\phi}(\alpha)}{\boldsymbol{\psi}_u^\top \mathbf{B}_{temp}\boldsymbol{\psi}_u + \sigma_y^2}}\right)$*, where* $\boldsymbol{\phi}(\alpha) = \boldsymbol{\psi}_k + \mathbf{B}_{kk}^{-1}\mathbf{B}_{ku}(\alpha)\boldsymbol{\psi}_u$,

See Appendix E for the proof. The denominator term $\boldsymbol{\psi}_u^\top \mathbf{B}_{temp} \boldsymbol{\psi}_u + \sigma_y^2$ is independent of α, thus $\text{Acc}_{\text{CBM}}(\alpha_1) > \text{Acc}_{\text{CBM}}(\alpha_2) \iff \boldsymbol{\phi}(\alpha_1)^\top \mathbf{B}_{kk} \boldsymbol{\phi}(\alpha_1) > \boldsymbol{\phi}(\alpha_2)^\top \mathbf{B}_{kk} \boldsymbol{\phi}(\alpha_2)$. Expanding this term yields

$$\begin{aligned}\boldsymbol{\phi}(\alpha)^\top \mathbf{B}_{kk} \boldsymbol{\phi}(\alpha) &= \langle \boldsymbol{\psi}_k, \mathbf{B}_{kk} \boldsymbol{\psi}_k \rangle_E + 2\langle \boldsymbol{\psi}_k, \mathbf{B}_{ku}(\alpha) \boldsymbol{\psi}_u \rangle_E + \langle \boldsymbol{\psi}_u, \mathbf{B}_{uk}(\alpha) \mathbf{B}_{kk}^{-1} \mathbf{B}_{ku}(\alpha) \boldsymbol{\psi}_u \rangle_E. \\ &= T_1 + 2T_2 + T_3\end{aligned}$$

This admits a geometric interpretation of optimal accuracy.

Supervised signal: $T_1 := \langle \boldsymbol{\psi}_k, \mathbf{B}_{kk} \boldsymbol{\psi}_k \rangle_E = ||\boldsymbol{\psi}_k||^2_{\mathbf{B}_{kk}}$. This measures task-relevant signal present in known concepts alone. Geometrically, it is the squared length of the supervised task direction $\boldsymbol{\psi}_k$ in the covariance geometry induced by $\mathbf{B}_{kk}$. This term is invariant under frustration.

Cross-alignment between supervised and unknown signal: $T_2(\alpha) := \langle \boldsymbol{\psi}_k, \mathbf{B}_{ku}(\alpha) \boldsymbol{\psi}_u \rangle_E$. This term captures how the unknown task direction $\boldsymbol{\psi}_u$ projects into known-concept space through the cross-covariance matrix $\mathbf{B}_{ku}(\alpha)$, and whether this projection reinforces or interferes with the supervised task direction $\boldsymbol{\psi}_k$. Positive alignment increases accuracy, whereas negative alignment reduces it. Frustration acts primarily by modifying this term.

Recoverable unknown signal: $T_3(\alpha) := \langle \boldsymbol{\psi}_u, \mathbf{B}_{uk}(\alpha) \mathbf{B}_{kk}^{-1} \mathbf{B}_{ku}(\alpha) \boldsymbol{\psi}_u \rangle_E = ||\boldsymbol{\psi}_u||^2_{\mathbf{B}_{uk}(\alpha) \mathbf{B}_{kk}^{-1} \mathbf{B}_{ku}(\alpha)}$. This term quantifies the magnitude of unknown task-relevant variation that becomes linearly recoverable from known concepts via cross-block correlations. It measures signal strength rather than directional alignment, and is zero if known and unknown concepts are independent ($\alpha = 0$).

The term $\boldsymbol{\psi}_u^\top \mathbf{B}_{temp} \boldsymbol{\psi}_u + \sigma_y^2 = T_4 + \sigma_y^2$ also admits a geometric interpretation: T_4 is the unknown signal not recoverable from supervised concepts, and the larger this term is the worse concept-optimal accuracy. Informally, T_1 describes known-knowns, T_2 and T_3 known-unknowns and T_4 unknown-unknowns.

4.3 Frustration between known and unknown concepts limits CBM task accuracy

We trained CBMs, black-box models, and SAEs across 60 parameter regimes of our synthetic data generator (see Appendix F.1). Each regime was repeated across 10 independent seeds, yielding 600 matched datasets. Full results are provided in Table S2.

Frustration significantly reduced CBM task accuracy. CBM accuracy was significantly lower under $\alpha = +1$ compared to both $\alpha = -1$ and $\alpha = 0$ (paired Wilcoxon $p < 10^{-4}$; Figure 3A). In contrast, black-box model accuracy did not differ between $\alpha = +1$ and $\alpha = -1$ (paired Wilcoxon $p = 0.13$, Figure 3A), and was slightly worse when $\alpha = 0$, indicating that frustration limits CBM performance without increasing intrinsic task difficulty.

4.4 Frustration impairs semantic interpretability of CBMs

We next examined the effect of frustration on concept-level interpretability. Frustration significantly impaired concept prediction accuracy. Concept mean

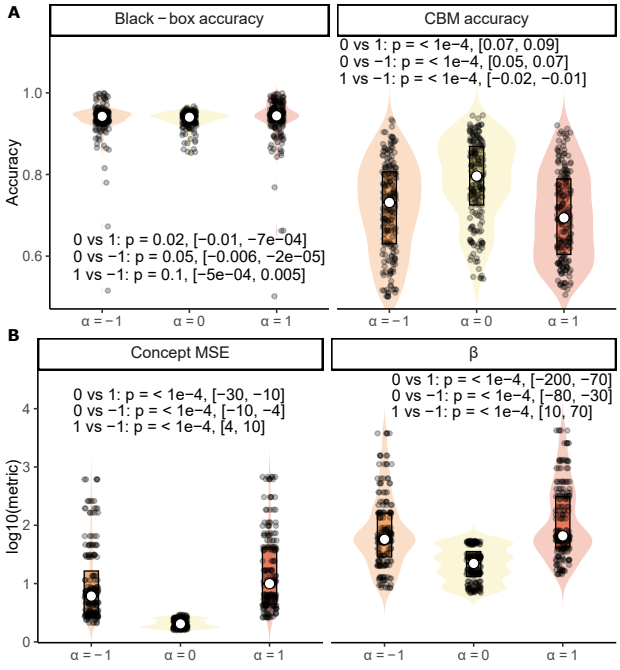

Figure 3: **Frustration degrades CBM accuracy and interpretability.** Box/violin plots show (A) black-box accuracy (left), CBM accuracy (right), (B) concept MSE (left), and semantic fidelity β (right) for $\alpha \in \{-1, 0, 1\}$ across 600 simulations. In-panel values show paired Wilcoxon p-values and 95% confidence intervals. Frustration reduces CBM performance and interpretability without affecting black-box accuracy.

squared error (MSE) was significantly higher under $\alpha = +1$ compared to both $\alpha = -1$ and $\alpha = 0$ (paired Wilcoxon $p < 10^{-4}$; Figure 3B). When trained on the synthetic data, a CBM learns concepts $\hat{\mathbf{c}}_{\text{known}}$ with covariance $\text{Cov}(\hat{\mathbf{c}}_{\text{known}})$ that may differ from the ground-truth covariance structure $\mathbf{B}_{\text{known}}$. To quantify this decrease in interpretability, we define the following score.

Definition 5 (Semantic fidelity)**.** *The semantic fidelity score β measures deviation between the covariance of predicted known concepts and the ground-truth covariance structure* $\beta = \frac{\|\text{Cov}(\hat{\mathbf{c}}_{\text{known}}) - \mathbf{B}_{\text{known}}\|_{\text{Frob}}}{\|\mathbf{B}_{\text{known}}\|_{\text{Frob}}}$.

Under frustration, β increased markedly (paired Wilcoxon $p < 10^{-4}$, Figure 3B), indicating distortion of the learned inter-concept relationships. Thus, frustration degrades both predictive and structural interpretability of CBMs.

4.5 A task-aligned geometry detects known-unknown concept frustration

We next evaluated whether frustration can be detected via our metric. The task-aligned frustration metric γ^{F_A} was significantly higher under $\alpha = +1$ compared to both $\alpha = -1$ and $\alpha = 0$ (paired Wilcoxon $p < 10^{-4}$; $\alpha = -1$: Cohen's $d_z = 0.49$, AUROC $= 0.61$; $\alpha = 0$: Cohen's $d_z = 0.80$, AUROC $= 0.80$, Figure 4A). In contrast, the Euclidean analogue γ^{E} did not reliably distinguish frustration (Figure 4A). Moreover, γ^{F_A} was elevated when the theoretical cross alignment term T_2 was non-zero (paired Wilcoxon $p < 10^{-4}$), whereas γ^{E} was not (Figure 4B). Thus, frustration is detectable in task-aligned Fisher geometry but not in Euclidean geometry. To assess the robustness of maximally frustrating SAE atom selection, we varied dictionary size, sparsity regularisation and random initialisation. The mean task-aligned frustration metric gap between frustrated and consistent regimes remained stable across these settings (Appendix F.1).

4.6 Detecting concept frustration and localising omitted concepts in real-world tasks

We next evaluated whether concept frustration can be detected in real-world language and vision tasks. For language we considered sarcasm detection. We used a dataset of 28,000 news headlines labelled with sarcasm [23] and embedded them using the pretrained DeBERTa-v3-base language model [24] (Appendix F.2). From a pretrained RoBERTa sentiment classifier [25] we derived three sentiment-related concepts: C_1 (positive sentiment), C_2 (negative sentiment), and C_3 (sentiment intensity). In this dataset, C_1 and C_2 are inversely correlated, while C_3 is positively correlated with both, forming a frustrating triple. A linear model confirmed all three concepts independently associated with sarcasm. We trained two CBMs on the same embedding space. CBM1 used (C_1, C_2) as supervised concepts, treating C_3 as unknown. CBM2 used all three concepts (C_1, C_2, C_3) as supervised. Both models were compared against the same SAE dictionary $\mathbf{D}$ learned from the embedding space. Frustration was computed only on the concept pair (C_1, C_2) for both models as $\text{Frust}^{F_A/E}(1, 2)$.

