A UNIFYING FRAMEWORK OF CONCEPT-BASED EXPLAINABLE AI WITH COMPLETENESS GUARANTEES

Anonymous authors
Paper under double-blind review

ABSTRACT

Concept-based explanations describe neural network predictions through human-understandable properties of inputs called *concepts*. The field encompasses approaches that differ in how they define and represent concepts and connect them to model predictions. We introduce a *theoretical framework* that describes these approaches in a common mathematical language and supports a shared analysis of their properties. For *concept discovery*, which identifies concepts automatically within a latent space of a trained model, we employ a *concept autoencoder* view. An *encoder* extracts concept representations from the model's latent space, and a *decoder* uses them to reconstruct the original latent representation. The autoencoder's *reconstruction error* measures how accurately its decoder recovers the original latent representation. We revisit *model completeness*: how well the concepts can reproduce the model's outputs. We show that model incompleteness of the concepts can be bounded by the autoencoder's reconstruction error. The autoencoder view also provides a common way to define individual concept *attributions*, which measure each concept's contribution to a prediction. We establish when these attributions sum to the model's prediction, and bound the discrepancy otherwise, thus providing *attribution completeness* guarantees.

1 INTRODUCTION

Deep learning has driven substantial advances across many tasks through its ability to learn rich representations directly from data (LeCun et al., 2015). However, these representations are not inherently aligned with human-understandable concepts, making the resulting predictions difficult to interpret (Samek et al., 2017). Explainable artificial intelligence (XAI) seeks to make model behavior and predictions understandable to humans. Concept-based XAI (C-XAI) approaches this goal through properties of inputs called *concepts*, such as textures, shapes, or object parts (Poeta et al., 2025). For example, the presence of wheels may help explain why an image is classified as a vehicle.

C-XAI encompasses methods that differ in what a concept is, how it is represented, and how it relates to predictions (Poeta et al., 2025). These different uses and representations of concepts have motivated mathematical frameworks for relating and comparing concept-based methods (Fel et al., 2023a; Li et al., 2024; Poché et al., 2025). We focus on two main problems in bridging C-XAI together: a) a common definition of a *concept*, and b) a suitably general definition of *concept extraction*. In the following, we present our framework of C-XAI.

We define a concept to be a function $c\colon \mathcal{X} \to [0,1]$, where $\mathcal{X}$ is the input space. Binary values indicate absence or presence of the concept in the input, while intermediate values allow graded descriptions. We write a model as $f = g \circ h\colon \mathcal{X} \xrightarrow{h} \mathcal{Z} \xrightarrow{g} \mathcal{Y}$, where h maps inputs to latent representations and g maps these representations to predictions. We define a *concept extractor* as a map $e\colon \mathcal{Z} \to [0,1]$, so that $c = e \circ h$ defines a concept on inputs. However, existing methods can represent a single concept through a vector or a collection of spatial responses, containing information beyond its scalar presence (Vielhaben et al., 2023; Fel et al., 2023b). We therefore separate the extracted representation from its activation by writing $e = \sigma \circ \pi$: a *probe* $\pi\colon \mathcal{Z} \to \mathcal{V}$ produces the concept representation, while an *activation*

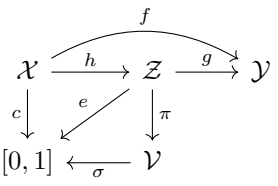

Figure 1: Concept extraction.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

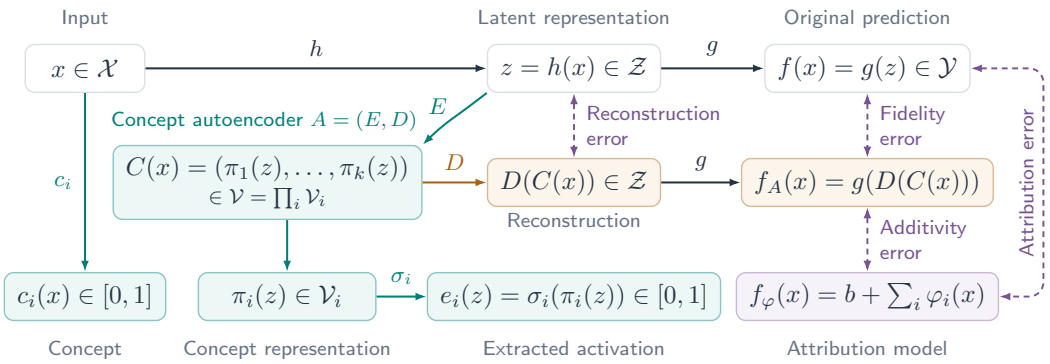

Figure 3: Overview of the framework for a scalar model output. The decomposition $f = g \circ h$ connects inputs, latent representations, and predictions. Concept probes π_i and activation functions σ_i form extractors $e_i = \sigma_i \circ \pi_i$ that estimate input-level concepts c_i. The concept autoencoder $A = (E, D)$ induces the reconstructed predictor f_A, while the attribution model f_φ adds the concept contributions to a baseline b. Solid arrows denote function evaluations; dashed links compare the indicated functions by RMSE over the input distribution. Model completeness instead considers the best recovery of f from the concept representations over prediction heads.

function $\sigma \colon \mathcal{V} \to [0, 1]$ measures its presence. This allows $\mathcal{V}$ to be multidimensional while ensuring that $c = \sigma \circ \pi \circ h$ remains a concept under our input-level definition (Fig. 1).

Unsupervised concept discovery methods learn probes $\pi_1, \dots, \pi_k$ that together form an *encoder* $E(z) = (\pi_1(z), \dots, \pi_k(z))$. The encoder maps the model's latent space $\mathcal{Z}$ to the joint concept representation space $\mathcal{V} = \mathcal{V}_1 \times \cdots \times \mathcal{V}_k = \prod_i \mathcal{V}_i$. Following earlier frameworks (Fel et al., 2023a; Poché et al., 2025), we pair it with a *decoder* $D \colon \mathcal{V} \to \mathcal{Z}$ that reconstructs the latent representation, forming a *concept autoencoder* $A = (E, D)$. The autoencoder also defines the *induced concept autoencoder model* (ICAM) $f_A = g \circ D \circ E \circ h$. Denoting $\gamma = g \circ D$ and $C = E \circ h$, we have $f_A = \gamma \circ C$ (Fig. 2).

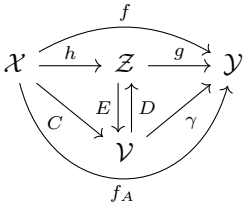

Figure 2: Concept autoenc.

We measure the *reconstruction error* $\mathrm{RE}(A)$ by the root mean squared error (RMSE) between h and $D \circ E \circ h$ over a distribution of inputs. Following ICE's comparison of reconstructed and original predictions (Zhang et al., 2021), we measure *fidelity error* $\mathrm{FE}(A)$ by the RMSE between f and f_A. We revisit *model completeness*: whether the concept representations $E(h(x))$ retain enough information to recover the model's outputs $f(x)$ (Yeh et al., 2020). We define the *model completeness error* $\mathrm{MCE}(A)$ as the infimum of the RMSE between f and $\eta \circ C$, over all eligible $\eta \colon \mathcal{V} \to \mathcal{Y}$, i.e.,

$$\mathrm{MCE}(A) = \inf_{\eta} \mathrm{RMSE}(f, \eta \circ C). \tag{1}$$

Choosing $\eta = \gamma$ gives f_A, so fidelity error bounds MCE. If g is L_g-Lipschitz continuous, a change in the latent representation produces at most a proportional change in the output. Together this makes the following chain of inequalities:

$$\mathrm{MCE}(A) \leq \mathrm{FE}(A) \leq L_g \, \mathrm{RE}(A). \tag{2}$$

For a scalar output $\mathcal{Y} = \mathbb{R}$ (e.g., a class score), an *attribution function* $\varphi = (\varphi_1, \dots, \varphi_k) \colon \mathcal{X} \to \mathbb{R}^k$ assigns each concept a score intended to describe its contribution to the prediction (Fel et al., 2023a; Poché et al., 2025). We also revisit *attribution completeness*: whether concept attributions together with a baseline sum to $f(x)$ (Sundararajan et al., 2017; Vielhaben et al., 2023). We measure the discrepancy by the *attribution error* $\mathrm{ATE}(A, \varphi)$, the RMSE between f and f_φ where $f_\varphi(x) = b + \sum_{i=1}^{k} \varphi_i(x)$ and b is a baseline value. We consider three attribution rules: occlusion and gradient-times-input, as defined by Fel et al. (2023a); Poché et al. (2025), and insertion.

The three attribution rules are defined using the concept autoencoder A. For example, occlusion measures the difference between $f_A(x)$ and the same value with the i-th concept set to zero. We

make the relationship of φ to A explicit using the triangle inequality

$$\mathrm{ATE}(A, \varphi) = \mathrm{RMSE}(f, f_\varphi) \leq \mathrm{RMSE}(f, f_A) + \mathrm{RMSE}(f_A, f_\varphi) = \mathrm{FE}(A) + \mathrm{ADD}(A, \varphi), \quad (3)$$

where $\mathrm{ADD}(A, \varphi) = \mathrm{RMSE}(f_A, f_\varphi)$ is the *additivity error*. In the case that γ is affine, for example when D is linear and g is a final linear layer, we show that all three rules are equal and $\mathrm{ADD}(A, \varphi) = 0$. This means that the attribution rules perfectly attribute the ICAM, and their error with respect to the original model can be bounded in the same way as MCE in Eq. (2). Finally, we also show a bound on the additivity error when γ is not affine using a bound on the curvature of γ. The key terms are summarized in Fig. 3.

