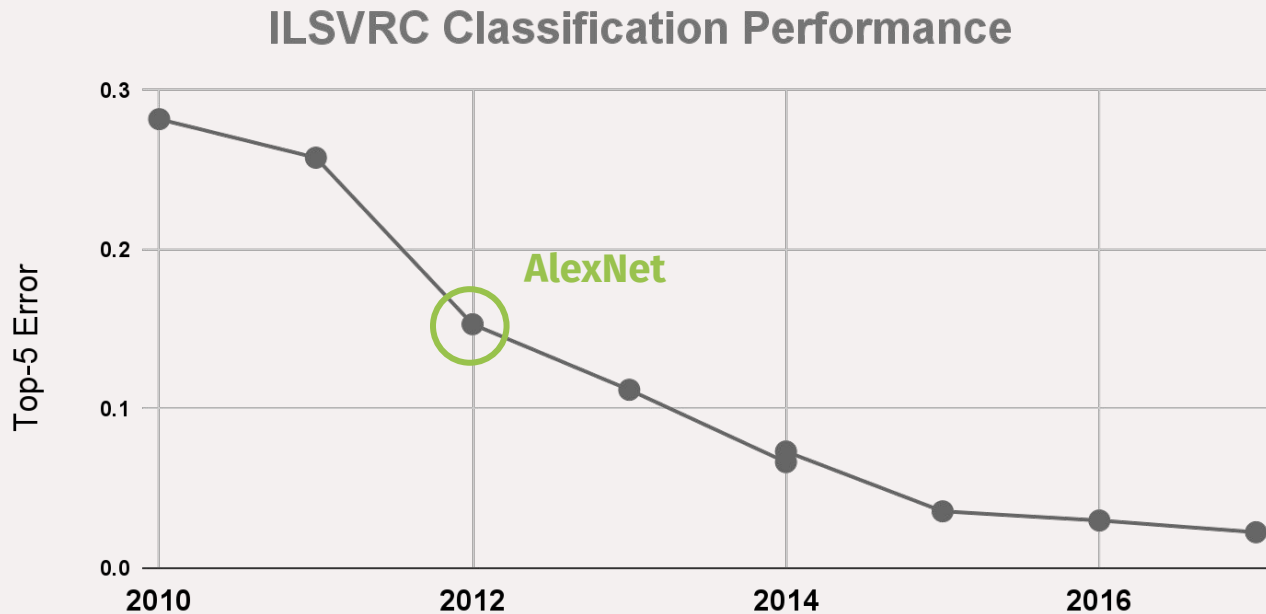


On the Evolution of Interpretability Methods

Ward Gauderis · Thomas Dooms · Geraint Wiggins · José Oramas

13 years since AlexNet, 8 since Attention and 3 since ChatGPT... Deep models are here to stay



Adoption has not only increased in volume but in sophistication.

Microsoft's speech recognition engine listens as well as a human

"This is an historic achievement" - Xuedong Huang



Andrew Tarantola, @terrortola
10.18.16 in [Personal Computing](#)

The Big Read **Driverless vehicles**

+ Add to myFT

Driverless cars inspire a new gold rush in California

MAY 23, 2017 by: [Leslie Hook](#) and [Tim Bradshaw](#)

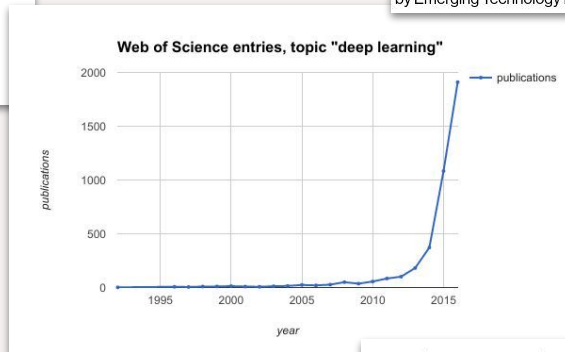
BUSINESS CULTURE GEAR IDEAS SCIENCE MORE

SIGN IN SUBSCRIBE

TOM SIMONITE BUSINESS 10.25.2019 02:00 AM

Google Search Now Reads at a Higher Level

The company is incorporating new software that better understands subtleties of language, with the biggest changes for queries outside the US.



Intelligent Machines

Deep-Learning Machine Listens to Bach, Then Writes Its Own Music in the Same Style

Can you tell the difference between music composed by Bach and by a neural network?

by Emerging Technology from the arXiv December 14, 2016

Article | [Open Access](#) | Published: 29 August 2019

Deep Learning to Improve Breast Cancer Detection on Screening Mammography

Li Shen , Laurie R. Margolies, Joseph H. Rothstein, Eugene Fluder, Russell McBride & Weiva Sieh

Scientific Reports 9, Article number: 12495 (2019) | [Cite this article](#)

9229 Accesses | 2 Citations | 27 Altmetric | [Metrics](#)

Yes! Interpreting these models is necessary for trusting their predictions in high-stakes scenarios.