$\text{Frust}^{F_A}(1, 2)$ was significantly higher in CBM1 than in CBM2 (paired Wilcoxon $p = 0.0488$, Cohen's $d_z = 0.45$, AUROC

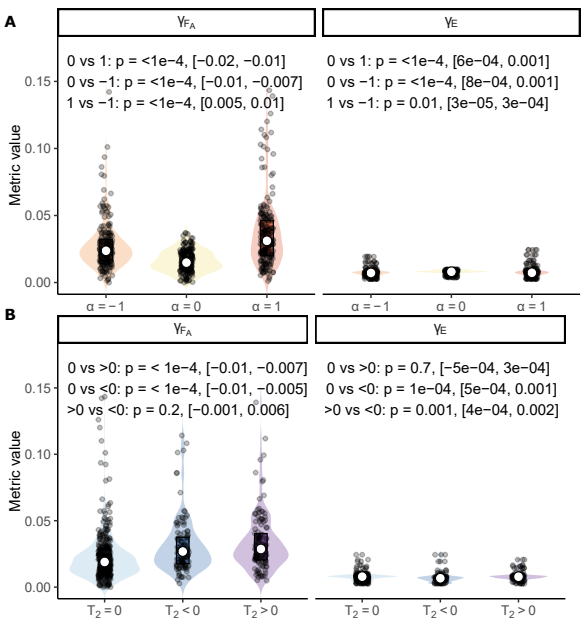

Figure 4: γ_{F_A} **detects concept frustration.** Box/violin plots show γ_{F_A} (left) and γ_E (right) across 600 simulations and 60 parameter regimes, grouped by (A) α and (B) sign of T_2. In-panel values show Wilcoxon p-values and 95% confidence intervals. γ_{F_A} increases under frustrated known–unknown concept covariance and non-zero cross-alignment T_2, whereas γ_E does not.

$= 0.80$; Figure 5A). In contrast, the Euclidean analogue $\mathrm{Frust}^{E}(1,2)$ did not differ between models. Further results are provided in Table S3.

For vision, we extended this analysis to 845 naturally occurring frustrated concept triples across three binary classification tasks in CUB-200-2011 [26], with images embedded using a pretrained CLIP vision encoder [27]. Triples were selected based on two criteria: (i) pairwise concept correlations satisfied $\mathrm{sign}(\mathrm{cor}(C_1, C_2)) \neq \mathrm{sign}(\mathrm{cor}(C_1, C_3)\mathrm{cor}(C_2, C_3))$; and (ii) all three concepts were independently associated with the downstream task. Across these triples, $\mathrm{Frust}^{F_A}(1,2)$ was higher in CBM1 than in CBM2 (Cohen's $d_z = 0.75$, AUROC $= 0.77$; Figure 5B). The Euclidean analogue $\mathrm{Frust}^{E}(1,2)$ showed the same trend but substantially weaker effect size and ability to distinguish the two models (Cohen's $d_z = 0.27$, AUROC $= 0.56$). Selection criteria, task-specific results and experimental details are provided in Appendix F.3.

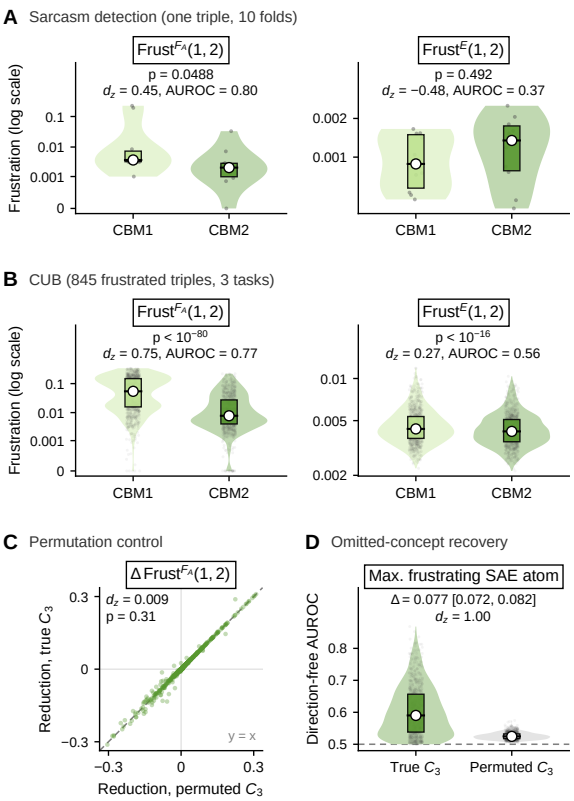

Figure 5: **Frustration in real-world tasks.** (A,B) Box/violin plots of $\mathrm{Frust}^{F_A}(1,2)$ and $\mathrm{Frust}^{E}(1,2)$ (log scale) for CBM1 and CBM2 on (A) sarcasm detection across 10-fold cross-validation and (B) 845 frustrated CUB concept triples pooled across three tasks. In-panel values show paired Wilcoxon p-values, Cohen's d_z and AUROC. (C) Reduction in $\mathrm{Frust}^{F_A}(1,2)$ (CBM1 − CBM2) after adding the true C_3 versus label-permuted C_3 (mean of five permutations); dashed line, $y = x$. (D) Direction-free AUROC of the maximally frustrating SAE atom from CBM1 for the true versus permuted C_3 labels; in-panel values show the paired difference (95% CI) and d_z; dashed line, chance.

To assess whether this reduction was specific to the omitted concept, we compared adding the true frustrating concept C_3 with a label-permuted control that preserved its prevalence while disrupting its relationship with the inputs and task. Adding the true concept yielded no appreciable additional reduction in $\mathrm{Frust}^{F_A}(1,2)$ compared with the permutation control (paired $d_z = 0.009$; Figure 5C). Nevertheless, the SAE direction identified by CBM1 as maximally frustrating recovered the true omitted concept better than the permuted version (mean direction-free AUROC 0.603 versus 0.526; paired difference 0.077, 95% CI $[0.072, 0.082]$; Figure 5D). Experimental details are provided in Appendix F.4

These results distinguish reductions in scalar frustration from the identification of omitted concepts. Expanding the supervised ontology can reduce Fisher frustration by altering the geometry of learned concept representations, without necessarily resolving the underlying representational inconsistency. In contrast, the maximally frustrating SAE direction captures more information about the true omitted concept than a matched permutation control, supporting its use to identify candidate missing semantic structure for subsequent interpretation and ontology refinement.

5 Discussion

Alignment between human-interpretable reasoning and machine representations remains a central challenge in explainable AI [28, 29]. A substantial body of work assumes that unsupervised machine representations are only valuable when orthogonal to human-defined concepts [30, 31, 20, 32]. Our results suggest a different perspective: machine-derived concepts can instead induce relationships between known concepts that contradict their expected semantics, motivating a reorganisation of the geometry of the known concept space. We term such concepts *frustrating*. Their presence can

compromise both the accuracy and interpretability of supervised concept models. Our framework has two complementary roles: scalar Fisher frustration quantifies task-aligned representational inconsistency and separates frustrated from consistent regimes in controlled experiments, while maximally frustrating SAE directions localise candidate omitted concepts better than matched permutation controls. However, scalar reductions in frustration after ontology expansion do not establish that the added concept specifically resolves the underlying inconsistency.

This process resembles how conceptual change occurs in scientific discovery [33], where progress often arises not from introducing independent variables but from discovering concepts that force a reinterpretation of existing knowledge. Recognising that the Earth is spherical rather than flat resolves apparent contradictions in geodesic distances observed when moving in depth. Similarly, Einstein's theory of relativity reorganised the classical understanding of space, time and motion, resolving tensions that could not be explained within Newtonian mechanics. This analogy motivates using concept frustration to identify candidate concepts for ontology refinement, while emphasising the need for semantic interpretation and validation to establish whether a proposed concept resolves contradictions in model reasoning.

Our theory further reveals that the predictive consequences of frustration depend on the interplay between concept covariance and concept–task geometry. Dependence between known and unknown concepts can increase recoverable task signal through T_3, while its alignment with the known task direction, captured by T_2, can either enhance or impair prediction. Thus, frustration does not universally reduce accuracy; its effect depends on how these geometric contributions interact. Extending this characterisation beyond the linear–Gaussian setting remains an important direction for future work.

6 Limitations and future work

Though encouraging, our results are subject to some limitations. First, our theoretical analysis relies on a linear–Gaussian generative model which enables analytical tractability. This is not an unrealistic assumption: CBMs typically employ a linear mapping from concepts to outcomes to preserve interpretability [13], and foundation models are often assumed to provide embeddings from which relevant concepts are approximately linearly recoverable [34–36]. However, real-world representation spaces may exhibit more complex associations. Second, our measure of pairwise frustration depends on selecting a maximally frustrating unsupervised direction from the SAE dictionary. While this choice captures the strongest geometric contradiction between concept directions, alternative strategies may provide complementary insights. Third, the task-aligned geometry is derived from a quadratic form based on the Fisher information metric of a simple black-box classifier. While this construction provides an analytically tractable geometry, more complex predictive models may induce richer structures.

Although the selected SAE directions recover omitted concepts better than permutation controls, their recovery performance remains modest (mean direction-free AUROC 0.603). Improving concept localisation and establishing the semantic meaning of selected directions are important priorities. Our experiments also relied on frozen pre-concept activations to isolate frustration from representation learning. Extending the framework to end-to-end CBMs would involve defining concept directions and a task-aligned Fisher geometry in the pre-concept activation space. Since end-to-end training allows this space to adapt to concept supervision, it may alter the geometry and magnitude of frustration. A systematic comparison with frozen-backbone models remains future work.

Further promising directions of future work include extending the theory beyond linear–Gaussian settings, analysing higher-order structures in concept similarity matrices, elaborating the connection to frustration in spin systems, and applying frustration-based diagnostics to large foundation models. Our perspective also motivates new model architectures that combine supervised concepts with unsupervised concept discovery. More broadly, viewing interpretability through the lens of geometric consistency between human concepts and machine representations provides a principled framework for identifying missing conceptual dimensions and refining ontologies through which both humans and machines reason about complex systems.

Acknowledgments and Disclosure of Funding

EP is funded by the Department of Science Innovation and Technology under PharosAI. CS is funded by the MANIFEST programme, core funded by the UK Government's Office for Life Sciences and the Medical Research Council (Grant Ref: MR/Z505158/1). AI is funded by the Martingale Foundation. CRSB is supported by a King's College London AI+ Fellowship. The authors declare no competing interests.

References

[1] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 1(5):206–215, 2019.

[2] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2017.

[3] Vikas Hassija, Vinay Chamola, Atmesh Mahapatra, Abhinandan Singal, Divyansh Goel, Kaizhu Huang, Simone Scardapane, Indro Spinelli, Mufti Mahmud, and Amir Hussain. Interpreting black-box models: a review on explainable artificial intelligence. *Cognitive Computation*, 16 (1):45–74, 2024.

[4] EU Member States. The EU AI Act, 2026. URL `https://artificialintelligenceact.eu/`.

[5] Reuben Binns and Michael Veale. Is that your final decision? Multi-stage profiling, selective effects, and Article 22 of the GDPR. *International Data Privacy Law*, 11:319–332, 12 2021. ISSN 20444001. doi: 10.1093/idpl/ipab020. URL `https://dx.doi.org/10.1093/idpl/ipab020`.

[6] Scott Lundberg and Su-In Lee. A Unified Approach to Interpreting Model Predictions, November 2017. URL `http://arxiv.org/abs/1705.07874`. arXiv:1705.07874 [cs].

[7] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why Should I Trust You?": Explaining the Predictions of Any Classifier, August 2016. URL `http://arxiv.org/abs/1602.04938`. arXiv:1602.04938 [cs].