Contributions We present a unifying mathematical framework for concept-based explanation methods. This abstraction enables general completeness guarantees for the concept representations and attributions, under explicit assumptions. Our contribution is the shared formulation and analysis that support these guarantees, rather than a new explanation method. Specifically, our contributions are threefold:

1. **A common formulation of concept-based explanation.** We distinguish concepts on inputs, their potentially multidimensional representations, and scalar activations within a shared formulation of concept testing, discovery, and bottleneck models.
2. **Reconstruction-based model completeness guarantees.** We show that fidelity error bounds model completeness error and is itself bounded by reconstruction error under a Lipschitz assumption on the prediction head. Exact reconstruction yields zero fidelity and model completeness errors.
3. **Attribution completeness through fidelity and additivity.** We bound attribution error by fidelity error plus additivity error within the ICAM. For affine decoded prediction heads, insertion, occlusion, and gradient-times-input give identical contributions with zero additivity error. For nonaffine decoded heads with a Lipschitz gradient, we bound additivity error using curvature and the magnitude of the concept representations.

2 Proposed Framework

In this section, we give the formal description of our framework.

Concept A *concept* is a function $\mathcal{X} \to [0, 1]$, where $\mathcal{X}$ is an *input space*. In the following, we assume that $\mathcal{X}$ is fixed.

Model A *model* is a function $\mathcal{X} \to \mathcal{Y}$, where $\mathcal{Y}$ is the *output space*. A *decomposition* of a model f is $(\mathcal{Z}, h, g)$ where $\mathcal{Z}$ is a *latent space*, $h\colon \mathcal{X} \to \mathcal{Z}$ is a *latent extractor*, $g\colon \mathcal{Z} \to \mathcal{Y}$ is a *prediction head*, and $f = g \circ h$. In the following, we assume the model and its decomposition are fixed.

Concept Extractor A *concept extractor* is a function $\mathcal{Z} \to [0, 1]$. A *decomposition* of a concept extractor e is $(\mathcal{V}, \pi, \sigma)$, where $\mathcal{V}$ is a *representation space*, $\pi\colon \mathcal{Z} \to \mathcal{V}$ is a *concept probe*, $\sigma\colon \mathcal{V} \to [0, 1]$ is a *concept activation function*, and $e = \sigma \circ \pi$.

Concept Autoencoder A *k-concept autoencoder* is (E, D), where $E = (\pi_1, \ldots, \pi_k)\colon \mathcal{Z} \to \mathcal{V}$ is a *k-concept encoder*, $\mathcal{V} = \mathcal{V}_1 \times \cdots \times \mathcal{V}_k$ is a *product representation space*, $\pi_i\colon \mathcal{Z} \to \mathcal{V}_i$ is the i-th *concept probe*, and $D\colon \mathcal{V} \to \mathcal{Z}$ is a *k-concept decoder*. An *induced concept autoencoder model* (ICAM) f_A of a concept autoencoder $A = (E, D)$ is $g \circ D \circ E \circ h$.

3 Overview of C-XAI

Concept-based explanations differ in how concepts enter the model, how they are obtained, and what the explanation describes (Poeta et al., 2025). Concepts may participate in prediction (*ante-hoc*) or be used to analyze a trained model (*post-hoc*). They may be specified through annotations or descriptions (*supervised*), or discovered without concept labels (*unsupervised*). In this section, we show how these methods can be described using our framework; Tab. 1 in Sec. A summarizes the instantiations.

3.1 Explainable-by-Design Models

Concept-based models use concepts directly in prediction. We distinguish their architectures independently of how the concepts are obtained or the models are trained.

A *concept bottleneck model* (CBM) is a model f paired with a decomposition $(\mathcal{Z}, h, g)$ of f where $\mathcal{Z} = [0,1]^k$ and $h = (c_1, \dots, c_k)$. Hence, each $c_i = \pi_i \circ h$ is a concept, where $\pi_i(z) = z_i$ is the i-th coordinate projection, and the prediction head $g\colon [0,1]^k \to \mathcal{Y}$ receives their activations (Fig. 4).

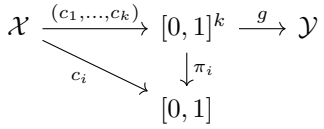

Figure 4: Bottleneck model.

A *concept embedding model* (CEM) is a model f paired with a decomposition $(\mathcal{Z}, h, g)$ where $\mathcal{Z} = \mathcal{V}_1 \times \cdots \times \mathcal{V}_k$ and k concept extractors e_i with decompositions $(\mathcal{V}_i, \pi_i, \sigma_i)$, where $\pi_i(z) = z_i$. Each probe π_i selects a concept representation $\pi_i(h(x)) \in \mathcal{V}_i$, and $\sigma_i\colon \mathcal{V}_i \to [0,1]$ maps it to a scalar activation. Thus $c_i = \sigma_i \circ \pi_i \circ h$ is a concept (Fig. 5). The prediction head g receives the representations $h(x)$ themselves. Every CBM is a CEM with $\mathcal{V}_i = [0,1]$ and identity activations.

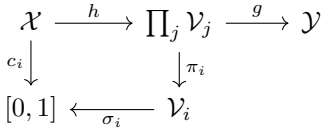

Figure 5: Embedding model.

The probability-based variants of Koh et al. (2020) and the prototype-presence constructions of ProtoTree and PIP-Net give CBM decompositions (Nauta et al., 2021; 2023). Concept logits, as used in the sequential and joint classification variants of Koh et al. (2020), illustrate scalar CEM representations. This construction also accommodates the scalar projections of Post-hoc CBMs without a residual branch and Label-free CBMs (Yuksekgonul et al., 2023; Oikarinen et al., 2023), and the similarity scores of ProtoPNet and LaBo (Chen et al., 2019; Yang et al., 2023). The vector-valued embeddings of Espinosa Zarlenga et al. (2022) illustrate multidimensional CEM representations, which can retain predictive information beyond their scalar concept activations.

Example: Supervised CBMs and CEMs We express the classification models of Koh et al. (2020) in our framework. The user specifies k binary concepts and supplies training data $\{(x_j, y_j, a_j)\}_{j=1}^n$, where $x_j \in \mathcal{X}$ is an input, $y_j \in \{1, \dots, N\}$ is its class label, and $a_j \in \{0,1\}^k$ contains its concept annotations. The entry $a_{j,i}$ indicates whether concept i is present in input x_j.

In the probability-bottleneck variant, the method learns concept predictors $\hat{c}_i\colon \mathcal{X} \to [0,1]$ and a prediction head $g\colon [0,1]^k \to \mathbb{R}^N$ producing class logits, so $\mathcal{Y} = \mathbb{R}^N$. Concept annotations encourage $\hat{c}_i(x_j) \approx a_{j,i}$, while class labels encourage $g(\hat{c}_1(x_j), \dots, \hat{c}_k(x_j))$ to predict y_j. The resulting model predicts entirely through these concept probabilities and therefore has the CBM decomposition $\big([0,1]^k, (\hat{c}_1, \dots, \hat{c}_k), g\big)$.

The other variants instead learn a concept-logit predictor $h\colon \mathcal{X} \to \mathbb{R}^k$ and a prediction head $g\colon \mathbb{R}^k \to \mathbb{R}^N$ that receives the logits directly. This gives the model decomposition $(\mathbb{R}^k, h, g)$. The architecture uses the fixed sigmoid $s(t) = (1 + \exp(-t))^{-1}$ to obtain concept probabilities from the logits. In our notation, $\mathcal{V}_i = \mathbb{R}$, the probe $\pi_i(z) = z_i$ selects the representation of concept i, and $\sigma_i = s$ gives its activation. Concept supervision encourages $e_i(h(x_j)) = s(h(x_j)_i) \approx a_{j,i}$, while the prediction head receives $h(x_j)$ itself. The decomposition together with these extractors is therefore a CEM in our terminology.

3.2 Post-hoc Concept-Based Explanations

Post-hoc methods explain a fixed model f with decomposition $(\mathcal{Z}, h, g)$ by identifying concepts in its latent representations and examining their relation to predictions.

3.2.1 Concept Detection

Concept detection asks whether a trained model's latent representations $h(x)$ retain information about a concept of interest. For a user-specified concept $c\colon \mathcal{X} \to [0,1]$, we seek an extractor $e = \sigma \circ \pi$ such that $e(h(x)) \approx c(x)$. Annotated examples allow this correspondence to be learned and evaluated.

The most direct approach associates concepts with existing model units. For $\mathcal{Z} = \prod_i \mathcal{V}_i$, projection probes $\pi_i(z) = z_i$ select neurons, spatial feature maps, or groups of units, whose concept responses are described by σ_i. This gives a CEM interpretation of the existing model decomposition. Network Dissection evaluates channel–concept associations against annotated spatial masks (Bau et al., 2017). CRP similarly treats channels as concept representations (Achtibat et al., 2023).

More generally, learned probes $\pi_i \colon \mathcal{Z} \to \mathcal{V}_i$ can use the full latent representation. We can compare learned probes to study relationships between concepts, as in Net2Vec (Fong & Vedaldi, 2018), or evaluate class-score sensitivity along concept directions, as in TCAV (Kim et al., 2018). CAR extends detection to nonlinear probes and studies concept–class associations and input features supporting concept detection (Crabbé & Schaar, 2022).