Clever Hans Effect



Article | [Open access](#) | Published: 11 March 2019

Unmasking Clever Hans predictors and assessing what machines really learn

[Sebastian Lapuschkin](#), [Stephan Wäldchen](#), [Alexander Binder](#), [Grégoire Montavon](#),
[Wojciech Samek](#)  & [Klaus-Robert Müller](#) 

[Nature Communications](#) **10**, Article number: 1096 (2019) | [Cite this article](#)

Yes! Interpreting these models is necessary for trusting their predictions in high-stakes scenarios

Article | Published: 31 May 2021

AI for radiographic COVID-19 detection selects shortcuts over signal

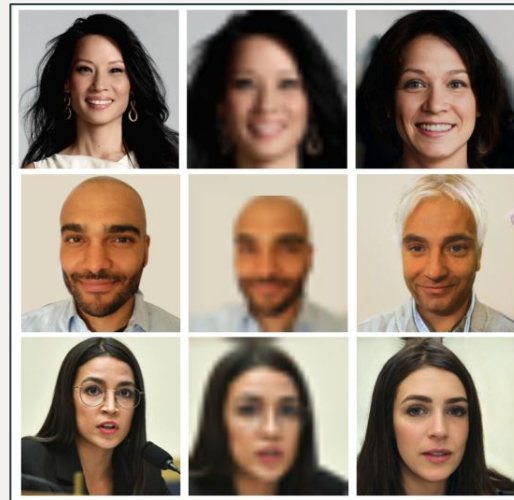
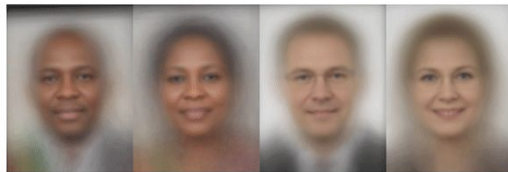
[Alex J. DeGrave](#), [Joseph D. Janizek](#) & [Su-In Lee](#) ✉

[Nature Machine Intelligence](#) 3, 610–619 (2021) | [Cite this article](#)



It can exploit shortcuts ...

Gender Classifier	Darker Male	Darker Female	Lighter Male	Lighter Female	Largest Gap
Microsoft	94.0%	79.2%	100%	98.3%	20.8%
FACE++	99.3%	65.5%	99.2%	94.0%	33.8%
IBM	88.0%	65.3%	99.7%	92.9%	34.4%



... and hallucinate

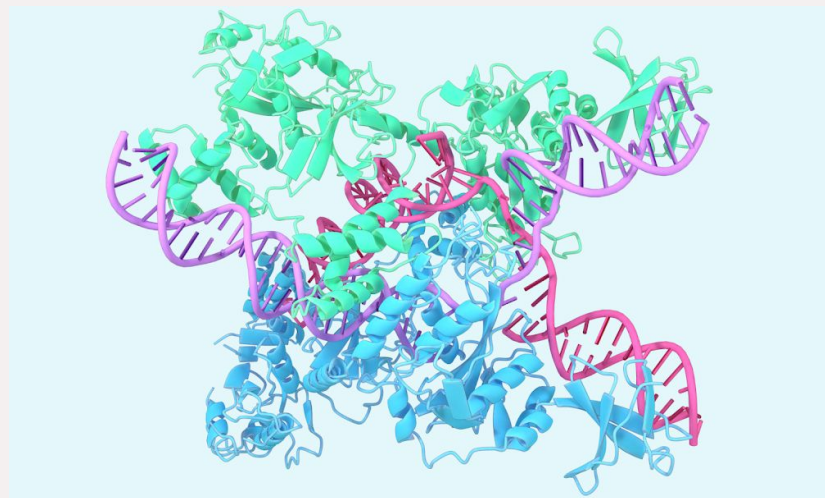
... be biased & unfair...

Current models outperform human experts. What do these models know that we don't?

Games (2016)

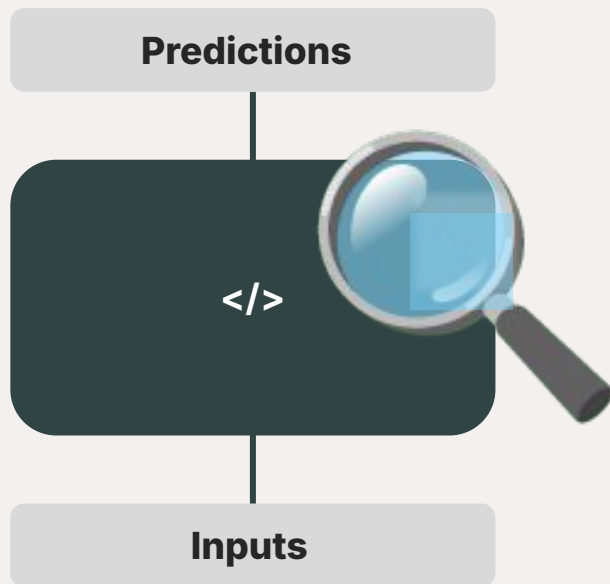


Sciences (2018)