[8] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic Attribution for Deep Networks, June 2017. URL `http://arxiv.org/abs/1703.01365`. arXiv:1703.01365 [cs].

[9] A. Saranya and R. Subhashini. A systematic review of explainable artificial intelligence models and applications: Recent developments and future trends. *Decision Analytics Journal*, 7:100230, 6 2023. ISSN 2772-6622. doi: 10.1016/J.DAJOUR.2023.100230.

[10] Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. *12th International Conference on Learning Representations, ICLR 2024*, 10 2023. URL `http://arxiv.org/abs/2309.08600`.

[11] Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. Toy models of superposition. 9 2022. URL `http://arxiv.org/abs/2209.10652`.

[12] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6541–6549, 2017.

[13] Pang Wei Koh, Thao Nguye, Yew Siang Tang, Stephen Mussmann, Emma Pierso, Been Kim, and Percy Liang. Concept bottleneck models. *37th International Conference on Machine Learning, ICML 2020*, PartF168147-7:5294–5304, 7 2020. URL `https://arxiv.org/abs/2007.04612v3`.

[14] Mateo Espinosa Zarlenga, Pietro Barbiero, Gabriele Ciravegna, Giuseppe Marra, Francesco Giannini, Michelangelo Diligenti, Zohreh Shams, Frederic Precioso, Stefano Melacci, Adrian Weller, Pietro Lio, and Mateja Jamnik. Concept embedding models: Beyond the accuracy-explainability trade-off. *Advances in Neural Information Processing Systems*, 35, 9 2022. ISSN 10495258. URL https://arxiv.org/abs/2209.09056v2.

[15] David Debot and Giuseppe Marra. Quantifying the accuracy-interpretability trade-off in concept-based sidechannel models. 10 2025. URL http://arxiv.org/abs/2510.05670.

[16] Anita Mahinpei, Justin Clark, Isaac Lage, Finale Doshi-Velez, and Weiwei Pan. Promises and pitfalls of black-box concept learning models. *CoRR*, abs/2106.13314, 2021. URL https://api.semanticscholar.org/CorpusID:235652059.

[17] Marton Havasi, Sonali Parbhoo, and Finale Doshi-Velez. Addressing leakage in concept bottleneck models, 10 2022.

[18] Yoshihide Sawada and Keigo Nakamura. Concept bottleneck model with additional unsupervised concepts. *IEEE Access*, 10:41758–41765, 2022. URL https://api.semanticscholar.org/CorpusID:246485723.

[19] Mert Yuksekgonul, Maggie Wang, and James Zou. Post-hoc concept bottleneck models. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=nA5AZ8CEyow.

[20] Aya Abdelsalam Ismail, Julius Adebayo, Hector Corrada Bravo, Stephen Ra, and Kyunghyun Cho. Concept bottleneck generative models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=L9U5MJJleF.

[21] Alexandre Rocchi-Henry, Thomas Fel, and Gianni Franchi. A geometric unification of concept learning with concept cones. 12 2025. URL http://arxiv.org/abs/2512.07355.

[22] M Mezard, G Parisi, and M Virasoro. *Spin Glass Theory and Beyond*. WORLD SCIENTIFIC, 1986. doi: 10.1142/0271. URL https://www.worldscientific.com/doi/abs/10.1142/0271.

[23] Rishabh Misra and Prahal Arora. Sarcasm detection using news headlines dataset. *AI Open*, 4:13–18, 2023. ISSN 2666-6510. doi: https://doi.org/10.1016/j.aiopen.2023.01.001. URL https://www.sciencedirect.com/science/article/pii/S2666651023000013.

[24] Pengcheng He, Jianfeng Gao, and Weizhu Chen. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing, 2021.

[25] Francesco Barbieri, Jose Camacho-Collados, Luis Espinosa Anke, and Leonardo Neves. TweetEval: Unified benchmark and comparative evaluation for tweet classification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1644–1650, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.148. URL https://aclanthology.org/2020.findings-emnlp.148.

[26] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[27] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. *Proceedings of Machine Learning Research*, 139:8748–8763, 2 2021. URL http://arxiv.org/abs/2103.00020.

[28] Christopher R. S. Banerji, Tapabrata Chakraborti, Aya Abdelsalam Ismail, Florian Ostmann, and Ben D. MacArthur. Train clinical AI to reason like a team of doctors. *Nature 2025 639:8053*, 639:32–34, 3 2025. ISSN 1476-4687. doi: 10.1038/d41586-025-00618-x. URL https://www.nature.com/articles/d41586-025-00618-x.

[29] Emanuele Marconato, Andrea Passerini, and Stefano Teso. Interpretability is in the mind of the beholder: A causal framework for human-interpretable representation learning. *Entropy*, 25(12), 2023. ISSN 1099-4300. doi: 10.3390/e25121574. URL https://www.mdpi.com/1099-4300/25/12/1574.

[30] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=Sy2fzU9gl.

[31] Zhi Chen, Yijie Bei, and Cynthia Rudin. Concept whitening for interpretable image recognition. *Nature Machine Intelligence*, 2(12):772–782, December 2020. ISSN 2522-5839. doi: 10.1038/s42256-020-00265-z. URL https://doi.org/10.1038/s42256-020-00265-z.

[32] Xiyu Meng, Yilong Lin, Yuhan Wu, and Lu Ying. Enhancing concept alignment with explanatory interactive disentangled representation learning. *Neural Networks*, 196:108346, 2026. ISSN 0893-6080. doi: https://doi.org/10.1016/j.neunet.2025.108346. URL https://www.sciencedirect.com/science/article/pii/S0893608025012274.

[33] Thomas S. Kuhn. *The Structure of Scientific Revolutions*. University of Chicago Press, 50th edition, 2012.

[34] Jihye Choi, Jayaram Raghuram, Yixuan Li, Suman Banerjee, and Somesh Jha. Adaptive concept bottleneck for foundation models. In *ICML 2024 Workshop on Foundation Models in the Wild*, 2024. URL https://openreview.net/forum?id=gVswBbGg2C.

[35] Usha Bhalla, Alex Oesterling, Suraj Srinivas, Flavio Calmon, and Himabindu Lakkaraju. Interpreting CLIP with sparse linear concept embeddings (spliCE). In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=7UyBKTFrtd.

[36] Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 39643–39666. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/park24c.html.

[37] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. URL https://arxiv.org/abs/1412.6980.

A Model architectures

Black box model. We employ a feedforward architecture with a single hidden layer of width h and ReLU activation,

$$\begin{aligned} H(\mathbf{a}) &= \mathrm{ReLU}(\mathbf{W}_H \mathbf{a} + b_H) \\ l(\mathbf{a}) &= \mathbf{w}_l^\top H(\mathbf{a}) + b_l, \\ \hat{y} &= \sigma(l(\mathbf{a})) \end{aligned}$$

where $H : \mathbb{R}^r \to \mathbb{R}^h$, $l : \mathbb{R}^h \to \mathbb{R}$ and σ is the sigmoid function. $\mathbf{W}_H \in \mathbb{R}^{h \times r}$ is a learned matrix of weights and $\mathbf{w}_l \in \mathbb{R}^h$ is a learned vector of weights. $b_H \in \mathbb{R}^h$ is a learned vector of biases and $b_l \in \mathbb{R}$ is a learned bias.

For binary tasks, parameters are learned by minimizing the empirical binary cross-entropy loss

$$\mathcal{L}_{\text{BB}} = \mathbb{E}\big[\ell_{\text{BCE}}(\hat{y}(\mathbf{a}), y)\big].$$

Sparse Autoencoder. SAEs are trained by minimizing

$$\mathcal{L}_{\text{SAE}} = \mathbb{E}\big[\|\mathbf{a} - \hat{\mathbf{a}}\|_2^2\big] + \lambda_{\text{SAE}}\, \mathbb{E}\big[\|s(\mathbf{a})\|_1\big],$$

where $\lambda_{\text{SAE}} > 0$ controls sparsity.

Concept Bottleneck Model. We parameterise CBMs with a linear concept mapping

$$\hat{\mathbf{c}}_{\text{known}}(\mathbf{a}) = \mathbf{Q}\mathbf{a},$$

with $\mathbf{Q} \in \mathbb{R}^{k_{\text{known}} \times r}$. The predicted concepts are then passed to a linear task head f, and the final binary prediction $\hat{y}(\mathbf{a})$ is obtained via a sigmoid transformation:

$$\hat{y}(\mathbf{a}) = \sigma(\mathbf{w}_{\text{CBM}}^{\top} \hat{\mathbf{c}}_{\text{known}}(\mathbf{a}) + b_{\text{CBM}}),$$

We consider CBM joint training, where concept and task losses are optimized simultaneously:

$$\mathcal{L}_{\text{joint}} = \mathcal{L}_{\text{task}}(y, \hat{y}) + \lambda_C \, \mathcal{L}_{\text{concept}}(\mathbf{c}_{\text{known}}, \hat{\mathbf{c}}_{\text{known}}),$$

where

$$\mathcal{L}_{\text{concept}} = \frac{1}{n} \sum_{i=1}^{n} \left\| \mathbf{c}_{\text{known}}^{(i)} - \hat{\mathbf{c}}_{\text{known}}^{(i)} \right\|_2^2,$$

and $\mathcal{L}_{\text{task}}$ is binary cross-entropy, and λ_C balances concept prediction and task prediction. Both the concept predictor and task head are updated jointly.

B Proof of Theorem 1

Here we prove the following theorem:

Theorem (Closed-form Fisher metric for one-hidden-layer binary model)**.** *Consider the black-box model defined above with sigmoid output. Then the Fisher information metric on activation space admits the closed-form expression*

$$F_A(\mathbf{a}) = p(\mathbf{a})(1 - p(\mathbf{a}))\, g(\mathbf{a}) g(\mathbf{a})^{\top},$$

where $p(\mathbf{a}) \equiv p(y = 1|\mathbf{a}) = \sigma(l(\mathbf{a}))$ *and*

$$g(\mathbf{a}) = \nabla_a l(\mathbf{a}).$$

Proof. Consider how the black box model is trained, we have an input $\mathbf{a} \in \mathbb{R}^r$ which is transformed by a hidden layer according to a function $H : \mathbb{R}^r \to \mathbb{R}^h$, defined:

$$H(\mathbf{a}) = \text{ReLU}(\mathbf{W}_H \mathbf{a} + b_H), \tag{2}$$

where $\mathbf{W}_H \in \mathbb{R}^{h \times r}$ is a learned matrix of weights and $b_H \in \mathbb{R}^h$ is a learned vector of biases.

This layer is then transformed to a logit via

$$l(\mathbf{a}) = \mathbf{w}_l^{\top} H(\mathbf{a}) + b_l, \tag{3}$$

where $\mathbf{w}_l \in \mathbb{R}^h$ is a learned vector of weights and $b_l \in \mathbb{R}$ is a learned bias.