Example: TCAV The user supplies a trained classifier $f \colon \mathcal{X} \to \mathbb{R}^N$ and selects a layer giving the decomposition $(\mathbb{R}^d, h, g)$. They specify a binary concept c through labeled examples $\{(x_j, a_j)\}_{j=1}^n$, where $a_j = c(x_j) \in \{0, 1\}$. In our framework, TCAV (Kim et al., 2018) uses an affine probe $\pi(z) = \langle w, z \rangle + \beta$ with representation space $\mathcal{V} = \mathbb{R}$. Taking $\sigma(t) = \mathbf{1}\{t > 0\}$ gives the concept extractor $e = \sigma \circ \pi$. The method learns w and β so that $e \circ h$ is as close as possible to c on the labeled examples.

3.2.2 Unsupervised Concept Discovery

Unsupervised concept discovery constructs concept representations without annotations specifying their meanings. We use our framework to describe methods that admit reconstruction, followed by a concrete example using ICE. For a fixed model decomposition $(\mathcal{Z}, h, g)$, these methods construct probes $E = (\pi_1, \ldots, \pi_k)$ and a decoder D, forming a concept autoencoder $A = (E, D)$ (Fel et al., 2023a; Poché et al., 2025). Many methods fit these functions by minimizing reconstruction error together with regularization or constraints (Fel et al., 2023a; Bhusal et al., 2025). Other constructions provide exact reconstruction algebraically (Vielhaben et al., 2023). The discovered representations are then interpreted through representative images, spatial responses, or concept-specific saliency maps, allowing users to identify and name the concepts (Zhang et al., 2021; Fel et al., 2023b).

The decoder also identifies the predictor whose concept contributions are being explained. Writing $C = E \circ h$ and $\gamma = g \circ D$, the reconstructed predictor is the ICAM $f_A = \gamma \circ C$. When activation functions σ_i are specified, this gives the CEM decomposition $(\mathcal{V}, C, \gamma)$, with input-level concepts $c_i = \sigma_i \circ \pi_i \circ h$. Standard feature-attribution methods can therefore treat each concept representation as a feature block of this predictor. For example, insertion and occlusion modify blocks of $C(x)$ before evaluating γ, while gradient-times-input combines these blocks with derivatives of γ (Fel et al., 2023a; Poché et al., 2025). These attributions describe the reconstructed prediction $f_A(x)$, which may differ from the original prediction $f(x)$. The next section uses this distinction to study how well the concept representations and their attributions recover the original model's outputs.

Example: ICE We express ICE's NMF construction in our framework (Zhang et al., 2021) (see also Fig. 6 in Sec. A). The user supplies a trained classifier with N classes, selects a class $r \in \{1, \ldots, N\}$, provides inputs $X = (x_1, \ldots, x_n) \in \mathcal{X}^n$ from that class, and chooses the number of concepts k. We explain the class logit $f \colon \mathcal{X} \to \mathbb{R}$ using the nonnegative feature maps preceding global average pooling and the final affine classifier. This gives the decomposition $(\mathcal{Z}, h, g)$ with $\mathcal{Z} = \mathbb{R}_{\geq 0}^{S \times d}$, where S is the number of spatial locations and d is the number of feature channels. The prediction head is $g(z) = b + \frac{1}{S} \sum_{s=1}^{S} \langle w, z_{s,:} \rangle$, where $w \in \mathbb{R}^d$ and $b \in \mathbb{R}$ are the trained classifier's parameters for class r. The original model remains fixed.

ICE stacks the supplied feature maps into $Z \in \mathbb{R}_{\geq 0}^{nS \times d}$ and fits a dictionary $U \in \mathbb{R}_{\geq 0}^{k \times d}$ and coefficients $H \in \mathbb{R}_{\geq 0}^{k \times nS}$ by minimizing $\|Z - H^\top U\|_F^2$. Each row of U is a learned concept direction. Once U is fixed, a new input receives one coefficient per concept and spatial location. We therefore take $\mathcal{V}_i = \mathbb{R}_{\geq 0}^S$ and identify $\mathcal{V} = \prod_i \mathcal{V}_i$ with $\mathbb{R}_{\geq 0}^{k \times S}$ by placing concept representations in rows. ICE's reconstruction operation is our decoder $D(V) = V^\top U$. Its numerical procedure for fitting coefficients with U fixed gives our encoder $E(z)$, seeking to minimize $\|z - D(V)\|_F^2$ over $V \in \mathcal{V}$. The probes are $\pi_i(z) = E(z)_{i,:}$, so each concept representation is an entire spatial coefficient map. ICE uses these maps to highlight image regions and their spatial means to select representative images for interpreting the concepts.

For an input x, ICE multiplies the spatial mean of $\pi_i(h(x))$ by the class-specific weight $(Uw)_i$ to obtain concept i's contribution. Its explanation displays these quantities as the concept's similarity score, weight, and contribution. Our general results in Sec. 4.3 explain why these contributions, together with the classifier bias, recover the reconstructed logit and how reconstruction quality controls their agreement with the original prediction.

4 Completeness Guarantees

We connect reconstruction quality to fidelity, model completeness, and attribution completeness. The results apply to the fitted encoder and decoder, independently of their learning objective or optimization procedure.

Common Assumptions and Notation Fix a model f with decomposition $(\mathcal{Z}, h, g)$, a concept autoencoder $A = (E, D)$, and a probability distribution p on $\mathcal{X}$. Write $C = E \circ h$ and $\gamma = g \circ D$, thus $f_A = \gamma \circ C$. Assume that $\mathcal{Y} = \mathbb{R}^N$, $\mathcal{Z} \subseteq \mathbb{R}^d$, that all relevant functions are measurable, and that the errors being bounded are finite. For functions $f_1, f_2 \colon \mathcal{X} \to \mathbb{R}^m$, define

$$\mathrm{RMSE}(f_1, f_2) = \sqrt{\mathbb{E}_{x\sim p}\big[\|f_1(x) - f_2(x)\|_2^2\big]}. \tag{4}$$

All errors use this distribution; choosing p uniform on a dataset gives empirical guarantees on that dataset. Further assumptions are introduced where needed.

We use RMSE because it aggregates discrepancies across inputs while retaining the scale of the quantities being compared. It satisfies $\mathrm{RMSE}(f_1, f_2) = \|f_1 - f_2\|_{L^2(p)}$, where the $L^2(p)$ norm is defined on square-integrable functions identified up to equality p-almost surely. We can therefore use properties of this norm, including the triangle inequality. The Euclidean norms measure discrepancies between individual representations in $\mathcal{Z}$ or outputs in $\mathcal{Y}$, while the $L^2(p)$ norm turns these pointwise discrepancies into distances between functions. More general norms could be used with corresponding regularity assumptions; we adopt this setting to keep the statements and proofs easy to follow.

4.1 Fidelity

The *fidelity error* measures output disagreement between f and f_A. Thus we define $\mathrm{FE}(A) = \mathrm{RMSE}(f, f_A)$. The *reconstruction error* measures the disagreement between z and $D(E(z))$ where $z = h(x)$, so we define $\mathrm{RE}(A) = \mathrm{RMSE}(h, D \circ C)$.

Now observe that if $h(x) = D(E(h(x)))$, then

$$f(x) = g(h(x)) = g(D(E(h(x)))) = f_A(x). \tag{5}$$

From this, we can easily see that if $\mathrm{RE}(A) = 0$, then $\mathrm{FE}(A) = 0$ as well. For approximate reconstruction, assume that g is L_g-Lipschitz for a finite constant $L_g \geq 0$:

$$\|g(z) - g(z')\|_2 \leq L_g \|z - z'\|_2 \qquad \text{for all } z, z' \in \mathcal{Z}. \tag{6}$$

This controls how strongly the prediction head can amplify reconstruction errors.

Theorem 1 (Reconstruction controls fidelity)**.** *Under the Lipschitz assumption in Eq. (6),*

$$\mathrm{FE}(A) \leq L_g \, \mathrm{RE}(A). \tag{7}$$

For a proof see Sec. B.1.

This result reflects the construction of the ICAM: f_A replaces $h(x)$ by its reconstruction $D(E(h(x)))$ before applying the same prediction head g. The usefulness of this dwells in the formalization, as we show this is a stepping stone to more interesting results. The Lipschitz assumption holds for many common neural-network heads, including finite compositions of affine maps and Lipschitz activations such as ReLU. For deeper compositions, products of the individual layer bounds give a valid but potentially conservative constant.

4.2 Model Completeness

Model completeness asks how well the original outputs can be recovered from $C(x)$ when the prediction head is allowed to vary. Let then Γ contain all functions $\mathcal{V} \to \mathbb{R}^N$. We define the *model completeness error* as $\mathrm{MCE}(A) = \inf_{\eta \in \Gamma} \mathrm{RMSE}(f, \eta \circ C)$.

Now, observe that $g \circ D \in \Gamma$ is an admissible head. The definition of the infimum therefore gives $\mathrm{MCE}(A) \leq \mathrm{RMSE}(f, g \circ D \circ C) = \mathrm{FE}(A)$. Combining this with Theorem 1, we get the following result.

Theorem 2 (Reconstruction controls completeness)**.** *Under the Lipschitz assumption in Eq. (6),*

$$\mathrm{MCE}(A) \leq \mathrm{FE}(A) \leq L_g \, \mathrm{RE}(A). \tag{8}$$

The connection is not immediately obvious, as completeness concerns only the concept representations $E(h(x))$, while reconstruction concerns both E and D. In particular, if $\mathrm{RE}(A) = 0$, then the concept representations are complete. In this view, the reason is clear: E retains enough information for D to perfectly reconstruct the latent representation almost surely, and thus recover the original output. This exact-reconstruction conclusion does not require the Lipschitz assumption. The theorem also gives an additional interpretation to minimizing reconstruction error when learning A: under the Lipschitz assumption, it lowers an upper bound on the incompleteness of the concept representations.