Transparency - AI Interpretability

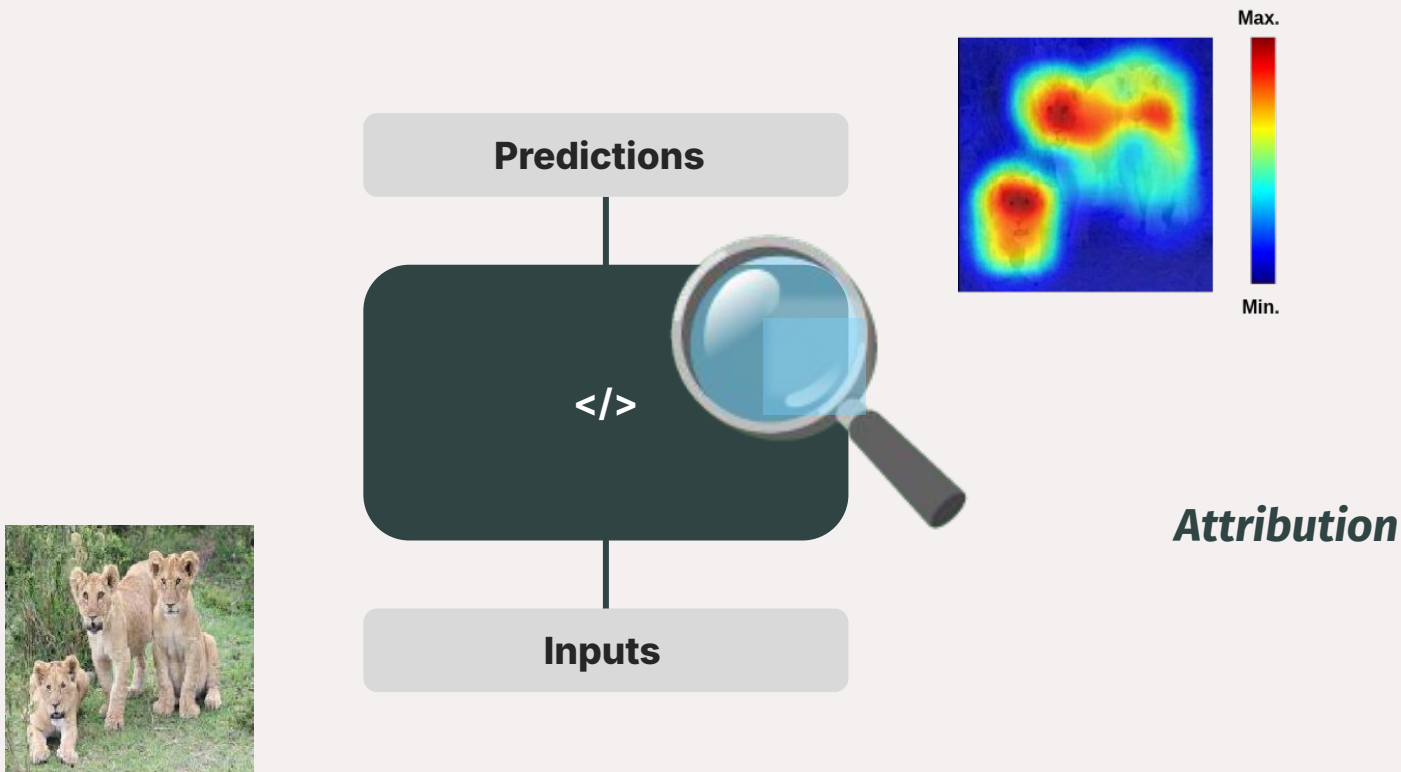
Desideratum: Understanding the inner-workings of the decision-making process