This is then converted into a probability via a sigmoid:

$$p(y = 1|\mathbf{a}) = \sigma(l(\mathbf{a})) \tag{4}$$
$$p(y = 0|\mathbf{a}) = 1 - \sigma(l(\mathbf{a})). \tag{5}$$

We want to calculate:

$$F_A(\mathbf{a}) = \mathbb{E}_{y \sim p(y|\mathbf{a})} \left[\nabla_{\mathbf{a}} \log p(y|\mathbf{a}) \nabla_{\mathbf{a}} \log p(y|\mathbf{a})^{\top} \right] \tag{6}$$

Since the model defines a Bernoulli distribution over $y \in \{0, 1\}$, we have:

$$\log p(y|\mathbf{a}) = y \log \sigma(l(\mathbf{a})) + (1 - y) \log(1 - \sigma(l(\mathbf{a}))), \tag{7}$$

hence, using the identity for sigmoid functions $\sigma'(l(\mathbf{a})) = \sigma(l(\mathbf{a}))(1 - \sigma(l(\mathbf{a})))$

$$\nabla_{\mathbf{a}} \log p(y|\mathbf{a}) = (y - \sigma(l(\mathbf{a})))\nabla_{\mathbf{a}} l(\mathbf{a}). \tag{8}$$

Taking expectations over y:

$$F_A(\mathbf{a}) = \mathbb{E}[((y - \sigma(l(\mathbf{a})))^2]\nabla_{\mathbf{a}} l(\mathbf{a}) \nabla_{\mathbf{a}} l(\mathbf{a})^\top \tag{9}$$

$$= \sigma(l(\mathbf{a}))(1 - \sigma(l(\mathbf{a})))\nabla_{\mathbf{a}} l(\mathbf{a}) \nabla_{\mathbf{a}} l(\mathbf{a})^\top \tag{10}$$

which follows as $\mathbb{E}[((y - \sigma(l(\mathbf{a})))^2] = \sigma(l(\mathbf{a}))(1 - \sigma(l(\mathbf{a})))$ is the variance of the Bernoulli distribution $\text{Ber}(\sigma(l(\mathbf{a})))$.

Recall

$$l(\mathbf{a}) = \mathbf{w}_l^\top \text{ReLU}(\mathbf{W}_H \mathbf{a} + b_H) + b_l, \tag{11}$$

noting that $\frac{\partial \text{ReLU}(\mathbf{z})}{\partial z_j} = \mathbb{1}\{z_j > 0\}$, and defining

$$m_i(\mathbf{a}) = \mathbb{1}\{(\mathbf{W}_H \mathbf{a} + b_H)_i > 0\}, \tag{12}$$

we deduce

$$\frac{\partial l(\mathbf{a})}{\partial a_i} = \sum_{j=1}^{h} (\mathbf{w}_l)_j m_j(\mathbf{a}) (\mathbf{W}_H)_{ji} \tag{13}$$

or

$$\nabla_{\mathbf{a}} l(\mathbf{a}) = \mathbf{W}_H^\top (m(\mathbf{a}) \odot \mathbf{w}_l), \tag{14}$$

where $(\mathbf{a} \odot \mathbf{b})_j = a_j b_j$ is the Hadamard product.

Thus,

$$F_A(\mathbf{a}) = p(\mathbf{a})(1 - p(\mathbf{a}))g(\mathbf{a})g(\mathbf{a})^\top \tag{15}$$

where $p(\mathbf{a}) = \sigma(l(\mathbf{a}))$ and $g(\mathbf{a}) = \mathbf{W}_H^\top (m(\mathbf{a}) \odot \mathbf{w}_l)$. □

C Globe treasure hunter data generation and training protocol

Geometric settings. We consider two geometric settings: a cylindrical ("flatworld") geometry and a spherical ("round-world") geometry. In both settings, points $\mathbf{p} \in \mathbb{R}^3$ represent possible treasure locations.

Spherical geometry. Let $\mathbb{S}^2 = \{\mathbf{u} \in \mathbb{R}^3 : \|\mathbf{u}\|_2 = 1\}$ denote the unit sphere. To generate a point:

1. Sample a surface direction $\mathbf{u} \sim \text{Unif}(\mathbb{S}^2)$.
2. Sample a radial depth $d \sim \text{Unif}([0, 1))$.
3. Define the final location as
$$\mathbf{p} = (1 - d)\mathbf{u}.$$

Thus $d = 0$ corresponds to the surface of the unit sphere, and increasing d moves radially inward to a sphere of radius $1 - d$.

Cylindrical geometry. Let $\mathbb{D}^2 = \{(x, y) \in \mathbb{R}^2 : x^2 + y^2 \leq 1\}$ denote the unit disk. To generate a point:

1. Sample a surface location $(x_u, y_u) \sim \text{Unif}(\mathbb{D}^2)$.
2. Sample a vertical depth $d \sim \text{Unif}([0, 1))$.
3. Define
$$\mathbf{p} = (x_u, y_u, -d).$$

In this setting, the Earth's surface corresponds to the plane $z = 0$, and increasing d translates orthogonally away from the plane without altering its intrinsic disk geometry. The North and South poles correspond to the loci $y = 1$ and $y = -1$ respectively.

Concept definitions. In both geometries we define three concepts. Let $\rho = 1 - d$ denote the radius of the spherical cross-section at depth d.

Supervised concepts (C_1, C_2).

We define C_1 and C_2 as surface-parallel distances to the North and South poles.

- In the spherical geometry, these are geodesic distances on the sphere of radius $\rho = 1 - d$:

$$C_1(\mathbf{p}) = \rho \cdot \arccos\left(\frac{\mathbf{p} \cdot (0, \rho, 0)}{\rho^2}\right), \tag{16}$$

$$C_2(\mathbf{p}) = \rho \cdot \arccos\left(\frac{\mathbf{p} \cdot (0, -\rho, 0)}{\rho^2}\right). \tag{17}$$

  Since $\mathbf{p} = \rho\mathbf{u}$ with $\mathbf{u} \in \mathbb{S}^2$, these simplify to

$$C_1(\mathbf{p}) = \rho \arccos(u_y), \qquad C_2(\mathbf{p}) = \rho \arccos(-u_y).$$

- In the cylindrical geometry, these are intrinsic (planar) distances within the disk at fixed depth:

$$C_1(\mathbf{p}) = \sqrt{x^2 + (y-1)^2}, \tag{18}$$

$$C_2(\mathbf{p}) = \sqrt{x^2 + (y+1)^2}. \tag{19}$$

Unsupervised concept.

$$C_3(\mathbf{p}) = d \quad \text{(depth)}. \tag{20}$$

Only (C_1, C_2) are treated as supervised concepts. Depth C_3 is unobserved by the CBM.

Concept-to-activation mapping. For each location $\mathbf{p}$, we define the concept vector

$$\mathbf{c}^{(\mathbf{p})} = (C_1(\mathbf{p}), C_2(\mathbf{p}), C_3(\mathbf{p}))^\top \in \mathbb{R}^3.$$

Activations are generated via a linear–Gaussian model:

$$\mathbf{a}^{(\mathbf{p})} = \mathbf{\Phi}\mathbf{c}^{(\mathbf{p})} + \boldsymbol{\epsilon}, \tag{21}$$

where:

- $\mathbf{\Phi} \in \mathbb{R}^{r \times 3}$ has entries independently sampled from $\mathcal{N}(0, 1)$,
- $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma_a^2 \mathbf{I}_r)$,
- $r = 64$ denotes the activation dimension.

Thus the activation space is a noisy linear embedding of the three underlying concepts.

Treasure hunter task. We fix a reference point on the equator,

$$\mathbf{e} = (1, 0, 0),$$

and define the binary classification task

$$Y(\mathbf{p}) = \mathbb{1}\{\|\mathbf{p} - \mathbf{e}\|_2 < R\}, \tag{22}$$

with $R = 0.75$.

Dataset construction. For each geometry, we generate $n = 8000$ samples. Data are split into training and test sets in a 75:25 ratio. All reported results are averaged over 50 independent repetitions, each with independently sampled $\mathbf{\Phi}$ and random train/test splits.

Model training. We train:

- A one-hidden-layer black-box model with width $h = 128$ using binary cross-entropy loss.
- A sparse autoencoder with $k_{\text{SAE}} = 60$ latent units and sparsity coefficient $\lambda_{\text{SAE}} = 10^{-3}$.
- A jointly trained CBM with concept loss weight $\lambda_C = 1$.

All models are trained using the Adam optimiser [37] with batch size 512. The black-box model and CBM are trained with learning rate 10^{-3} for 30 epochs. The SAE is trained with learning rate 2×10^{-3} for 60 epochs, as unsupervised feature discovery typically requires longer training.

Fisher geometry. The Fisher metric is computed using the closed-form expression derived in Theorem 1. To obtain a stable task-aligned quadratic form, we average over activations satisfying

$$p(\mathbf{a}) \in (p_{\text{low}}, p_{\text{high}})^{1},$$

yielding $\bar{F}_A$ as defined above.

We examined several intervals for $(p_{\text{low}}, p_{\text{high}}) \in \{(0,1), (0.1,0.9), (0.2,0.8), (0.3,0.7), (0.4,0.6)\}$ and investigated their impact on the difference between γ_{F_A} in the cylindrical vs the spherical geometries (Figure 6). We found that the results were qualitatively comparable across different choices of interval, but quantitatively strongest $(p_{\text{low}}, p_{\text{high}}) = (0.2, 0.8)$, so we selected this interval for all experiments in the manuscript for consistency.

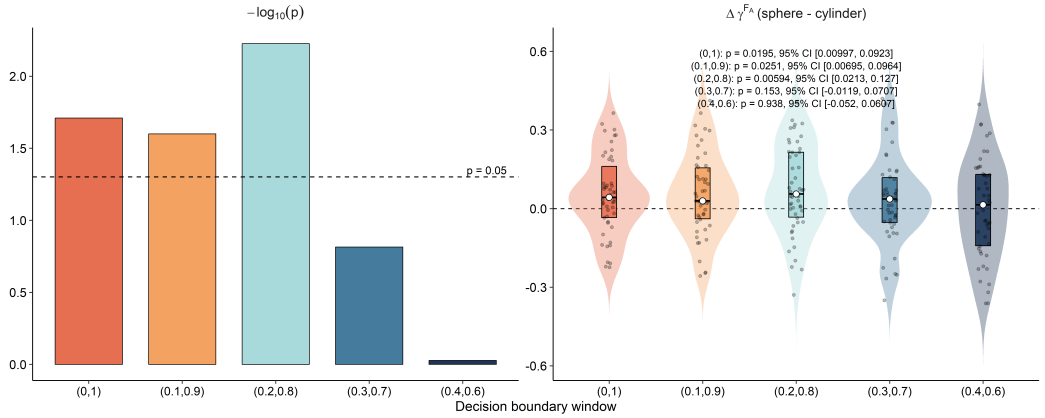

Figure 6: **Effect of decision-boundary window size on frustration detection.** Left: Bar plot shows $-\log_{10}(p)$ for the corresponding paired Wilcoxon tests comparing γ^{F_A} between spherical and cylindrical geometries. Right: Violin and box plots show the paired differences $\Delta\gamma^{F_A} = \gamma^{F_A}_{\text{sphere}} - \gamma^{F_A}_{\text{cylinder}}$ across 50 simulations for different decision-boundary windows $(p_{\text{low}}, p_{\text{high}})$ used to compute the Fisher metric. Across all boundary windows, γ^{F_A} is consistently higher in the spherical geometry than in the cylindrical geometry, indicating robust detection of concept frustration. The effect is quantitatively strongest for the intermediate window $(p_{\text{low}}, p_{\text{high}}) = (0.2, 0.8)$.