4.3 Attribution Completeness

Take scalar outputs, $N = 1$, and assume that each $\mathcal{V}_i \subseteq \mathbb{R}^{m_i}$ contains its ambient zero vector 0_i, where $m_i \geq 1$. Equip $\mathcal{W} = \prod_i \mathbb{R}^{m_i}$ with its Euclidean inner product and norm. Write $b = \gamma(0_{\mathcal{V}})$, where $0_{\mathcal{V}} = (0_1, \ldots, 0_k)$. The isolation map $P_i(v_1, \ldots, v_k) = (0_1, \ldots, 0_{i-1}, v_i, 0_{i+1}, \ldots, 0_k)$ retains concept i; both $P_i v$ and the representation $v - P_i v$ with that concept removed belong to $\mathcal{V}$.

An *attribution* is a function $\varphi = (\varphi_1, \ldots, \varphi_k) \colon \mathcal{X} \to \mathbb{R}^k$ that assigns each concept a contribution to a prediction. We study three choices:

$$\varphi_i(x) = \gamma(P_i C(x)) - b \qquad \text{(Insertion)}, \tag{9}$$

$$\varphi_i(x) = \gamma(C(x)) - \gamma(C(x) - P_i C(x)) \qquad \text{(Occlusion)}, \tag{10}$$

$$\varphi_i(x) = \langle \nabla \gamma(C(x)), P_i C(x) \rangle \qquad \text{(Gradient-times-input)}. \tag{11}$$

Insertion adds a concept to the baseline, occlusion removes it from the full representation, and gradient-times-input pairs its coordinates with their output sensitivities. For the last rule, γ must also be defined and differentiable on an open neighborhood of each $C(x)$ in $\mathcal{W}$, agreeing with $g \circ D$ on its intersection with $\mathcal{V}$.

An attribution function φ induces an *attribution model* $f_\varphi \colon \mathcal{X} \to \mathbb{R}$, defined by

$$f_\varphi(x) = b + \sum_{i=1}^{k} \varphi_i(x). \tag{12}$$

The *attribution error* measures output disagreement between f and f_φ, so we define $\mathrm{ATE}(A, \varphi) = \mathrm{RMSE}(f, f_\varphi)$. Similarly, the *additivity error* measures output disagreement between f_A and f_φ, so we define $\mathrm{ADD}(A, \varphi) = \mathrm{RMSE}(f_A, f_\varphi)$.

Observe that attributions computed through $\gamma = g \circ D$ concern the ICAM f_A, whose predictions may differ from those of the original model f. Our attribution model f_φ makes this distinction explicit: its agreement with f_A is measured by additivity error, while its agreement with f is measured by attribution error. Applying the triangle inequality for RMSE to f, f_A, and f_φ gives

$$\mathrm{ATE}(A, \varphi) \leq \mathrm{FE}(A) + \mathrm{ADD}(A, \varphi). \tag{13}$$

If $\mathrm{ADD}(A, \varphi) = 0$, then $f_\varphi = f_A$ almost surely, and hence $\mathrm{ATE}(A, \varphi) = \mathrm{FE}(A)$. Thus, even attributions that perfectly recover the ICAM output retain its disagreement with the original model. This lets us study additivity within the ICAM separately and use fidelity to connect the resulting guarantees to the original predictions.

In the following proposition, we provide a general case where $\mathrm{ADD}(A, \varphi) = 0$.

Proposition 1 (Complete attributions for affine decoded heads)**.** *Suppose that γ is affine, i.e., there are $a_i \in \mathbb{R}^{m_i}$ such that $\gamma(v_1, \ldots, v_k) = b + \sum_{i=1}^{k} \langle a_i, v_i \rangle$ for all $v \in \mathcal{V}$. Then the attribution $\varphi_i(x) = \langle a_i, \pi_i(h(x)) \rangle$ has zero additivity error, i.e.,* $\mathrm{ADD}(A, \varphi) = 0$.

For a proof see Sec. B.2. Insertion, occlusion, and gradient-times-input all give precisely this attribution for an affine decoded head. We verify this in Sec. B.3. Thus, all three rules have zero additivity error, and their attribution error equals fidelity error.

This is an important case because many concept discovery methods use affine decoders. If g is also affine, as when explaining a class logit through the final affine layer, then $\gamma = g \circ D$ is affine even when the encoder is nonlinear. Theorem 1 therefore gives

$$\mathrm{ATE}(A, \varphi) = \mathrm{FE}(A) \leq L_g \, \mathrm{RE}(A) \tag{14}$$

for all three rules. For a scalar head $g(z) = \langle w, z \rangle + \beta$, we may take $L_g = \|w\|_2$, so this bound can be evaluated directly from the classifier weights and reconstruction error. Small reconstruction error together with a small Lipschitz constant therefore guarantees that the attributions approximately recover the original model output, and exact reconstruction makes them complete. This conclusion applies to the affine class scores; applying softmax generally makes the decoded head nonaffine.

Bounding the Nonaffinity of γ When explaining an earlier layer, the remaining prediction head g may be nonlinear, and a nonlinear decoder can also make $\gamma = g \circ D$ nonaffine. The three attribution rules may then give different contributions with nonzero additivity error. To extend the affine case, we control how far γ departs from an affine approximation by bounding the variation of its gradient.

Assume that γ is also defined and differentiable on an open set $\Omega \subseteq \mathcal{W}$ containing the segments from zero to $C(x)$ and $P_i C(x)$, and from $C(x)$ to $C(x) - P_i C(x)$, for every x and i. Its values on $\Omega \cap \mathcal{V}$ agree with $g \circ D$. Assume that $\mathbb{E}_{x \sim p}\left[\|C(x)\|_2^4\right] < \infty$ and that the gradient of γ is M-Lipschitz for a finite $M \geq 0$:

$$\|\nabla\gamma(v) - \nabla\gamma(w)\|_2 \leq M \|v - w\|_2 \qquad \text{for all } v, w \in \Omega. \tag{15}$$

Theorem 3 (Curvature bounds additivity error)**.** *Under these assumptions, insertion, occlusion, and gradient-times-input satisfy*

$$\mathrm{ADD}(A, \varphi) \leq M \sqrt{\mathbb{E}_{x \sim p}\left[\|C(x)\|_2^4\right]}. \tag{16}$$

A proof is in Sec. B.4.

4.4 Empirical Tightness

To assess how conservative these bounds are, we evaluate them on a ResNet-50 trained on ImageNet-1k, split after its last residual stage and before its final affine layer, where the prediction head is affine and L_g is available in closed form. We fit concept autoencoders with PCA, NMF, k-means, a sparse autoencoder, and an autoencoder with a nonlinear decoder, for $k \in \{5, 25, 50\}$ concepts and three seeds; details and all results are in Sec. C. All bounds hold in every configuration, but their tightness differs considerably.

The fidelity bound of Theorem 1 is conservative: the fidelity error is about a quarter of $L_g \, \mathrm{RE}(A)$ before the final layer and about a tenth after the last residual stage, almost independently of the method and of k. The slack has two sources. First, reconstruction errors lie mostly in directions to which the prediction head is insensitive, whereas L_g accounts for its most sensitive direction. Second, after the last residual stage, the errors at different spatial locations partly cancel when the head averages over them.

For model completeness, heads fitted to the concept representations reduce the error of the decoded head $g \circ D$ by at most about 9%, so $g \circ D$ is already close to the best head we could find.

For attributions, the tightness depends on the decoder. With the four affine decoders, insertion, occlusion, and gradient-times-input are additive up to floating-point error, so the attribution error equals the fidelity error and the attribution bound is attained, as Proposition 1 predicts. With the nonlinear decoder, the attribution bound, which adds the curvature term of Theorem 3 to the fidelity error, still holds but exceeds the attribution error by two to three orders of magnitude.

5 Related Work

Concepts from Dictionary Learning Fel et al. (2023a) unify k-means, PCA, NMF, and sparse autoencoders through dictionary learning, abstracting methods such as ACE (Ghorbani et al., 2019), ICE (Zhang et al., 2021), and CRAFT (Fel et al., 2023b). The same encoder–decoder formulation also accommodates FACE (Bhusal et al., 2025). Their coefficient inference is our encoder E, their dictionary defines $D(v) = Bv$, and their reconstructed predictor is our ICAM. Allowing a decoder bias also accommodates Anthropic's sparse autoencoders (Bricken et al., 2023; Templeton et al., 2024). Our guarantees apply to these fitted constructions under the stated assumptions, independently of the learning objective or whether optimization reaches a global optimum. For affine g, as in the holistic framework's penultimate-layer analysis, $\gamma = g \circ D$ is affine, so our results apply to insertion and to the occlusion and gradient-times-input rules they consider. Our analysis complements their optimality guarantees for importance rankings and FACE's bounds on predictive disagreement by connecting reconstruction to model completeness and distinguishing attribution completeness from fidelity. The formulation also permits nonlinear decoders, with approximate guarantees under the corresponding Lipschitz and curvature assumptions.