Discovery

Transparency - AI Interpretability

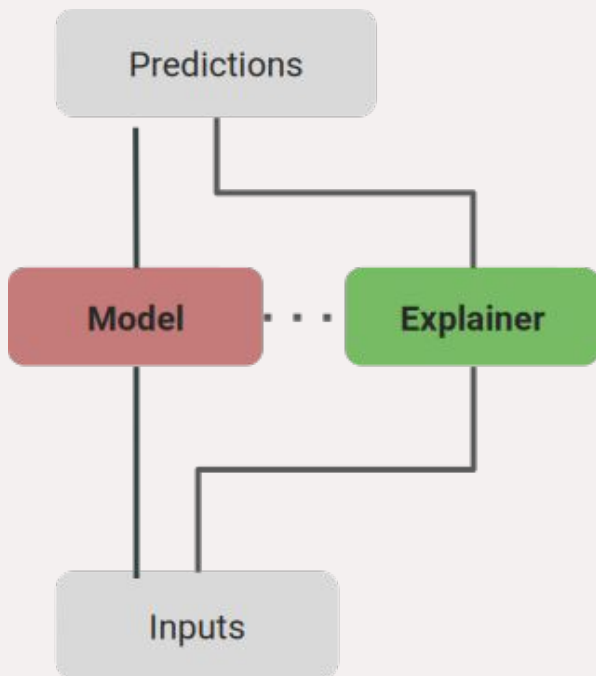
In practice: Relating inputs to outputs



Transparency - AI Interpretability

In practice: Relating inputs with outputs

Post-Hoc Attribution (XAI)

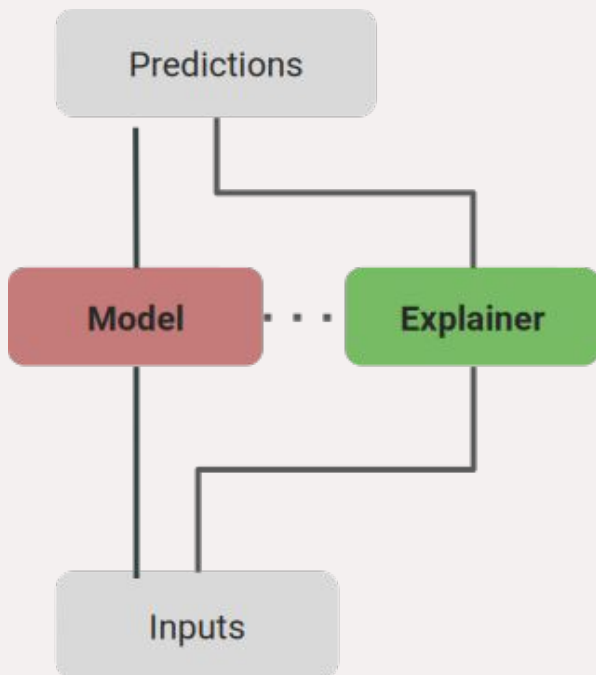


- Gradient and inversion-based methods
- Occlusion methods (e.g. RISE)
- Surrogate methods (e.g. LIME)

Transparency - AI Interpretability

Issue: Produced explanations are not always faithful

Post-Hoc Attribution (XAI)



- Gradient and inversion-based methods
- Occlusion methods (e.g. RISE)
- Surrogate methods (e.g. LIME)

ARTICLE |  **FREE ACCESS**

Sanity checks for saliency maps

Authors:  [Julius Adebayo](#),  [Justin Gilmer](#),  [Michael Muelly](#),  [Ian Goodfellow](#),  [Moritz Hardt](#),  [Been Kim](#) | [Authors Info & Claims](#)

[NIPS'18: Proceedings of the 32nd International Conference on Neural Information Processing Systems](#)
Pages 9525 - 9536

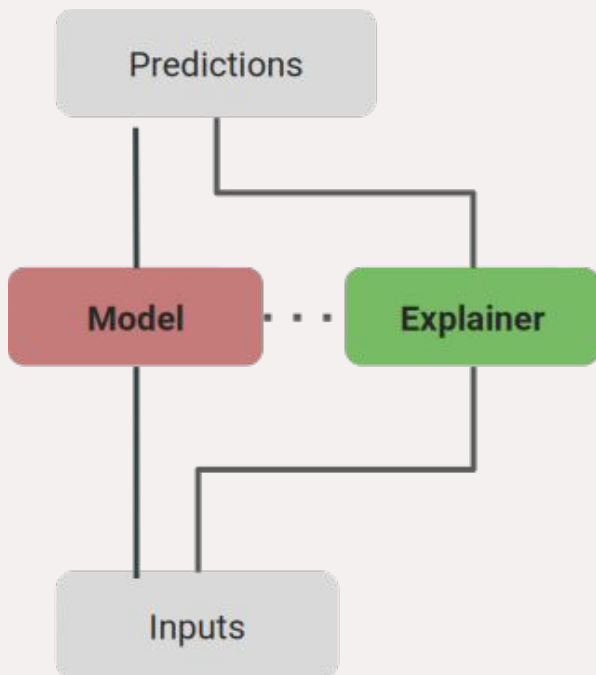
Published: 03 December 2018 [Publication History](#)



Transparency - AI Interpretability

Design models that are interpretable in first place

Post-Hoc Attribution (XAI)