Statistical testing. For each metric, comparisons between cylindrical and spherical geometries are performed using a Wilcoxon signed-rank test across 50 repetitions.

D Synthetic data generation and and construction of the structured interaction matrix $\mathrm{M}(\alpha)$

In Section 4.1, we generated synthetic data from a linear-Gaussian concept model with controllable interactions between supervised and unsupervised concepts. We denote the full concept vector by

$$\mathbf{c} = (\mathbf{c}_k, \mathbf{c}_u) \in \mathbb{R}^k,$$

[1] Here, p represents probability, not the location vector $\mathbf{p}$.

where $\mathbf{c}_k \in \mathbb{R}^{k_{\text{known}}}$ contains the supervised (known) concepts and $\mathbf{c}_u \in \mathbb{R}^{k_{\text{unknown}}}$ the unsupervised (unknown) concepts, with $k = k_{\text{known}} + k_{\text{unknown}}$.

We first sampled a fixed symmetric positive definite covariance matrix

$$\mathbf{B}_{\text{known}} \in \mathbb{R}^{k_{\text{known}} \times k_{\text{known}}},$$

which defines the covariance structure among known concepts, and a fixed symmetric positive definite base covariance

$$\mathbf{B}_{\text{temp}} \in \mathbb{R}^{k_{\text{unknown}} \times k_{\text{unknown}}},$$

which captures the component of unknown-concept variation intrinsic to the unknown concepts. In the notation of the Results, $\mathbf{B}_{kk} \equiv \mathbf{B}_{\text{known}}$.

To control semantic consistency versus frustration between known and unknown concepts, we constructed an α-dependent interaction matrix

$$\mathbf{M}(\alpha) \in \mathbb{R}^{k_{\text{known}} \times k_{\text{unknown}}},$$

where the sign of α selects whether unknown concepts reinforce ($\alpha < 0$) or oppose ($\alpha > 0$) the correlation structure among known concepts, and $|\alpha|$ controls interaction strength.

Let

$$b_{ij} = (\mathbf{B}_{\text{known}})_{ij}, \qquad 1 \le i < j \le k_{\text{known}}.$$

Each unknown concept index $u \in \{1, \dots, k_{\text{unknown}}\}$ was assigned to mediate one selected known pair $(i(u), j(u))$. This assignment rule was fixed across all values of α, and all remaining entries of the corresponding column were set to zero. For each such u, we set two nonzero entries in column u of $\mathbf{M}(\alpha)$ as follows.

$$M(\alpha)_{i(u),\,u} = b_{i(u)j(u)}|\alpha|, \qquad M(\alpha)_{j(u),\,u} = -\operatorname{sign}(\alpha) \cdot |b_{i(u)j(u)}| \cdot |\alpha|.$$

At $\alpha = 0$, this construction gives $\mathbf{M}(0) = 0$, so known and unknown concepts are independent. In the notation of the Results, we identify

$$\mathbf{B}_{ku}(\alpha) \equiv \mathbf{M}(\alpha), \qquad \mathbf{B}_{uk}(\alpha) \equiv \mathbf{M}(\alpha)^\top.$$

Given $\mathbf{M}(\alpha)$, we defined the unknown-concept covariance as

$$\mathbf{B}_{uu}(\alpha) = \mathbf{B}_{\text{temp}} + \mathbf{M}(\alpha)^\top \mathbf{B}_{\text{known}}^{-1} \mathbf{M}(\alpha),$$

and assembled the full concept covariance matrix as

$$\mathbf{B}(\alpha) = \begin{pmatrix} \mathbf{B}_{\text{known}} & \mathbf{M}(\alpha) \\ \mathbf{M}(\alpha)^\top & \mathbf{B}_{\text{temp}} + \mathbf{M}(\alpha)^\top \mathbf{B}_{\text{known}}^{-1} \mathbf{M}(\alpha) \end{pmatrix}. \tag{23}$$

By Schur complement, $\mathbf{B}(\alpha)$ is symmetric positive definite for all α. Moreover,

$$\operatorname{Cov}(\mathbf{c}_k) = \mathbf{B}_{\text{known}}$$

is preserved across regimes, so varying α changes only the coupling between known and unknown concepts.

We then generated n independent concept samples according to

$$\mathbf{c}^{(i)} \sim \mathcal{N}(0, \mathbf{B}(\alpha)), \qquad i = 1, \dots, n.$$

Ground-truth task labels were generated from concepts through a noisy linear map,

$$\mathbf{a}^{(i)} = \mathbf{\Phi}\mathbf{c}^{(i)} + \boldsymbol{\epsilon}^{(i)},$$

where $\mathbf{\Phi} \in \mathbb{R}^{r \times k}$ has entries independently sampled from $\mathcal{N}(0, 1)$ and

$$\boldsymbol{\epsilon}^{(i)} \sim \mathcal{N}(0, \sigma_a^2 \, \mathbf{I}_r).$$

The parameter σ_a^2 therefore controls the difficulty of recovering concepts from activations.

To construct the task, we first sampled a base task vector

$$\boldsymbol{\psi}^* \sim \mathcal{N}(0, \mathbf{I}_k),$$

and defined

$$\boldsymbol{\psi} = \big((1-\omega)\boldsymbol{\psi}^*_{1:k_{\text{known}}},\ \omega\,\boldsymbol{\psi}^*_{k_{\text{known}}+1:k}\big),$$

with $\omega \in [0,1]$. Writing $\boldsymbol{\psi} = (\boldsymbol{\psi}_k, \boldsymbol{\psi}_u)$, the parameter ω controls the extent to which the task depends on known versus unknown concepts: $\omega = 0$ yields a task depending only on known concepts, whereas $\omega = 1$ yields a task depending only on unknown concepts.

Finally, we defined the latent task score

$$\tau^{(i)} = \boldsymbol{\psi}_k^\top \mathbf{c}_k^{(i)} + \boldsymbol{\psi}_u^\top \mathbf{c}_u^{(i)} + \eta^{(i)}, \qquad \eta^{(i)} \sim \mathcal{N}(0, \sigma_y^2),$$

and generated binary labels as

$$y^{(i)} = \mathbb{1}\{\tau^{(i)} > 0\}.$$

The parameter σ_y^2 controls the intrinsic noise of the task and therefore the difficulty of predicting y even from the full concept set.

E Proof of Theorem 2

Here we prove the following theorem:

Theorem (Concept-optimal Accuracy in the Linear–Gaussian Model)**.** *Let the concept vector be denoted* $\mathbf{c} = (\mathbf{c}_k, \mathbf{c}_u)$ *where* $\mathbf{c}_k = (c_1, \cdots, c_{k_{\text{known}}})$ *denotes known concepts and* $\mathbf{c}_u = (c_{k_{\text{known}}+1}, \cdots, c_k)$ *unknown concepts, under our model* $\mathbf{c} = (\mathbf{c}_k, \mathbf{c}_u) \sim \mathcal{N}(0, \mathbf{B})$ *with block covariance*

$$\mathbf{B} = \begin{pmatrix} \mathbf{B}_{kk} & \mathbf{B}_{ku} \\ \mathbf{B}_{uk} & \mathbf{B}_{uu} \end{pmatrix},$$

where

$$\mathbf{B}_{uu}(\alpha) = \mathbf{B}_{temp} + \mathbf{B}_{uk}(\alpha)\mathbf{B}_{kk}^{-1}\mathbf{B}_{ku}(\alpha),$$

and $\mathbf{B}_{kk}$ *is positive definite (and thus invertible).*

Let the weight vector be denoted $\boldsymbol{\psi} = (\boldsymbol{\psi}_k, \boldsymbol{\psi}_u)$ *where* $\boldsymbol{\psi}_k = (w_1, \cdots, w_{k_{\text{known}}})$ *denotes known concept-task weights and* $\boldsymbol{\psi}_u = (w_{k_{\text{known}}+1}, \cdots, w_k)$ *unknown concept-task weights.*

Let the task variable be

$$\tau = \boldsymbol{\psi}_k^\top \mathbf{c}_k + \boldsymbol{\psi}_u^\top \mathbf{c}_u + \eta, \qquad \eta \sim \mathcal{N}(0, \sigma_y^2),$$

and define the binary label

$$y = \mathbb{1}\{\tau > 0\}.$$

Then the concept-optimal Bayes classifier using only the known concepts $\mathbf{c}_k$*, which minimises the risk*

$$R(\hat{y}) = \mathbb{E}_{(\mathbf{c}_k, y)}\left[\mathbb{1}\{\hat{y}(\mathbf{c}_k) \neq y\}\right],$$

is

$$\hat{y}(\mathbf{c}_k) = \mathbb{1}\{\boldsymbol{\phi}(\alpha)^\top \mathbf{c}_k > 0\},$$

and its classification accuracy is

$$\text{Acc}_{\textit{CBM}}(\alpha) = \frac{1}{2} + \frac{1}{\pi}\arctan\left(\sqrt{\frac{\boldsymbol{\phi}(\alpha)^\top \mathbf{B}_{kk}\boldsymbol{\phi}(\alpha)}{\boldsymbol{\psi}_u^\top \mathbf{B}_{temp}\boldsymbol{\psi}_u + \sigma_y^2}}\right),$$

where

$$\boldsymbol{\phi}(\alpha) = \boldsymbol{\psi}_k + \mathbf{B}_{kk}^{-1}\mathbf{B}_{ku}(\alpha)\boldsymbol{\psi}_u,$$

Proof. To compute the optimal classifier given known concepts, we must calculate the conditional distribution $\tau|\mathbf{c}_k$.

We note that as our toy model is linear-Gaussian this conditional distribution will also be Gaussian and can thus be explicitly calculated via $\mathbb{E}[\tau|\mathbf{c}_k]$ and $\text{Var}(\tau|\mathbf{c}_k)$.