Multidimensional Concept Discovery MCD (Vielhaben et al., 2023) represents concepts by subspaces of feature channels. Its component projections are our probes, and its decoder sums the components: $D(v_1, \ldots, v_k) = \sum_i v_i$. For the complete basis decomposition, including the residual, $D \circ E = \mathrm{id}_{\mathcal{Z}}$. Together with its affine prediction head and blockwise relevance scores, this makes MCD's attribution-completeness relation a special case of exact reconstruction and Proposition 1. Our formulation identifies the structural conditions behind this result and additionally establishes model completeness of the full decomposition for any prediction head. Our approximate bounds address discarded components and nonlinear heads under the corresponding regularity assumptions. These distribution-dependent errors complement MCD's geometric completeness score based on classifier weights. Our product representation spaces also accommodate HU-MCD (Grobrügge et al., 2025) and the vector-valued concept blocks of subspace-aware sparse autoencoders (Dalili & Mahdavi, 2026), without requiring the same discovery procedure or exact reconstruction.

Completeness and Evaluation Yeh et al. (2020) measure completeness through normalized classification accuracy against ground-truth labels, optimizing a decoder for thresholded, normalized concept projections while retaining the original prediction head. Our model completeness error instead measures recovery of the original outputs $f(x)$ over all measurable heads on $C(x)$, separating information retained by the encoder from the performance of a particular decoder. Their ConceptSHAP distributes a global completeness score among concepts, whereas our attribution error concerns recovery of individual model outputs. Other evaluations address semantic readability and perturbation-based faithfulness (Li et al., 2024), or whether explanations enable an LLM to simulate model predictions, as in ConSim (Poché et al., 2025). SAEBench further shows that improvements in sparse-autoencoder proxy metrics need not improve practical interpretability evaluations (Karvonen et al., 2025). Our contribution is a common formulation and conditional guarantees for existing constructions; we do not introduce a discovery algorithm, and our guarantees do not establish semantic readability or successful simulation.

6 Conclusion

We introduced a common language for concept-based prediction, detection, and discovery, distinguishing input-level concepts, potentially multidimensional representations, and scalar activations. For concept autoencoders, exact reconstruction gives zero fidelity and model completeness errors, while a Lipschitz prediction head provides quantitative guarantees for approximate reconstruction. We also bound attribution error by fidelity error plus additivity error within the ICAM. Insertion, occlusion, and gradient-times-input have zero additivity error for affine decoded heads, and our curvature bound extends the analysis to smooth nonaffine heads.

These guarantees apply to existing methods through their encoder–decoder structure, independently of their training objectives. They concern output recovery and attribution completeness; the semantic meaning of concepts requires separate evaluation.

AI USE STATEMENT

We used generative AI tools to assist with developing and refining the conceptual framework, mathematical statements, and proofs. We used a generative AI coding assistant to help implement, debug, and optimize the evaluation code. We also used generative AI to support literature exploration, assist with analyzing experimental results, prepare figures, and draft and revise the structure, language, and formatting of the paper.

The authors reviewed the AI-assisted mathematical arguments, figures, code, and text. The code was tested on small configurations, and its outputs were checked against theoretical predictions. AI-assisted text was read and revised by the authors, and every reported number was checked against the logged results. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

All assumptions of our theoretical results are stated explicitly in Sec. 4 and the proofs are either in the main text or in Sec. B. For the empirical evaluation, Sec. C describes the model, data, boundaries, concept discovery methods, prediction heads, and the estimation of M, and reports all results as means over three seeds together with their standard deviations. The evaluation uses only public resources: the torchvision ResNet-50 weights (`IMAGENET1K_V2`) and the ImageNet-1k validation set. We provide the source code as supplementary material; each run is fully specified by a configuration file and a seed, which determines the sampled images, the split into fitting and evaluation halves, and the initialization of all methods and heads.

REFERENCES

Reduan Achtibat, Maximilian Dreyer, Ilona Eisenbraun, Sebastian Bosse, Thomas Wiegand, Wojciech Samek, and Sebastian Lapuschkin. From attribution maps to human-understandable explanations through Concept Relevance Propagation. *Nature Machine Intelligence*, 5(9):1006–1019, September 2023. ISSN 2522-5839. doi: 10.1038/s42256-023-00711-8. URL https://www.nature.com/articles/s42256-023-00711-8. Publisher: Nature Publishing Group.

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*, 2017. URL https://arxiv.org/abs/1704.05796.

Dipkamal Bhusal, Michael Clifford, Sara Rampazzi, and Nidhi Rastogi. FACE: Faithful automatic concept extraction. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/3e59c3d09840b4e40a21a1f2ea771f89-Abstract-Conference.html.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. https://transformer-circuits.pub/2023/monosemantic-features/index.html.

Chaofan Chen, Oscar Li, Daniel Tao, Alina Barnett, Cynthia Rudin, and Jonathan K Su. This looks like that: Deep learning for interpretable image recognition. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/adf7ee2dcf142b0e11888e72b43fcb75-Abstract.html.

Jonathan Crabbé and Mihaela Van Der Schaar. Concept activation regions: A generalized framework for concept-based explanations. In *Advances in Neural Information Processing Systems 35*, pp. 2590–2607. Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2022. ISBN

978-1-7138-7108-8. doi: 10.52202/068431-0188. URL http://www.proceedings.com/068431-0188.html.

Seyed Arshan Dalili and Mehrdad Mahdavi. Subspace-aware sparse autoencoders for effective mechanistic interpretability, 2026. URL https://arxiv.org/abs/2606.06333. Preprint.

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

Thomas Fel, Victor Boutin, Louis Béthune, Remi Cadene, Mazda Moayeri, Léo Andéol, Mathieu Chalvidal, and Thomas Serre. A holistic approach to unifying automatic concept extraction and concept importance estimation. In *Advances in Neural Information Processing Systems*, volume 36, 2023a. URL https://papers.neurips.cc/paper_files/paper/2023/hash/abf3682c9cf9245a0294a4bebe4544ff-Abstract-Conference.html.

Thomas Fel, Agustin Picard, Louis Béthune, Thibaut Boissin, David Vigouroux, Julien Colin, Rémi Cadène, and Thomas Serre. CRAFT: Concept recursive activation FacTorization for explainability. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023b. URL https://arxiv.org/abs/2211.10154.

Ruth Fong and Andrea Vedaldi. Net2Vec: Quantifying and explaining how concepts are encoded by filters in deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. URL https://arxiv.org/abs/1801.03454.

Amirata Ghorbani, James Wexler, James Y Zou, and Been Kim. Towards automatic concept-based explanations. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/77d2afcb31f6493e350fca61764efb9a-Abstract.html.

Arne Grobrügge, Niklas Kühl, Gerhard Satzger, and Philipp Spitzer. Towards human-understandable multi-dimensional concept discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. URL https://arxiv.org/abs/2503.18629.

Adam Karvonen, Can Rager, Johnny Lin, Curt Tigges, Joseph Isaac Bloom, David Chanin, Yeu-Tong Lau, Eoin Farrell, Callum Stuart Mcdougall, Kola Ayonrinde, Demian Till, Matthew Wearden, Arthur Conmy, Samuel Marks, and Neel Nanda. SAEBench: A comprehensive benchmark for sparse autoencoders in language model interpretability. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 29223–29264. PMLR, 2025. URL https://proceedings.mlr.press/v267/karvonen25a.html.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viégas, and Rory Sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV). In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2668–2677. PMLR, 2018. URL https://proceedings.mlr.press/v80/kim18d.html.

Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5338–5348. PMLR, 2020. URL https://proceedings.mlr.press/v119/koh20a.html.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015. doi: 10.1038/nature14539. URL https://www.nature.com/articles/nature14539.

Meng Li, Haoran Jin, Ruixuan Huang, Zhihao Xu, Defu Lian, Zijia Lin, Di Zhang, and Xiting Wang. Evaluating readability and faithfulness of concept-based explanations. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 607–625, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.36. URL https://aclanthology.org/2024.emnlp-main.36/.

Meike Nauta, Ron van Bree, and Christin Seifert. Neural prototype trees for interpretable fine-grained image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. URL https://arxiv.org/abs/2012.02046.

Meike Nauta, Jörg Schlötterer, Maurice van Keulen, and Christin Seifert. PIP-Net: Patch-based intuitive prototypes for interpretable image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2744–2753, 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Nauta_PIP-Net_Patch-Based_Intuitive_Prototypes_for_Interpretable_Image_Classification_CVPR_2023_paper.html.

Tuomas Oikarinen, Subhro Das, Lam M. Nguyen, and Tsui-Wei Weng. Label-free concept bottleneck models. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2304.06129.

Antonin Poché, Alon Jacovi, Agustin Martin Picard, Victor Boutin, and Fanny Jourdan. ConSim: Measuring concept-based explanations' effectiveness with automated simulatability. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5594–5615, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.279. URL https://aclanthology.org/2025.acl-long.279/.

Eleonora Poeta, Gabriele Ciravegna, Eliana Pastor, Tania Cerquitelli, and Elena Baralis. Concept-based explainable artificial intelligence: A survey. *ACM Comput. Surv.*, November 2025. ISSN 0360-0300. doi: 10.1145/3774643. URL https://doi.org/10.1145/3774643. Just Accepted.

Wojciech Samek, Thomas Wiegand, and Klaus-Robert Müller. Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. *arXiv preprint arXiv:1708.08296*, 2017. URL https://arxiv.org/abs/1708.08296.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3319–3328. PMLR, 2017. URL https://proceedings.mlr.press/v70/sundararajan17a.html.

Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, Hoagy Cunningham, Nicholas L Turner, Callum McDougall, Monte MacDiarmid, C. Daniel Freeman, Theodore R. Sumers, Edward Rees, Joshua Batson, Adam Jermyn, Shan Carter, Chris Olah, and Tom Henighan. Scaling monosemanticity: Extracting interpretable features from Claude 3 Sonnet. *Transformer Circuits Thread*, 2024. URL https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html.

Johanna Vielhaben, Stefan Blücher, and Nils Strodthoff. Multi-dimensional concept discovery (MCD): A unifying framework with completeness guarantees. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=KxBQPz7HKh.

Yue Yang, Artemis Panagopoulou, Shenghao Zhou, Daniel Jin, Chris Callison-Burch, and Mark Yatskar. Language in a bottle: Language model guided concept bottlenecks for interpretable image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19187–19197, 2023. URL https://arxiv.org/abs/2211.11158.

Chih-Kuan Yeh, Been Kim, Sercan Ö. Arık, Chun-Liang Li, Tomas Pfister, and Pradeep Ravikumar. On completeness-aware concept-based explanations in deep neural networks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 20554–20565, 2020. URL https://papers.nips.cc/paper_files/paper/2020/file/ecb287ff763c169694f682af52c1f309-Paper.pdf.

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.

Ruihan Zhang, Prashan Madumal, Tim Miller, Krista A. Ehinger, and Benjamin I. P. Rubinstein. Invertible concept-based explanations for CNN models with non-negative concept activation vectors. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 11682–11690, 2021. doi: 10.1609/aaai.v35i13.17389. URL https://ojs.aaai.org/index.php/AAAI/article/view/17389.

A Instantiations of Concept-Based Methods

Tab. 1 summarizes how the methods discussed in Sec. 3 and 5 fit our framework. For explainable-by-design models, the latent space is the representation space, $\mathcal{Z} = \mathcal{V}$, and the probes $\pi_i(z) = z_i$ form the identity encoder. The trivial concept autoencoder $E = D = \mathrm{id}$ then gives $f_A = f$ and $\gamma = g$, so $\mathrm{MCE}(A) = \mathrm{FE}(A) = 0$, and attribution completeness depends only on whether g is affine. Concept detection methods learn probes for individual user-specified concepts without a decoder, so the guarantees of Sec. 4 do not apply to them directly. All concept discovery methods in the table use affine decoders, so γ is affine whenever g is.

Table 1: Instantiations of concept-based methods in our framework: CBMs (Koh et al., 2020), PCBM (Yuksekgonul et al., 2023), LF-CBM (Oikarinen et al., 2023), CEM (Espinosa Zarlenga et al., 2022), TCAV (Kim et al., 2018), CAR (Crabbé & Schaar, 2022), ACE (Ghorbani et al., 2019), CRAFT (Fel et al., 2023b), ICE (Zhang et al., 2021), SAE (Bricken et al., 2023), and MCD (Vielhaben et al., 2023). Columns give the latent space $\mathcal{Z}$, the concept representation space $\mathcal{V}_i$, the probe π_i, the activation function σ_i, the decoder D, and whether the decoded head $\gamma = g \circ D$ is affine; "if g" means that γ is affine whenever g is. A dash marks a component the method does not specify. s is the logistic sigmoid, U is a dictionary with concept directions u_i in its rows, NMF encoders compute nonnegative least-squares coefficients with U fixed, and Q_i projects channels onto the i-th MCD subspace, with the residual subspace included as a concept.

Method	$\mathcal{Z}$	$\mathcal{V}_i$	$\pi_i(z)$	σ_i	D	γ affine
Explainable by design						
CBM, probabilities	$[0,1]^k$	$[0,1]$	z_i	id	id	if g
CBM, logits	$\mathbb{R}^k$	$\mathbb{R}$	z_i	s	id	if g
PCBM, LF-CBM	$\mathbb{R}^k$	$\mathbb{R}$	z_i	–	id	yes
CEM	$\prod_i \mathbb{R}^{2m}$	$\mathbb{R}^{2m}$	z_i	$s(\langle w, v\rangle + \beta)$	id	no
Concept detection						
TCAV	$\mathbb{R}^d$	$\mathbb{R}$	$\langle w, z\rangle + \beta$	$\mathbf{1}\{t > 0\}$	–	–
CAR	$\mathbb{R}^d$	$\mathbb{R}$	kernel SVM	$\mathbf{1}\{t > 0\}$	–	–
Concept discovery						
ACE	$\mathbb{R}^d$	$\{0,1\}$	$\mathbf{1}\{i = \arg\min_j \lVert z - u_j\rVert\}$	–	$U^\top v$	if g
CRAFT	$\mathbb{R}^d_{\geq 0}$	$\mathbb{R}_{\geq 0}$	$E(z)_i$ (NMF)	–	$U^\top v$	if g
ICE	$\mathbb{R}^{S\times d}_{\geq 0}$	$\mathbb{R}^S_{\geq 0}$	$E(z)_{i,:}$ (NMF)	–	$V^\top U$	yes
SAE	$\mathbb{R}^d$	$\mathbb{R}_{\geq 0}$	$\mathrm{ReLU}(\langle w_i, z\rangle + \beta_i)$	–	$U^\top v + \beta_D$	if g
MCD	$\mathbb{R}^{S\times d}$	$\mathbb{R}^{S\times d}$	$zQ_i^\top$	–	$\sum_i v_i$	if g

For CEM, we take as the representation of concept i the pair of positive and negative embeddings $(\hat{c}_i^+, \hat{c}_i^-) \in \mathbb{R}^{2m}$, the shared scoring function as σ_i, and include the probability-weighted mixing of the two embeddings in g; this mixing makes g nonaffine. Following Fel et al. (2023a), we read ACE's clustering of segment activations as k-means dictionary learning with one-hot coefficients.

ICE Fig. 6 summarizes the ICE example of Sec. 3.2.2. Since D and g are both affine, so is the decoded head,

$$\gamma(V) = g(V^\top U) = b + \frac{1}{S}\sum_{s=1}^{S}\sum_{i=1}^{k} V_{i,s}\langle u_i, w\rangle = b + \sum_{i=1}^{k}(Uw)_i\,\bar{v}_i, \qquad \bar{v}_i = \frac{1}{S}\sum_{s=1}^{S} V_{i,s}, \tag{17}$$

where u_i is the i-th row of U. This is the affine form of Proposition 1 with $a_i = S^{-1}(Uw)_i\mathbf{1}_S$. Insertion, occlusion, and gradient-times-input therefore all return ICE's contributions $\varphi_i(x) = (Uw)_i\,\bar{v}_i$, which sum with b to $f_A(x)$, so $\mathrm{ATE}(A,\varphi) = \mathrm{FE}(A)$. The head g is Lipschitz with $L_g = \|w\|_2/\sqrt{S}$, and Theorem 1 gives $\mathrm{FE}(A) \le \|w\|_2\,\mathrm{RE}(A)/\sqrt{S}$.

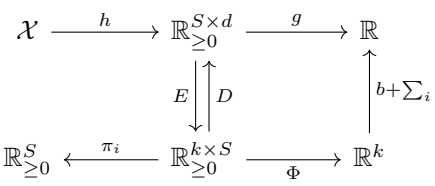

Figure 6: ICE (Zhang et al., 2021) as a concept autoencoder. The encoder E computes nonnegative coefficients $V = E(z)$ for the fixed dictionary $U \in \mathbb{R}^{k\times d}_{\ge 0}$, and the decoder is $D(V) = V^\top U$. The probe π_i returns the spatial coefficient map of concept i, which ICE uses to highlight image regions. The prediction head is the class logit $g(z) = b + \frac{1}{S}\sum_s \langle w, z_{s,:}\rangle$, and $\Phi(V) = \big((Uw)_i\,\bar{v}_i\big)_{i=1}^{k}$ returns the concept contributions. The right square commutes, $g \circ D = b + \sum_i \Phi_i$, so the contributions sum to $f_A(x)$ exactly; since $D \circ E$ only approximates the identity, they recover $f(x)$ up to $\mathrm{FE}(A)$.

B Proofs for Sec. 4

We use the notation and assumptions of Sec. 4.

B.1 Proof of Theorem 1

Proof. For every $x \in \mathcal{X}$, Lipschitz continuity gives

$$\|f(x) - f_A(x)\|_2 = \|g(h(x)) - g(D(C(x)))\|_2 \le L_g\|h(x) - D(C(x))\|_2. \tag{18}$$

Squaring and taking expectations preserves the inequality by monotonicity of expectation. Linearity of expectation allows L_g^2 to be taken outside, and taking square roots gives the result. □

B.2 Proof of Proposition 1

Proof. For every $x \in \mathcal{X}$, the definitions of f_φ and C give

$$f_\varphi(x) = b + \sum_{i=1}^{k}\varphi_i(x) = b + \sum_{i=1}^{k}\langle a_i, \pi_i(h(x))\rangle = \gamma(C(x)) = f_A(x).$$

Thus $f_\varphi = f_A$, and consequently $\mathrm{ADD}(A,\varphi) = \mathrm{RMSE}(f_A, f_\varphi) = 0$. □

B.3 Attribution Rules for Affine Decoded Heads

Suppose that γ has the affine form in Proposition 1. For insertion, only block i is retained, so

$$\gamma(P_iC(x)) - b = \langle a_i, \pi_i(h(x))\rangle.$$

For occlusion, subtracting the prediction with block i removed cancels the baseline and all other block contributions:

$$\gamma(C(x)) - \gamma(C(x) - P_iC(x)) = \langle a_i, \pi_i(h(x))\rangle.$$

For gradient-times-input, we use the affine formula on $\mathcal{W}$, whose gradient is the constant vector $(a_1, \ldots, a_k)$. Consequently,

$$\langle \nabla\gamma(C(x)), P_iC(x)\rangle = \langle a_i, \pi_i(h(x))\rangle.$$

All three rules therefore satisfy the attribution formula of Proposition 1.