Perspective | Published: 13 May 2019

Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead

[Cynthia Rudin](#) ✉

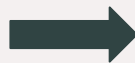
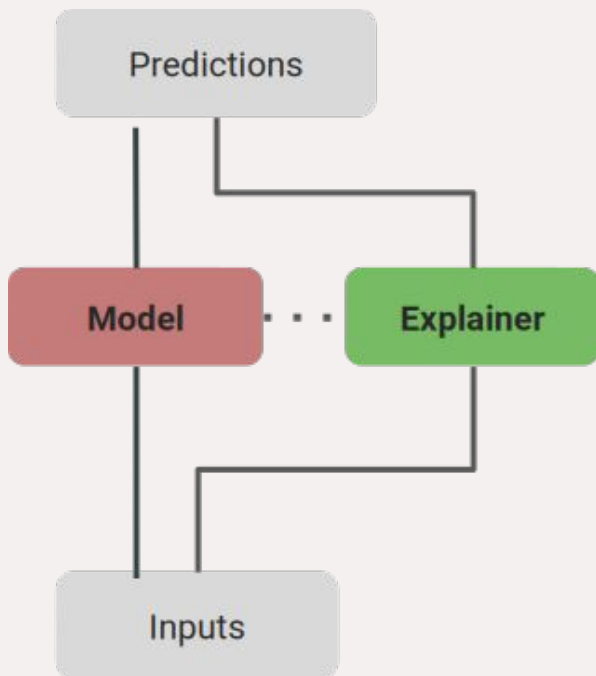
[Nature Machine Intelligence](#) 1, 206–215 (2019) | [Cite this article](#)

- XAI provides a false sense of security
- Not every problem requires a complex model

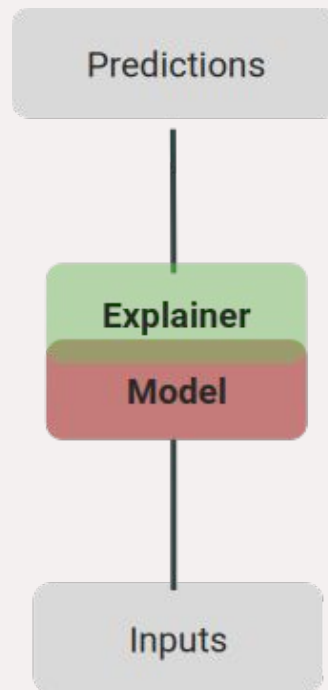
Transparency - AI Interpretability

Integrate explanation capabilities at design time

Post-Hoc Attribution (XAI)