We first note that

$$\begin{aligned}\mathbb{E}[\tau|\mathbf{c}_k] &= \mathbb{E}[\boldsymbol{\psi}_k^\top \mathbf{c}_k + \boldsymbol{\psi}_u^\top \mathbf{c}_u + \eta|\mathbf{c}_k] \\ &= \boldsymbol{\psi}_k^\top \mathbf{c}_k + \boldsymbol{\psi}_u^\top \mathbb{E}[\mathbf{c}_u|\mathbf{c}_k] \\ &= \boldsymbol{\psi}_k^\top \mathbf{c}_k + \boldsymbol{\psi}_u^\top \mathbf{B}_{uk}\mathbf{B}_{kk}^{-1}\mathbf{c}_k \\ &= (\boldsymbol{\psi}_k + \mathbf{B}_{kk}^{-1}\mathbf{B}_{ku}\boldsymbol{\psi}_u)^\top \mathbf{c}_k \\ &= \boldsymbol{\phi}^\top \mathbf{c}_k.\end{aligned}$$

Which follows from linearity of expectation, the fact that $\eta \sim \mathcal{N}(0, \sigma_y^2)$, the fact that $\mathbf{c} \sim \mathcal{N}(0, \mathbf{B})$ and the conditional distribution of multivariate Gaussian distributions, and where we have defined $\boldsymbol{\phi} = \boldsymbol{\psi}_k + \mathbf{B}_{kk}^{-1}\mathbf{B}_{ku}\boldsymbol{\psi}_u$. We note that $\boldsymbol{\phi} := \boldsymbol{\phi}(\alpha)$ is a function of our frustration parameter α.

We now consider

$$\begin{aligned}\text{Var}[\tau|\mathbf{c}_k] &= \text{Var}[\boldsymbol{\psi}_k^\top \mathbf{c}_k + \boldsymbol{\psi}_u^\top \mathbf{c}_u + \eta|\mathbf{c}_k] \\ &= \boldsymbol{\psi}_u^\top \text{Var}[\mathbf{c}_u|\mathbf{c}_k]\boldsymbol{\psi}_u + \sigma_y^2 \\ &= \boldsymbol{\psi}_u^\top (\mathbf{B}_{uu} - \mathbf{B}_{uk}\mathbf{B}_{kk}^{-1}\mathbf{B}_{ku})\boldsymbol{\psi}_u + \sigma_y^2 \\ &= \boldsymbol{\psi}_u^\top \mathbf{B}_{\text{temp}}\boldsymbol{\psi}_u + \sigma_y^2.\end{aligned}$$

Which follows from the variance of a constant being zero the fact that $\eta \sim \mathcal{N}(0, \sigma_y^2)$ and is an independent variable, the fact that $\mathbf{c} \sim \mathcal{N}(0, \mathbf{B})$, the conditional distribution of multivariate Gaussian distributions, and the construction of $\mathbf{B}$ which ensures $\mathbf{B}_{\text{temp}} = \mathbf{B}_{uu} - \mathbf{B}_{uk}\mathbf{B}_{kk}^{-1}\mathbf{B}_{ku}$. We note that via our construction this variance is not dependent on α, the frustration parameter.

Whence $\tau|\mathbf{c}_k \sim \mathcal{N}(\boldsymbol{\phi}(\alpha)^\top \mathbf{c}_k, \boldsymbol{\psi}_u^\top \mathbf{B}_{\text{temp}}\boldsymbol{\psi}_u + \sigma_y^2)$. By definition the concept-optimal Bayes predictor is the risk minimiser

$$R(\hat{y}) = \mathbb{E}_{(\mathbf{c}_{\text{known}}, y)}\left[\ell(\hat{y}(\mathbf{c}_{\text{known}}), y)\right],$$

with respect to the 0-1 loss. Since $\mathbb{E}\left[\ell(\hat{y}(\mathbf{c}_{\text{known}}), y)\right] = \mathbb{E}\left[\mathbb{E}\left[\ell(\hat{y}(\mathbf{c}_{\text{known}}), y) \mid \mathbf{c}_{\text{known}}\right]\right]$, which is minimised by setting

$$\hat{y}(\mathbf{c}_{\text{known}}) = \text{argmin}_{z\in\{0,1\}}\mathbb{E}\left[\ell(z, y) \mid \mathbf{c}_{\text{known}}\right].$$

When $l(z, y)$ is the 0-1 loss, we obtain as usual $\hat{y}(\mathbf{c}_{\text{known}}) = \text{argmax}_z \mathbb{P}(Y = z|\mathbf{c}_k = \mathbf{c}_{\text{known}})$. Hence a concept-optimal model will predict $y = 1$ iff $\mathbb{P}(Y = 1|\mathbf{c}_k = \mathbf{c}_{\text{known}}) > 1/2$, and thus will predict $y = 1$ when $\mathbb{P}[\tau > 0|\mathbf{c}_{\text{known}}] > 1/2$ and zero otherwise. It therefore follows that the optimal classifier given $\mathbf{c}_k$ is simply:

$$\hat{y}(\mathbf{c}_k) = \mathbb{1}\{\boldsymbol{\phi}(\alpha)^\top \mathbf{c}_k > 0\}. \tag{24}$$

As the true outcome is $y = \mathbb{1}\{\tau > 0\}$ We note that the optimal classifier makes an error when $\text{sign}(\boldsymbol{\phi}(\alpha)^\top \mathbf{c}_k) \neq \text{sign}(\tau)$.

Let us define $S := \mathbb{E}[\tau|\mathbf{c}_k] = \boldsymbol{\phi}(\alpha)^\top \mathbf{c}_k$ and $R := \tau - S$. We note that as $\mathbf{c}_k \sim \mathcal{N}(0, \mathbf{B}_{kk})$, it follows that $S \sim \mathcal{N}(0, \boldsymbol{\phi}(\alpha)^\top \mathbf{B}_{kk}\boldsymbol{\phi}(\alpha))$, we denote $\sigma_S^2 = \boldsymbol{\phi}(\alpha)^\top \mathbf{B}_{kk}\boldsymbol{\phi}(\alpha)$. We also note that

$$\begin{aligned}R &= \tau - S \\ &= \boldsymbol{\psi}_k^\top \mathbf{c}_k + \boldsymbol{\psi}_u^\top \mathbf{c}_u + \eta - (\boldsymbol{\psi}_k + \mathbf{B}_{kk}^{-1}\mathbf{B}_{ku}\boldsymbol{\psi}_u)^\top \mathbf{c}_k \\ &= \boldsymbol{\psi}_u^\top (\mathbf{c}_u - \mathbf{B}_{uk}\mathbf{B}_{kk}^{-1}\mathbf{c}_k) + \eta.\end{aligned}$$

It is clear that $\mathbb{E}[R] = 0$. We note that

$$\begin{aligned}\text{Var}[R] &= \mathbb{E}[(\boldsymbol{\psi}_u^\top (\mathbf{c}_u - \mathbf{B}_{uk}\mathbf{B}_{kk}^{-1}\mathbf{c}_k))^2] + 2\mathbb{E}[\eta]\mathbb{E}[\boldsymbol{\psi}_u^\top (\mathbf{c}_u - \mathbf{B}_{uk}\mathbf{B}_{kk}^{-1}\mathbf{c}_k)]] + \mathbb{E}[\eta^2] \\ &= \mathbb{E}[(\boldsymbol{\psi}_u^\top (\mathbf{c}_u - \mathbf{B}_{uk}\mathbf{B}_{kk}^{-1}\mathbf{c}_k))^2] + \sigma_y^2 \\ &= \boldsymbol{\psi}_u^\top \mathbb{E}[(\mathbf{c}_u - \mathbf{B}_{uk}\mathbf{B}_{kk}^{-1}\mathbf{c}_k)^2]\boldsymbol{\psi}_u + \sigma_y^2 \\ &= \boldsymbol{\psi}_u^\top \mathbb{E}[(\mathbf{c}_u - \mathbb{E}[\mathbf{c}_u|\mathbf{c}_k])^2]\boldsymbol{\psi}_u + \sigma_y^2 \\ &= \boldsymbol{\psi}_u^\top \text{Var}[\mathbf{c}_u \mid \mathbf{c}_k]\boldsymbol{\psi}_u + \sigma_y^2 \\ &= \boldsymbol{\psi}_u^\top (\mathbf{B}_{uu} - \mathbf{B}_{uk}B_{kk}^{-1}\mathbf{B}_{ku})\boldsymbol{\psi}_u + \sigma_y^2,\end{aligned}$$

which follows from the properties of a multivariate Gaussian. Thus $R \sim \mathcal{N}(0, \boldsymbol{\psi}_u^\top(\mathbf{B}_{uu} - \mathbf{B}_{uk}\mathbf{B}_{kk}^{-1}\mathbf{B}_{ku})\boldsymbol{\psi}_u + \sigma_y^2)$, we denote $\sigma_R^2 = \boldsymbol{\psi}_u^\top(\mathbf{B}_{uu} - \mathbf{B}_{uk}\mathbf{B}_{kk}^{-1}\mathbf{B}_{ku})\boldsymbol{\psi}_u + \sigma_y^2$.

Thus the accuracy of the Bayes optimal classifier is $\mathbb{P}[\text{sign}(S) = \text{sign}(S + R)]$. We note that S and R are independent Gaussian variables (i.e., $\text{Cov}(R, S) = 0$) and that $\text{Var}(S + R) = \sigma_R^2 + \sigma_S^2$ and $\text{Cov}(S, S + R) = \sigma_S^2$, thus the correlation co-efficient between $S + R$ and S, is $\rho_{S,S+R} = \frac{\sigma_S}{\sqrt{\sigma_S^2+\sigma_R^2}}$.

It can therefore be demonstrated through simple integration that the accuracy of the Bayes-optimal CBM given $\mathbf{c}_k$ is given by:

$$\begin{aligned}
\mathbb{P}[\text{sign}(S) = \text{sign}(S + R)] &= \frac{1}{2} + \frac{1}{\pi}\arcsin\rho_{S,S+R} \\
&= \frac{1}{2} + \frac{1}{\pi}\arcsin\left(\frac{\sigma_S}{\sqrt{\sigma_S^2 + \sigma_R^2}}\right) \\
&= \frac{1}{2} + \frac{1}{\pi}\arctan\left(\frac{\sigma_S}{\sigma_R}\right) \\
&= \frac{1}{2} + \frac{1}{\pi}\arctan\left(\sqrt{\frac{\boldsymbol{\phi}(\alpha)^\top\mathbf{B}_{kk}\boldsymbol{\phi}(\alpha)}{\boldsymbol{\psi}_u^\top(\mathbf{B}_{uu} - \mathbf{B}_{uk}\mathbf{B}_{kk}^{-1}\mathbf{B}_{ku})\boldsymbol{\psi}_u + \sigma_y^2}}\right) \\
&= \frac{1}{2} + \frac{1}{\pi}\arctan\left(\sqrt{\frac{\boldsymbol{\phi}(\alpha)^\top\mathbf{B}_{kk}\boldsymbol{\phi}(\alpha)}{\boldsymbol{\psi}_u^\top\mathbf{B}_{temp}\boldsymbol{\psi}_u + \sigma_y^2}}\right)
\end{aligned}$$

We denote this quantity as $\text{Acc}_{\text{CBM}}(\alpha)$. □

F Experiments

All experiments were run on one NVIDIA DGX Spark (128GB of unified system memory), and overall required <1 hour of training time.

F.1 Experiments with synthetic data

We used the linear-Gaussian synthetic data generator defined above to study interactions between supervised ("known") and unsupervised ("unknown") concepts under controlled covariance structure.