B.4 Proof of Theorem 3

Proof. Choose $a(x) = \nabla\gamma(0_{\mathcal{V}})$ for insertion and $a(x) = \nabla\gamma(C(x))$ for occlusion and gradient-times-input. For each input x, define the affine approximation

$$\gamma_{\mathrm{aff},x}(v) = b + \langle a(x), v\rangle, \qquad v \in \mathcal{W}. \tag{19}$$

The bias is fixed at b, so every approximation agrees with γ at zero. Define $f_{\mathrm{aff}}(x) = \gamma_{\mathrm{aff},x}(C(x))$. The triangle inequality for RMSE gives

$$\mathrm{ADD}(A, \varphi) \le \mathrm{RMSE}(f_A, f_{\mathrm{aff}}) + \mathrm{RMSE}(f_{\mathrm{aff}}, f_\varphi). \tag{20}$$

The standard Taylor bound for a function with an M-Lipschitz gradient gives

$$|\gamma(v) - \gamma(w) - \langle \nabla\gamma(w), v - w\rangle| \le \frac{M}{2}\|v - w\|_2^2 \tag{21}$$

whenever the segment from w to v lies in Ω. We use this bound to control both terms in Eq. (20).

For the first term, apply Eq. (21) from zero to $C(x)$ for insertion, and from $C(x)$ to zero for the other two rules. Using $b = \gamma(0_{\mathcal{V}})$ and the chosen $a(x)$, both cases give

$$|f_A(x) - f_{\mathrm{aff}}(x)| = |\gamma(C(x)) - b - \langle a(x), C(x)\rangle| \le \frac{M}{2}\|C(x)\|_2^2. \tag{22}$$

Squaring and taking expectations preserves the inequality by monotonicity of expectation. By linearity of expectation, the constant $M^2/4$ can be taken outside, and taking square roots yields

$$\mathrm{RMSE}(f_A, f_{\mathrm{aff}}) \le \frac{M}{2}\sqrt{\mathbb{E}_{x\sim p}\big[\|C(x)\|_2^4\big]}. \tag{23}$$

For the second term, fix $x \in \mathcal{X}$. The isolated blocks satisfy

$$C(x) = \sum_{i=1}^{k} P_i C(x), \qquad \|C(x)\|_2^2 = \sum_{i=1}^{k} \|P_i C(x)\|_2^2. \tag{24}$$

By linearity of the inner product, the affine prediction therefore decomposes as

$$f_{\mathrm{aff}}(x) = b + \sum_{i=1}^{k} \langle a(x), P_i C(x)\rangle. \tag{25}$$

We compare each attribution $\varphi_i(x)$ with its corresponding affine contribution $\langle a(x), P_i C(x)\rangle$.

For insertion, applying Eq. (21) from zero to $P_i C(x)$ gives

$$|\gamma(P_i C(x)) - b - \langle \nabla\gamma(0_{\mathcal{V}}), P_i C(x)\rangle| \le \frac{M}{2}\|P_i C(x)\|_2^2.$$

For occlusion, applying the same bound from $C(x)$ to $C(x) - P_i C(x)$ and rearranging gives

$$\begin{aligned}&|\gamma(C(x)) - \gamma(C(x) - P_i C(x)) - \langle \nabla\gamma(C(x)), P_i C(x)\rangle| \\ &\quad\le \frac{M}{2}\|P_i C(x)\|_2^2.\end{aligned}$$

For gradient-times-input, the attribution equals $\langle a(x), P_i C(x)\rangle$ by definition. Thus, for all three rules,

$$|\varphi_i(x) - \langle a(x), P_i C(x)\rangle| \le \frac{M}{2}\|P_i C(x)\|_2^2. \tag{26}$$

Using the common baseline b and summing over concepts gives

$$\begin{aligned}|f_{\mathrm{aff}}(x) - f_\varphi(x)| &= \left|\sum_{i=1}^{k}\big(\langle a(x), P_i C(x)\rangle - \varphi_i(x)\big)\right| \\ &\le \sum_{i=1}^{k} |\langle a(x), P_i C(x)\rangle - \varphi_i(x)| \\ &\le \frac{M}{2}\sum_{i=1}^{k}\|P_i C(x)\|_2^2 = \frac{M}{2}\|C(x)\|_2^2.\end{aligned}$$

The inequalities use the triangle inequality and Eq. (26), and the last equality uses Eq. (24). Taking the root mean square as above gives

$$\mathrm{RMSE}(f_{\mathrm{aff}}, f_\varphi) \le \frac{M}{2}\sqrt{\mathbb{E}_{x\sim p}\big[\|C(x)\|_2^4\big]}. \tag{27}$$

Substituting Eqs. (23) and (27) into Eq. (20) proves

$$\mathrm{ADD}(A, \varphi) \le M\sqrt{\mathbb{E}_{x\sim p}\big[\|C(x)\|_2^4\big]}. \qquad \square$$

C Evaluation of Tightness of the Bounds

We evaluate how closely the bounds of Sec. 4 are attained by concept autoencoders obtained with common concept discovery methods.

Setup We use a ResNet-50 trained on ImageNet-1k and, for each of three random seeds, 25,000 images drawn from the ImageNet-1k validation set, which define the empirical distribution p. Half of the images are used to fit the concept autoencoders and prediction heads, and all reported errors are computed on the other half. We split the model at two boundaries: after the last residual stage (*layer4*, $\mathcal{Z} = \mathbb{R}^{S\times d}$ with $S = 49$ and $d = 2048$) and before the final affine layer (*penultimate*, $\mathcal{Z} = \mathbb{R}^d$). At both boundaries the prediction head is affine, $g(z) = W\bar{z}+\beta$, where $\bar{z} = S^{-1}\sum_{s=1}^{S} z_s$ is the spatial average at layer4 and $\bar{z} = z$ at the penultimate boundary. Consequently, the Lipschitz constant of Theorem 1 is available in closed form: $L_g = \|W\|_2$ (the largest singular value of W) at the penultimate boundary, and $L_g = \|W\|_2/\sqrt{S}$ at layer4, since averaging over S locations shrinks the Euclidean norm on $\mathbb{R}^{S\times d}$ by at least a factor $\sqrt{S}$, with equality for errors that are identical at every location. We extract $k \in \{5, 25, 50\}$ concepts with five methods that cover different decoder types: PCA (orthogonal linear decoder), NMF (nonnegative linear decoder), k-means (centroid decoder with soft assignments as concept representations), a sparse autoencoder (affine decoder), and an autoencoder with a nonlinear decoder. Methods acting on spatial latents are applied to each spatial feature vector separately, so each concept has one coefficient per location. PCA, NMF, and k-means use their scikit-learn implementations. Both autoencoders share the encoder $\mathbb{R}^d \to \mathbb{R}^{4k} \to \mathbb{R}^k$ with a ReLU after each layer; the SAE has the affine decoder $\mathbb{R}^k \to \mathbb{R}^d$ and an ℓ_1 penalty on the concept representation, and the nonlinear autoencoder has the decoder $\mathbb{R}^k \to \mathbb{R}^{4k} \to \mathbb{R}^d$ with a ReLU after the hidden layer.

Tightness Ratios Since the errors differ by orders of magnitude across boundaries and methods, we report each bound through the ratio of its left-hand side to its right-hand side,

$$\rho_{\mathrm{FE}} = \frac{\mathrm{FE}(A)}{L_g\,\mathrm{RE}(A)}, \qquad \rho_{\mathrm{ATE}} = \frac{\mathrm{ATE}(A,\varphi)}{\mathrm{FE}(A) + M\sqrt{\mathbb{E}_{x\sim p}\big[\|C(x)\|_2^4\big]}}. \tag{28}$$

A ratio of one means that the bound is attained, and smaller ratios indicate slack. The corresponding ratio $\mathrm{MCE}(A)/\,\mathrm{FE}(A)$ for model completeness cannot be computed, since MCE is an infimum over all heads, and we report the upper estimate $\hat{\rho}_{\mathrm{MCE}}$ described below. Tab. 2 reports the fidelity and model completeness ratios for every boundary, method, and k, and Tab. 3 reports the attribution ratios for the nonlinear autoencoder, the only method for which they differ from one.

Fidelity The fidelity bound holds in every configuration, with ρ_{FE} between 0.092 and 0.106 at layer4 and between 0.250 and 0.270 at the penultimate boundary. The slack has a structural explanation. Write $\varepsilon(x) = h(x) - D(C(x))$ for the reconstruction error and $\bar{\varepsilon}(x)$ for its spatial average, so that $f(x) - f_A(x) = W\bar{\varepsilon}(x)$. The bound then follows from two inequalities,

$$\mathrm{FE}(A) = \sqrt{\mathbb{E}\|W\bar{\varepsilon}\|_2^2} \le \|W\|_2\sqrt{\mathbb{E}\|\bar{\varepsilon}\|_2^2} \le \frac{\|W\|_2}{\sqrt{S}}\sqrt{\mathbb{E}\|\varepsilon\|_2^2} = L_g\,\mathrm{RE}(A). \tag{29}$$

The first holds because W stretches no vector by more than its largest singular value $\|W\|_2$, and the second because averaging over S locations shrinks the Euclidean norm by at least a factor $\sqrt{S}$.