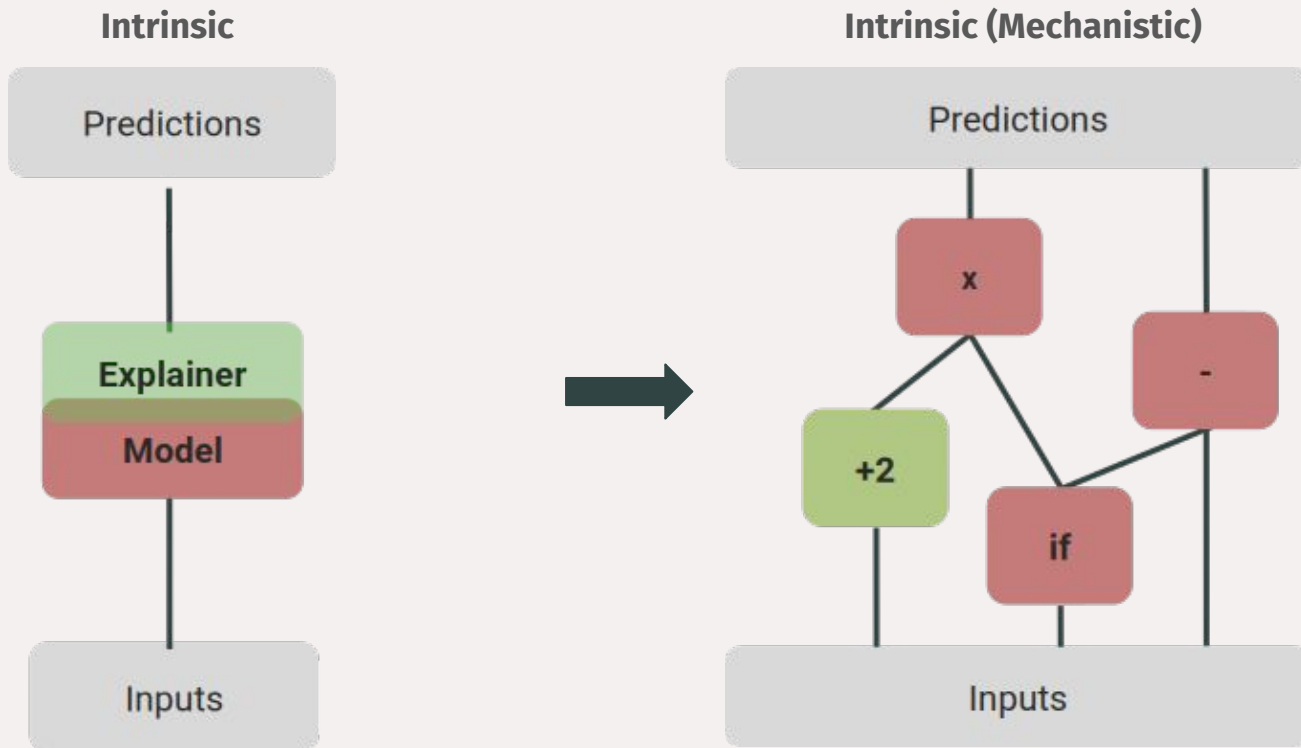
Intrinsic



- Decision Trees
- ProtoPNets
- Concept Bottleneck Models

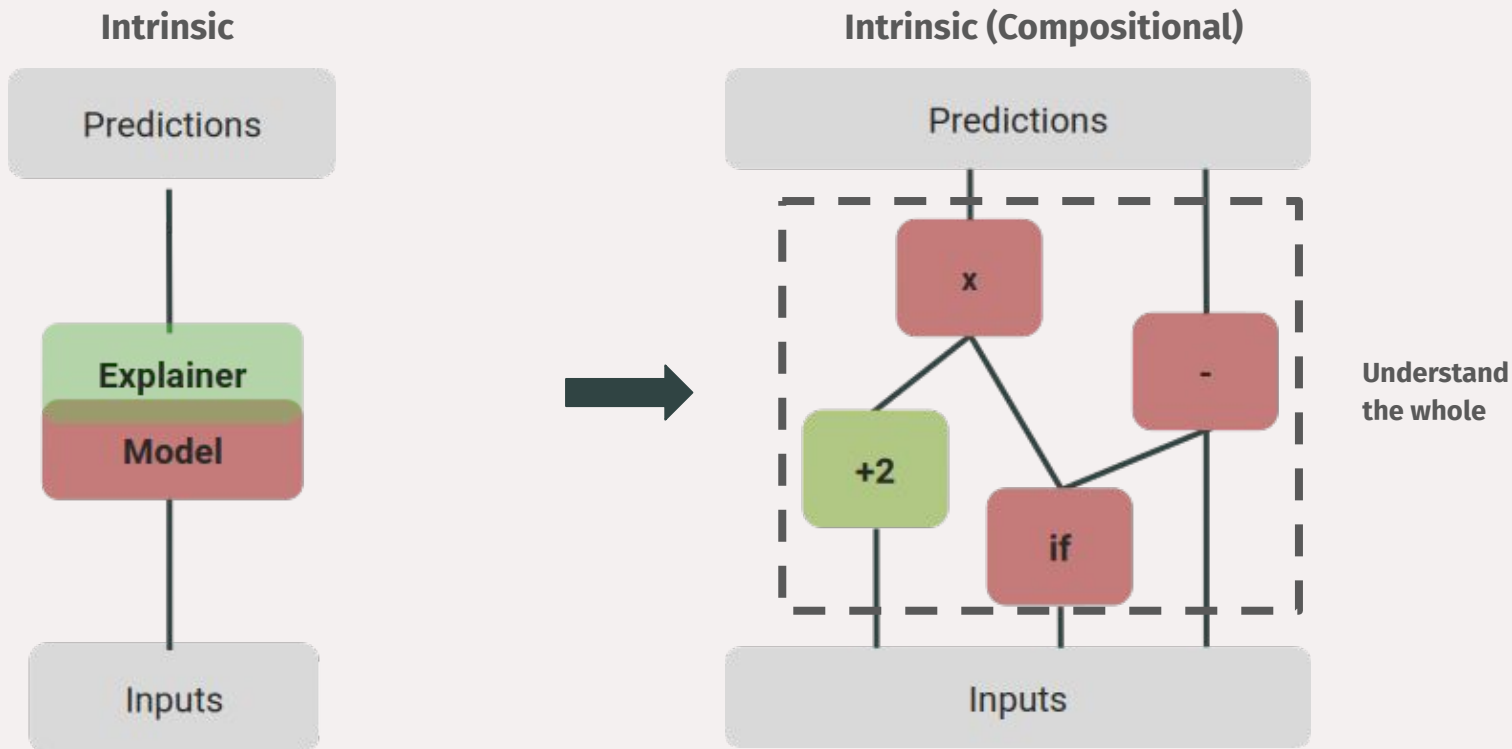
Transparency - AI Interpretability

Going from feature occurrence to understanding internal processes



Transparency - AI Interpretability

From internal processes to understanding the whole



Coming up...

Early Interpretability Methods

José Oramas

From Classical to Mechanistic Interpretability

Ward Gauderis, José Oramas & Thomas Dooms

From Mechanistic to Compositional Interpretability

Ward Gauderis & Thomas Dooms

Early Interpretability Methods

José Oramas

Coming up...