For each simulation we generated:

$$\begin{aligned}
n &= 8000, \\
k &= 50, \\
k_{\text{known}} &\in \{10, 20, 30, 40\}, \\
r &= 64, \\
\sigma_a &= 0.3, \\
\sigma_y &= 1.5.
\end{aligned}$$

Concepts were sampled as

$$\mathbf{c} = (\mathbf{c}_{\text{known}}, \mathbf{c}_{\text{unknown}}) \sim \mathcal{N}(0, \mathbf{B}(\alpha)),$$

where $\alpha \in \{-1, 0, 1\}$ controls semantic alignment between known and unknown concepts:

- $\alpha = -1$: semantic alignment (consistent)
- $\alpha = 0$: independence
- $\alpha = +1$: semantic opposition (frustration)

The outcome was generated as

$$Y = \boldsymbol{\psi}_k^\top\mathbf{c}_{\text{known}} + \omega\boldsymbol{\psi}_u^\top\mathbf{c}_{\text{unknown}} + \varepsilon,$$

where $\omega \in \{0, 0.25, 0.5, 0.75, 1\}$ controls the proportion of task signal arising from unknown concepts and $\varepsilon \sim \mathcal{N}(0, \sigma_y^2)$.

Observed features were generated via a linear projection

$$X = \mathbf{A}\mathbf{c} + \eta,$$

where $\eta \sim \mathcal{N}(0, \sigma_a^2 I)$.

Each of the 60 parameter regimes was simulated across 10 independent seeds (600 datasets total). Data were split 75:25 into train and test sets.

Robustness to SAE construction Selecting the maximally frustrating SAE atom could make the frustration score sensitive to dictionary size, sparsity regularisation or optimisation initialisation. We therefore performed an additional synthetic robustness analysis, comparing frustrated ($\alpha = +1$) and consistent ($\alpha = -1$) regimes across dictionary sizes $K_{\mathrm{SAE}} \in \{30, 60, 120, 240, 480\}$ and sparsity penalties $\lambda_{\mathrm{SAE}} \in \{0.01, 0.1, 1, 3\}$. Each configuration was evaluated across eight data-generating seeds and three SAE initialisations per seed, yielding 480 paired evaluations. Within each seed and regime, the data, black-box model, Fisher geometry and supervised concept directions were held fixed while varying the SAE configuration.

Our primary endpoint was the discrimination gap $\Delta\gamma_{F_A} = \gamma_{F_A}(\alpha = +1) - \gamma_{F_A}(\alpha = -1)$. Table 1 reports marginal means, averaging over the remaining SAE settings, seeds and initialisations. Although absolute frustration scores increased with dictionary size, the mean gap remained positive and stable across all dictionary-size and sparsity combinations.

Table 1: Robustness of Fisher frustration to SAE construction. Results are averaged across eight data-generating seeds and three SAE initialisations. Hyperparameter rows report marginal means over the remaining settings; gaps are calculated before rounding.

Hyperparameter	Setting	γ_{F_A} (Frustrated)	γ_{F_A} (Consistent)	$\Delta\gamma_{F_A}$
Dictionary size	30	0.042	0.025	0.0173
	60	0.052	0.035	0.0173
	120	0.068	0.050	0.0186
	240	0.089	0.071	0.0187
	480	0.106	0.090	0.0166
Sparsity penalty	0.01	0.073	0.055	0.0179
	0.1	0.072	0.055	0.0176
	1	0.072	0.054	0.0179
	3	0.070	0.053	0.0174

F.2 Language experiment (sarcasm detection)

Dataset and embeddings. We used the News Headlines sarcasm dataset (28,000 headlines; 13,000 labelled sarcastic) [23]. Each headline was embedded using DeBERTa-v3-base, producing a 768-dimensional representation $\mathbf{a}$ [24].

Concept construction. We defined three continuous sentiment concepts derived from RoBERTa-base-sentiment logits [25]: C_1 (positive sentiment), C_2 (negative sentiment), and C_3 (sentiment intensity), defined as the negative neutrality logit. In this dataset, C_1 and C_2 are inversely correlated, while C_3 is positively correlated with both. A linear model confirmed that all three concepts are independently associated with sarcasm.

Models and training. Two CBMs were trained using 10-fold cross-validation: CBM1 with known concepts (C_1, C_2) and CBM2 with known concepts (C_1, C_2, C_3). Both models were trained using joint optimisation of concept reconstruction loss ($\lambda_C = 1$) and task loss for 30 epochs with learning rate 10^{-3} and evaluated using 10-fold cross-validation. A one-hidden-layer MLP was trained with width $h = 128$ for 60 epochs. We also trained an SAE with a learning rate of 2×10^{-3} for 60 epochs with $k_{\mathrm{SAE}} = 300$ latent units and a sparsity coefficient $\lambda_{\mathrm{SAE}} = 10^{-3}$. All models were trained using the Adam optimiser with a batch size of 512.

Evaluation. We computed task accuracy, Euclidean frustration $\mathrm{Frust}^{E}(1,2)$, and Fisher frustration $\mathrm{Frust}^{F_A}(1,2)$. Frustration was evaluated on the shared concept subspace (C_1, C_2). Paired Wilcoxon signed-rank tests were used across folds.

F.3 Vision experiments across frustrated CUB concept triples

Dataset and embeddings. We used the CUB-200-2011 dataset [26], available at https://data.caltech.edu/records/65de6-vp158, to construct three binary classification tasks: Flycatcher–Oriole, Grosbeak–Oriole and Gull–Tern. Images were embedded using the pretrained CLIP ViT-B/16 encoder, yielding 512-dimensional representations $\mathbf{a}$ [27].

Concept triple selection. Within each task, we identified predictive concepts from the ground-truth binary attribute annotations, requiring $|\mathrm{cor}(C_i, Y)| \geq 0.20$. Among these concepts, we selected triples satisfying

$$\mathrm{sign}(\mathrm{cor}(C_1, C_2)) \neq \mathrm{sign}(\mathrm{cor}(C_1, C_3)\mathrm{cor}(C_2, C_3)),$$

with absolute correlations of at least 0.05 on all three concept–concept edges. Selection used annotation correlations before computing Fisher frustration, rather than observed frustration scores. We randomly sampled 400 qualifying triples per task where more were available and retained all qualifying triples otherwise. This yielded 400 Flycatcher–Oriole, 400 Grosbeak–Oriole and 45 Gull–Tern triples: 845 triples in total.

Models and training. For each task, we used one stratified 70:30 training–test split with seed 0. Embeddings were standardised using the training-set mean and standard deviation. We trained a shared black-box classifier with one hidden layer of width 128 for 30 epochs and computed its Fisher quadratic form on the training inputs. A shared SAE with 300 dictionary atoms was trained for 60 epochs with sparsity penalty 10^{-3}. The black-box model, Fisher quadratic form and SAE dictionary were reused across all triples within each task.

For each triple, CBM1 supervised (C_1, C_2) and CBM2 supervised (C_1, C_2, C_3). Both were linear CBMs trained jointly on concept and task prediction for 30 epochs, with concept-loss weight $\lambda_C = 1$. Frustration was evaluated only on the shared pair (C_1, C_2), using the same task-level Fisher quadratic form and SAE dictionary. For CBM2, the concept-weight matrix was restricted to its first two rows before scoring.

Statistical analysis. We compared CBM1 and CBM2 using two-sided paired Wilcoxon signed-rank tests across triples. Paired Cohen's d_z was calculated as

$$d_z = \frac{\mathrm{mean}(\delta_i)}{\mathrm{sd}(\delta_i)}, \qquad \delta_i = \mathrm{Frust}_{\mathrm{CBM1},i} - \mathrm{Frust}_{\mathrm{CBM2},i},$$

using the sample standard deviation. Positive values indicate higher frustration in CBM1. AUROC measures discrimination between CBM1 and CBM2 scores, with higher scores predicting CBM1. Pooled analyses concatenate triples across the three tasks.

F.4 Permutation control and omitted-concept recovery

Experimental design. We used the same 845 frustrated concept triples described in Appendix F.3. For each triple, we trained CBM1 on (C_1, C_2) and compared two ontology expansions: CBM2 supervised the true omitted concept C_3, whereas CBM2-permuted supervised a randomly permuted version of its labels. Permutations were performed across the task images, preserving the overall prevalence of C_3 while disrupting its relationship with the inputs and task. We generated five independent permutations per triple.

Within each triple, CBM1 was trained once and reused. CBM2 and all five permutation-control models were trained from an identical initialisation template and with an identical minibatch order, so that only the added concept labels differed between these arms. Models were trained for 30 epochs using Adam with learning rate 10^{-3}, batch size 512 and concept-loss weight $\lambda_C = 1$. We used one task-stratified 70:30 training–test split. A black-box classifier, and SAE dictionary with 300 atoms were fitted once per task and shared across triples. Frustration was evaluated only on the shared pair (C_1, C_2) in every model.

Specificity of scalar frustration reduction. For each triple i, we defined the reduction following addition of the true concept as

$$R_i^{\text{true}} = \text{Frust}^{F_A}_{\text{CBM1},i}(1,2) - \text{Frust}^{F_A}_{\text{CBM2},i}(1,2),$$

and calculated the corresponding reduction $R_{i,d}^{\text{perm}}$ for each permutation $d \in \{1,\ldots,5\}$. Our paired endpoint was

$$D_i = R_i^{\text{true}} - \frac{1}{5}\sum_{d=1}^{5} R_{i,d}^{\text{perm}},$$

so positive values indicate a greater reduction after adding the true concept. The pooled mean difference was approximately 0.00007 (95% bootstrap CI $[-0.00042, 0.00056]$), with paired Cohen's $d_z = 0.009$ and a two-sided paired Wilcoxon $p = 0.308$. Thus, we found no evidence that adding the true concept produced a greater scalar reduction than adding its permuted labels.

Recovery of the omitted concept. For each triple, we selected the SAE atom identified as maximally frustrating by CBM1 and evaluated the association between its encoder activation and the omitted concept labels. Selection used the known concept directions, Fisher geometry and SAE dictionary, without using C_3 labels in the atom-selection objective. The same selected atom was evaluated against the true labels and each of the five permutations.

Recovery was measured using direction-free single-feature AUROC,

$$A_{\text{df}} = \max(A, 1 - A),$$

where A is the AUROC obtained using the atom activation to predict the concept label. This measures association regardless of activation orientation. For each triple, permutation-control AUROCs were averaged across the five draws before calculating the paired true-minus-permuted difference.

Interpretation and uncertainty. The selected atom recovered the true omitted concept better than its permuted labels (mean direction-free AUROC 0.603 versus 0.526; paired difference 0.077, 95% CI $[0.072, 0.082]$), with pooled paired Cohen's $d_z = 1.00$. These results support association with genuine omitted semantic structure.

G Code and data availability

Code and data to reproduce the results of this paper are available at `https://github.com/CRSBanerji/concept-frustration`.