Table 2: Tightness of the fidelity and model completeness bounds, mean over three seeds; the standard deviation over seeds is at most 0.012 for $\hat{\rho}_{\mathrm{MCE}}$ and at most 0.002 for the other ratios. $\rho_{\mathrm{FE}} = \alpha\kappa$ is the tightness of Theorem 1, with alignment α and spatial coherence κ from Eq. (30). $\hat{\rho}_{\mathrm{MCE}}$ upper-bounds the tightness of Theorem 2.

Boundary	Method	k	ρ_{FE}	α	κ	$\hat{\rho}_{\mathrm{MCE}}$
layer4	PCA	5	0.106	0.270	0.394	0.967
		25	0.103	0.266	0.386	0.927
		50	0.099	0.261	0.381	0.939
	NMF	5	0.106	0.269	0.395	0.975
		25	0.102	0.264	0.389	0.960
		50	0.101	0.263	0.385	0.964
	k-means	5	0.105	0.267	0.394	0.993
		25	0.103	0.266	0.388	0.983
		50	0.102	0.264	0.385	0.978
	SAE	5	0.106	0.270	0.392	0.968
		25	0.102	0.265	0.386	0.934
		50	0.099	0.261	0.381	0.943
	Nonlinear AE	5	0.104	0.267	0.390	0.985
		25	0.096	0.256	0.375	0.979
		50	0.092	0.250	0.367	0.971
penultimate	PCA	5	0.270	0.270	1	0.956
		25	0.265	0.265	1	0.908
		50	0.261	0.261	1	0.916
	NMF	5	0.270	0.270	1	0.969
		25	0.264	0.264	1	0.944
		50	0.261	0.261	1	0.949
	k-means	5	0.267	0.267	1	1.000
		25	0.264	0.264	1	1.000
		50	0.262	0.262	1	1.000
	SAE	5	0.269	0.269	1	0.978
		25	0.265	0.265	1	0.916
		50	0.264	0.264	1	0.916
	Nonlinear AE	5	0.268	0.268	1	0.975
		25	0.257	0.257	1	0.969
		50	0.250	0.250	1	0.980

The tightness ratios of the two steps, the *alignment* α and the *spatial coherence* κ, multiply to the tightness ratio of the bound,

$$\rho_{\mathrm{FE}} = \alpha\kappa, \qquad \alpha = \frac{\sqrt{\mathbb{E}\|W\bar{\varepsilon}\|_2^2}}{\|W\|_2\sqrt{\mathbb{E}\|\bar{\varepsilon}\|_2^2}}, \qquad \kappa = \frac{\sqrt{S\,\mathbb{E}\|\bar{\varepsilon}\|_2^2}}{\sqrt{\mathbb{E}\|\varepsilon\|_2^2}}, \tag{30}$$

and both lie in $[0, 1]$ by Eq. (29). The alignment α measures how much of the averaged error lies along the directions that W amplifies most, its top right singular vectors, and the spatial coherence κ measures how much of the error survives averaging over locations, with $\kappa = 1$ at the penultimate boundary. Both factors equal one for an error that is identical at every location and points along the direction that W amplifies most, so the constant L_g in Theorem 1 cannot be improved without assumptions on the error. Empirically, the alignment is nearly the same at both boundaries and for all methods, $\alpha \approx 0.26$, so the reconstruction error is spread over many directions rather than concentrated along the directions that W amplifies most. At layer4, averaging over locations removes a further part of the error, $\kappa \approx 0.38$, which explains the additional slack at this boundary. As k grows, RE and FE decrease for all methods, and the ratio ρ_{FE} decreases slightly, on average from 0.105 to 0.099 at layer4 and from 0.269 to 0.260 at the penultimate boundary between $k = 5$ and $k = 50$.

Table 3: Tightness of the attribution bound for the nonlinear autoencoder, mean over three seeds. Bound is $\mathrm{FE}(A) + M\sqrt{\mathbb{E}\left[\|C(x)\|_2^4\right]}$ with FE on the explained scores, shared by insertion (Ins.), occlusion (Occ.), and gradient-times-input (G×I), and ρ_{ATE} is the tightness of the attribution bound obtained by combining Eq. (13) with Theorem 3. The bound, and with it ρ_{ATE}, varies across seeds by up to 40% because M is estimated by sampling, whereas ADD varies by at most 0.4. For the four methods with affine decoders, $\mathrm{ADD} < 2 \cdot 10^{-5}$ and $\rho_{\mathrm{ATE}} = 1$ for all three attribution functions.

			$\mathrm{ADD}(A, \varphi)$			$\rho_{\mathrm{ATE}} \times 10^3$		
Boundary	k	Bound	Ins.	Occ.	G×I	Ins.	Occ.	G×I
layer4	5	2281	1.31	1.39	0.10	3.0	3.0	2.9
	25	2381	2.34	2.92	0.06	2.9	1.9	2.4
	50	2749	3.49	4.20	0.06	2.7	1.4	2.0
penultimate	5	2593	0.78	0.87	0.25	3.0	2.8	2.9
	25	1909	1.70	3.32	0.13	3.5	2.2	2.9
	50	1726	2.40	4.46	0.09	3.9	2.2	2.9

Model Completeness Model completeness error is an infimum over all heads and cannot be computed exactly. Every fitted head $\hat{\gamma}$ is admissible, so its held-out error $\widehat{\mathrm{MCE}} = \mathrm{RMSE}(f, \hat{\gamma} \circ C)$ estimates an upper bound on $\mathrm{MCE}(A)$. We fit two heads on the fitting half. The first refits the decoded head within its own family, starting from $g \circ D$: when D is affine on each location, the functions $g \circ D$ are exactly the affine maps of the averaged representation $\bar{v}$, so we fit $\hat{\gamma}(v) = \hat{W}\bar{v} + \hat{\beta}$ by least squares, and for the nonlinear decoder we fine-tune copies of D and g by gradient descent. The second adds a learned correction to the decoded head, $\hat{\gamma} = g \circ D + r$, where r is a multilayer perceptron initialized to zero. Both heads start from $g \circ D$, and we select them by validation error, so they do not exceed $\mathrm{FE}(A)$ beyond sampling error. We report the smaller of the two as $\widehat{\mathrm{MCE}}$ and $\hat{\rho}_{\mathrm{MCE}} = \widehat{\mathrm{MCE}}/\,\mathrm{FE}(A)$, which upper-bounds the true tightness of Theorem 2. Consequently, our results can certify slack in this bound, since $\mathrm{FE}(A) - \widehat{\mathrm{MCE}}$ lower-bounds $\mathrm{FE}(A) - \mathrm{MCE}(A)$, but they cannot certify its tightness. The gap is modest throughout: $\hat{\rho}_{\mathrm{MCE}}$ ranges from 0.91 to 1.00. It is largest for PCA and the SAE, and a nonlinear correction reduces the error by at most 9% over the refitted decoded head. For k-means at the penultimate boundary, $\hat{\rho}_{\mathrm{MCE}} = 1$: its assignments are nearly one-hot, and $g \circ D$ maps each cluster to the mean of f over that cluster, which is already the best any head can do.

Attribution We explain the score of the predicted class for each input and evaluate all three attribution functions of Eqs. (9) to (11). The theory assumes a scalar output, whereas the explained class varies between inputs. The bounds still apply, because the triangle inequality in Eq. (13) and the Taylor bound in the proof of Theorem 3 hold for each input and every class separately; we therefore compute FE on the explained scores only and take M as the largest curvature constant over the explained classes. For the four methods with affine decoders, $\gamma = g \circ D$ is a composition of affine maps, since g is affine at both boundaries, and is therefore affine even though the SAE and k-means have nonlinear encoders. In agreement with Proposition 1, the additivity error vanishes up to floating-point error for all three attribution functions, $\mathrm{ADD}(A, \varphi) < 2 \cdot 10^{-5}$ compared with $\mathrm{FE} \approx 6$ on the explained scores, and the attribution error coincides with the fidelity error, so $\rho_{\mathrm{ATE}} = 1$. Only the nonlinear decoder produces curvature. We estimate M from the largest ratio of gradient differences over sampled pairs of nearby concept representations, which lower-bounds the true constant, so the reported bound is itself an estimate. The curvature term does not depend on the scale of the concept representations: replacing E by λE and D by $D(\cdot/\lambda)$ for $\lambda > 0$ leaves f_A and the attributions unchanged, scales M by λ^{-2}, and scales $\sqrt{\mathbb{E}\left[\|C(x)\|_2^4\right]}$ by λ^2, so the looseness of the bound reflects the curvature of γ rather than the units of the concept representations. For the nonlinear autoencoder (Tab. 3), both bounds hold for all three attribution functions but are loose, with ρ_{ATE} between 0.001 and 0.004. Gradient-times-input is nearly additive, with $\mathrm{ADD} \leq 0.25$, whereas insertion and occlusion reach 3.5 and 4.5 at $k = 50$. Better fidelity does not imply better attributions: as k grows, FE on the explained scores decreases while ATE for insertion stays nearly

constant, and for occlusion ATE even falls below FE because the additivity error partly cancels the fidelity error.

Limitations of the Evaluation All errors are estimated on a finite held-out sample. Model completeness error is only bounded from above, and M is estimated from below. The closed-form Lipschitz constant restricts the evaluation to boundaries with affine prediction heads, and at earlier boundaries L_g would itself have to be bounded.