Early Interpretability Methods

José Oramas

From Classical to Mechanistic Interpretability

Ward Gauderis, José Oramas & Thomas Dooms

From Mechanistic to Compositional Interpretability

Ward Gauderis & Thomas Dooms

This chapter...

Post-hoc semantic mapping of neuron activations

What does a neuron do?

Inherently interpretable methods

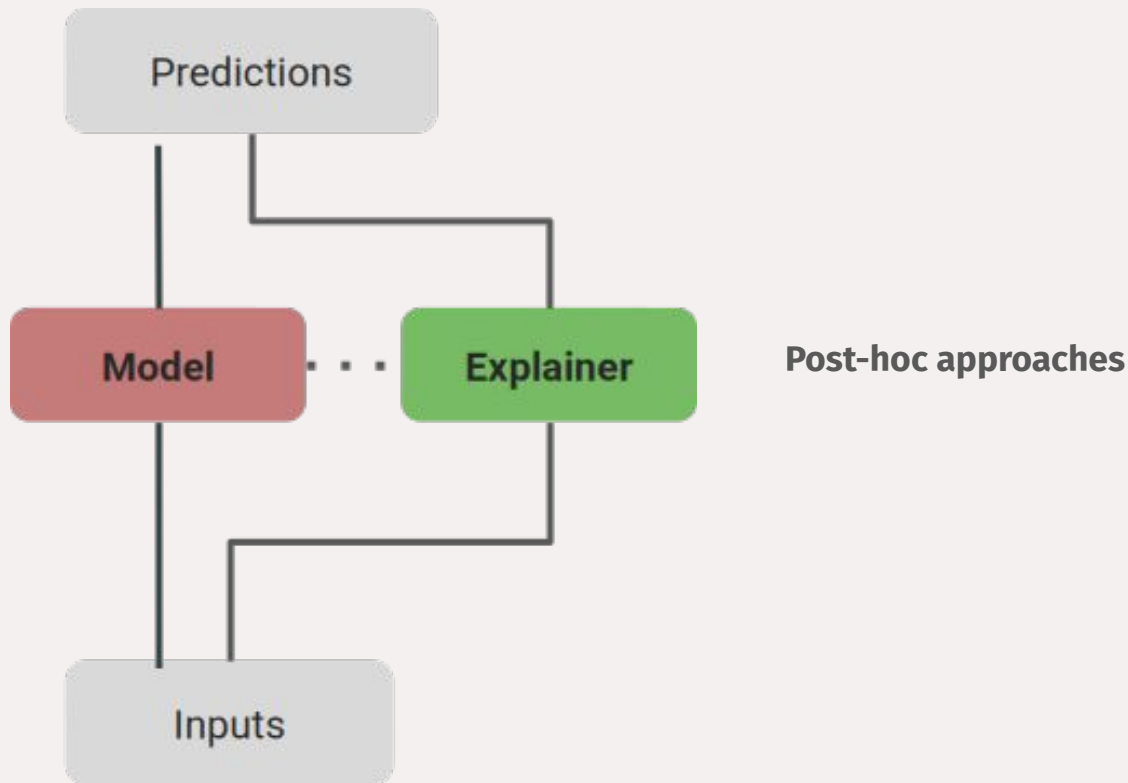
Can we make neurons understandable?

Evaluation protocols

When do we understand neurons?

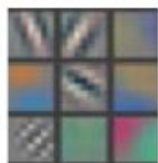
Post-hoc semantic mapping of neuron activations.

What does a neuron do?



Feature visualization

Idea: Dense visual inspection of projected features across the model.



Layer 1

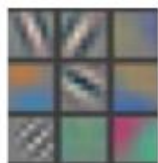


How?

- Feed-forward until a neuron of interest.
- Project the activations back to the input space.
- Collect visualizations of top activations.

Feature visualization

Idea: Dense visual inspection of projected features across the model.