H Declaration of LLM usage

We used LLMs for editing the manuscript and for generating some of the code used for running the experiments. All ideas, methods, analyses and conclusions were conceived by the authors. Every LLM suggestion was manually scrutinized, reviewed and edited before input to the manuscript or code. No LLM outputs were used without our manual supervision. The authors retain full responsibility for the content.

I Supplementary Tables

Metric	Cylinder	Sphere	Hodges–Lehmann median paired difference (sphere−cylinder)	p
BB acc.	0.876 [0.874, 0.878]	0.894 [0.891, 0.896]	0.0180 [0.0152, 0.0207]	$\mathbf{< 10^{-4}}$
CBM acc.	0.811 [0.759, 0.862]	0.824 [0.782, 0.866]	0.0132 [0.00805, 0.0177]	$\mathbf{3.9 \times 10^{-3}}$
Concept MSE	2.81e−2 [2.58, 3.05]e−2	2.99e−2 [2.70, 3.29]e−2	1.10e−3 [−6.34, 3.15]e−3	0.284
γ^{F_A}	0.126 [0.0896, 0.163]	0.198 [0.163, 0.233]	7.25e−2 [0.0213, 0.127]	$\mathbf{5.94 \times 10^{-3}}$
γ^{E}	4.85e−2 [4.30, 5.39]e−2	4.47e−2 [3.88, 5.05]e−2	−3.52e−3 [−8.25, 9.15]e−4	0.118

Table S1: Comparison of performance and geometric metrics between cylindrical (flatworld) and spherical (round-world) geometries. Values are means (top) with 95% confidence intervals (bottom). Paired Wilcoxon signed-rank tests compare matched repetitions (sphere minus cylinder).

Metric	$\alpha = 0$ Independent	$\alpha = -1$ Consistent	$\alpha = 1$ Frustrated	Median diff (+1 minus -1) (95% CI)	p $+1$ vs -1
BB acc.	0.935 [0.932, 0.938]	0.936 [0.930, 0.943]	0.936 [0.929, 0.944]	2.03e−3 [−5.02e−4, 4.77e−3]	0.132
CBM acc.	0.783 [0.769, 0.797]	0.718 [0.703, 0.734]	0.699 [0.684, 0.714]	−1.63e−2 [−2.25, −1.03]e−2	$< \mathbf{10^{-4}}$
Concept MSE	2.10 [2.05, 2.15]	40.9 [26.0, 55.8]	63.7 [43.4, 84.1]	6.23 [4.42, 12.5]	$< \mathbf{10^{-4}}$
β	26.0 [23.8, 28.3]	258 [168, 347]	373 [264, 483]	39.8 [15.0, 74.0]	$< \mathbf{10^{-4}}$
γ^{F_A}	1.60e−2 [1.48, 1.71]e−2	2.76e−2 [2.51, 3.02]e−2	3.67e−2 [3.31, 4.04]e−2	7.76e−3 [5.47e−3, 1.00e−2]	$< \mathbf{10^{-4}}$
γ^{E}	7.62e−3 [7.38, 7.85]e−3	7.43e−3 [6.96, 7.90]e−3	7.99e−3 [7.37, 8.61]e−3	1.51e−4 [3.28e−5, 3.35e−4]	0.0108
T_1	251 [195, 307]	251 [195, 307]	251 [195, 307]	–	–
T_2	0 [0, 0]	−0.327 [−0.982, 0.327]	0.404 [−0.448, 1.26]	0.304 [−1.34, 2.07]	0.742
T_3	0 [0, 0]	2.76e6 [−6.27e5, 6.15e6]	3.27e5 [−6.18e4, 7.15e5]	615 [237, 1.37e3]	$< \mathbf{10^{-4}}$
T_4	326 [255, 398]	326 [255, 398]	326 [255, 398]	–	–

Metric	Median diff (+1 minus 0) (95% CI)	p 0 vs +1	Median diff (-1 minus 0) (95% CI)	p 0 vs -1
BB acc.	5.24e−3 [7.40e−4, 1.02e−2]	0.0232	2.70e−3 [1.60e−5, 5.98e−3]	0.050
CBM acc.	−7.72e−2 [−8.83e−2, −6.66e−2]	$< \mathbf{10^{-4}}$	−5.83e−2 [−6.72e−2, −4.92e−2]	$< \mathbf{10^{-4}}$
Concept MSE	20.2 [11.8, 25.7]	$< \mathbf{10^{-4}}$	5.75 [4.23, 14.3]	$< \mathbf{10^{-4}}$
β	136 [69.9, 176]	$< \mathbf{10^{-4}}$	44.3 [28.6, 79.7]	$< \mathbf{10^{-4}}$
γ^{F_A}	1.70e−2 [1.44e−2, 1.99e−2]	$< \mathbf{10^{-4}}$	9.29e−3 [7.30e−3, 1.14e−2]	$< \mathbf{10^{-4}}$
γ^{E}	−9.00e−4 [−1.11e−3, −5.96e−4]	$< \mathbf{10^{-4}}$	−9.59e−4 [−1.14e−3, −7.72e−4]	$< \mathbf{10^{-4}}$
T_1	–	–	–	–
T_2	0.934 [−0.211, 2.02]	0.112	−0.277 [−1.26, 0.95]	0.601
T_3	2.06e3 [1.15e3, 3.47e3]	$< \mathbf{10^{-4}}$	639 [386, 953]	$< \mathbf{10^{-4}}$
T_4	–	–	–	–

Table S2: Summary statistics across concept regimes defined by α. The upper block reports regime means with 95% confidence intervals together with the paired comparison between the frustrated ($\alpha = 1$) and consistent ($\alpha = -1$) settings. The lower block reports paired Hodges–Lehmann median shift estimates with 95% confidence intervals and paired Wilcoxon signed-rank p-values for comparisons against the independent baseline ($\alpha = 0$).

Metric	CBM1 Mean (95% CI)	CBM2 Mean (95% CI)	Median diff (CBM1 − CBM2)	p
CBM acc.	0.728 [0.657, 0.798]	0.805 [0.777, 0.834]	−7.20e−2 [−1.77e−1, 1.04e−3]	0.058
γ^{E}	1.01e−3 [6.59e−4, 1.36e−3]	1.31e−3 [8.00e−4, 1.82e−3]	−2.75e−4 [−7.82e−4, 1.31e−4]	0.492
γ^{F_A}	4.47e−2 [−1.71e−2, 1.06e−1]	5.30e−3 [−1.83e−3, 1.24e−2]	3.13e−3 [3.21e−4, 1.12e−1]	**0.0488**

Table S3: Comparison of CBM1 and CBM2 performance metrics in the language task. Values are mean (top) with 95% confidence intervals (bottom). Median differences are Hodges–Lehmann estimates from paired Wilcoxon signed-rank tests (CBM1 − CBM2).

Metric	CBM1 Mean (95% CI)	CBM2 Mean (95% CI)	Median diff (CBM1 − CBM2)	p
CBM acc.	0.639 [0.547, 0.731]	0.755 [0.680, 0.829]	−1.27e−1 [−2.62e−1, 2.25e−2]	0.0972
γ^{E}	7.22e−3 [6.48e−3, 7.95e−3]	6.98e−3 [5.87e−3, 8.09e−3]	4.29e−4 [−9.77e−4, 1.46e−3]	0.557
γ^{F_A}	6.79e−2 [5.67e−3, 1.30e−1]	6.49e−3 [−3.83e−3, 1.68e−2]	5.71e−2 [4.65e−5, 1.22e−1]	**0.0488**

Table S4: Comparison of CBM1 and CBM2 on the CUB gull–tern classification task using CLIP ViT-B/16 embeddings. Values are mean (top) with 95% confidence intervals (bottom). Median differences are Hodges–Lehmann estimates from paired Wilcoxon signed-rank tests (CBM1 − CBM2).

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We mention eight primary contributions in the abstract and introduction: introducing a geometric framework to compare supervised human concepts and unsupervised model representations (Section 3.1), formalizing concept frustration (Section 3.2), developing task-aligned similarity measures to detect frustration (Sections 3.2 & 3.3, and Figure 2), showing that frustration is identifiable in the task-aligned geometry while Euclidean comparisons fail (Section 4.6, Figure 4, and Tables S3 & S4), deriving a closed-form Bayes-optimal expression for a linear Gaussian model (Section 4.2 & Appendix E), identifying where frustration impacts performance (Sections 4.3 & 4.4), and carrying out experiments on synthetic data and real language and vision tasks (Sections 4.3, 4.4. 4.5 & 4.6).

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We include a "Limitations and future work" section (Section 6) where we explicitly describe three limitations of our work. These are: the specificity of our theoretical analysis to linear-Gaussian models, the limitations of using the maximally frustrating direction as a measure of pairwise frustration, and the remaining need to generalize our derivation of task-aligned geometry to a broader range of model architectures.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.

- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: We prove our two main theorems in Sections B and E. Each proof was checked for correctness by multiple authors.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide the code and data at `https://github.com/CRSBanerji/concept-frustration` alongside a README file that provides instructions on running the code. We do not include the CUB-200-2011 dataset due to size constraints, but we provide a link to it in Section F.3. In this section, we also provide full experimental details, including training hyperparameters and model architecture details. These allow for the reproduction of our experimental results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example

(a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The code and data used to produce our experimental results can be found at `https://github.com/CRSBanerji/concept-frustration`. We could not include the CUB-200-2011 dataset due to size constraints, but we provide a link to it in Section F.3 along with the rest of our experimental details. The repository provides a README file that provides instructions on reproducing our experiments.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We included experimental details in Section F.2, F.3, F.4. This includes optimizer choice, training hyperparameters (learning rate, batch size & number of epochs) and details of model architecture.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.

- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Where relevant, we have provided error bars, Wilcoxon p-values and confidence intervals (95%). These are included in Figures 2B, 3, 4, 5 & 6b.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide a specification of compute resources and a rough upper limit of execution time in Section F.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have carefully reviewed the NeurIPS Code of Ethics and ensured that our paper conforms to it in every aspect.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: This work is primarily foundational, focusing on comparing human-supervised and unsupervised concept representations in predictive models. However, we explicitly discuss the accuracy–interpretability trade-off and broader societal implications in high-consequence domains (e.g., medicine, scientific discovery, criminal justice) in both the introduction and discussion.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: [N/A]

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.

- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Two published datasets were used: 28,000 news headlines labelled with sarcasm and the CUB-200-2011 dataset describing images of birds and associated concepts and species labels were used. Both published datasets are cited according to the requests of the original owners. Three open-source pre-trained models were used: DeBERTa-v3-base, RoBERTa-base-sentiment and CLIP ViT-B/16 vision encoder. All models are cited according to the requests of the original owners.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The only new asset we introduce is our code, and we provide this at `https://github.com/CRSBanerji/concept-frustration` with a README that provides full documentation. Details on training, architecture, and limitations can be found in the paper.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

Justification: We did not conduct any crowdsourcing experiments or experiments involving human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: We did not conduct any crowdsourcing experiments or experiments involving human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: We have included an LLM usage declaration in the appendix (Section H).

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.