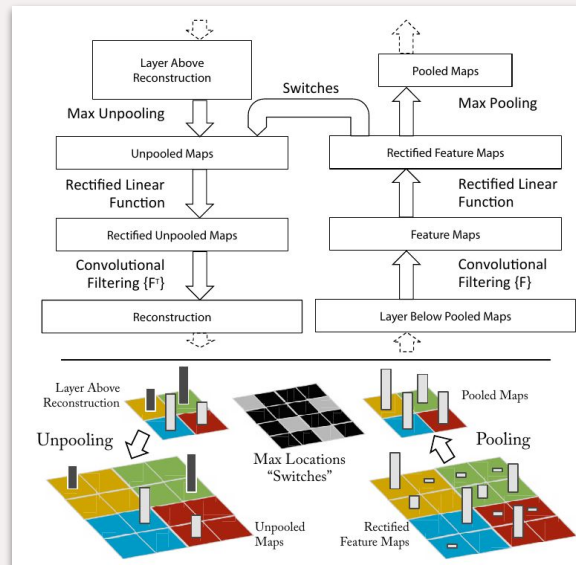


Layer 1



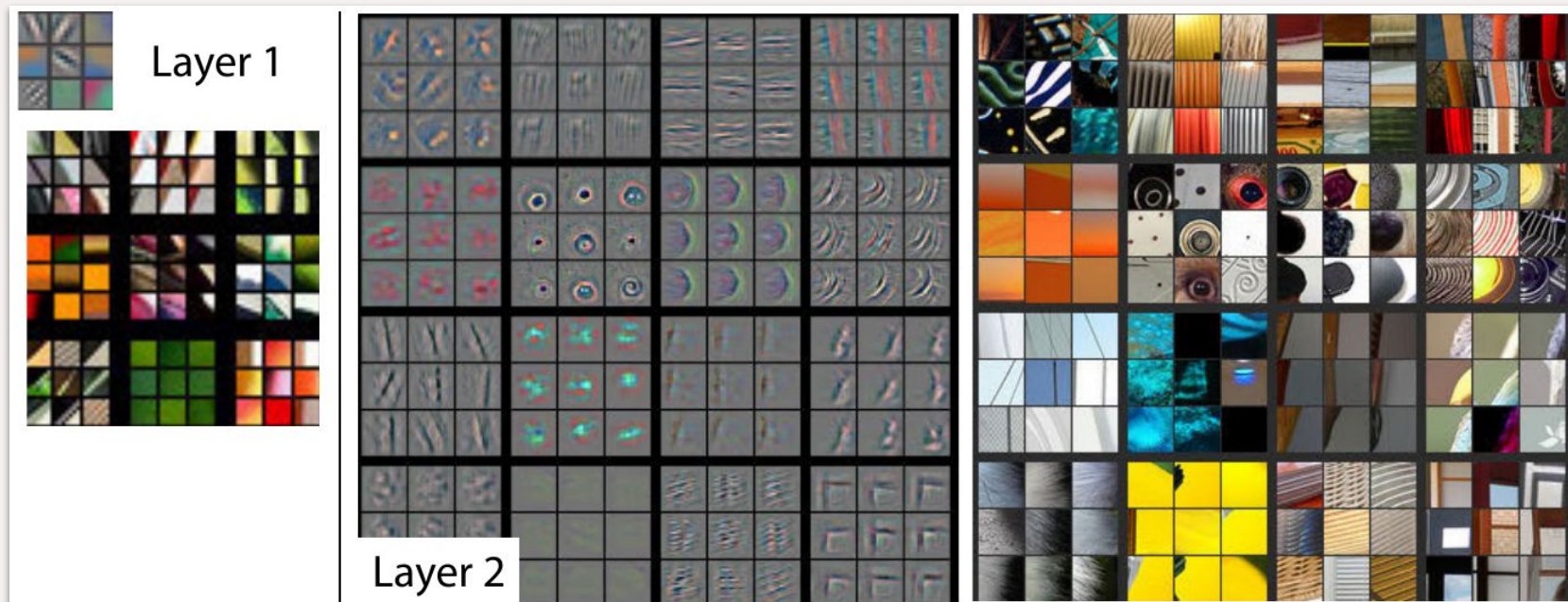
How?

- Feed-forward until a neuron of interest.
- Project the activations back to the input space.
- Collect visualizations of top activations.



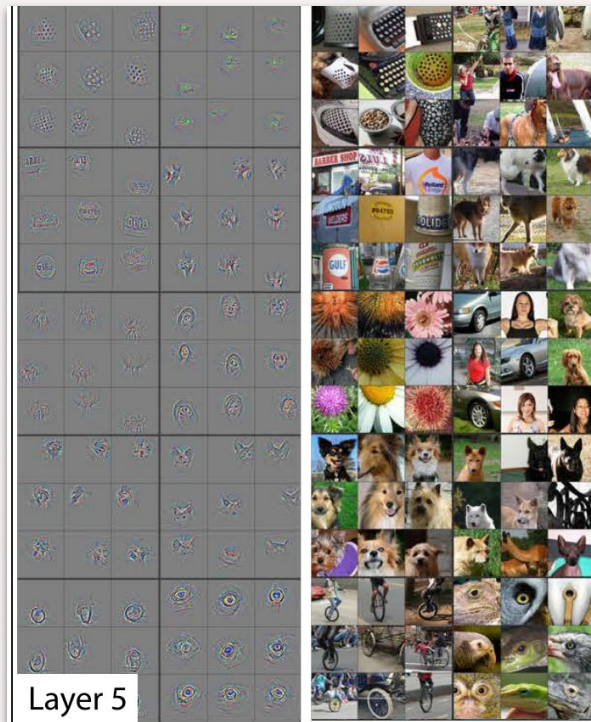
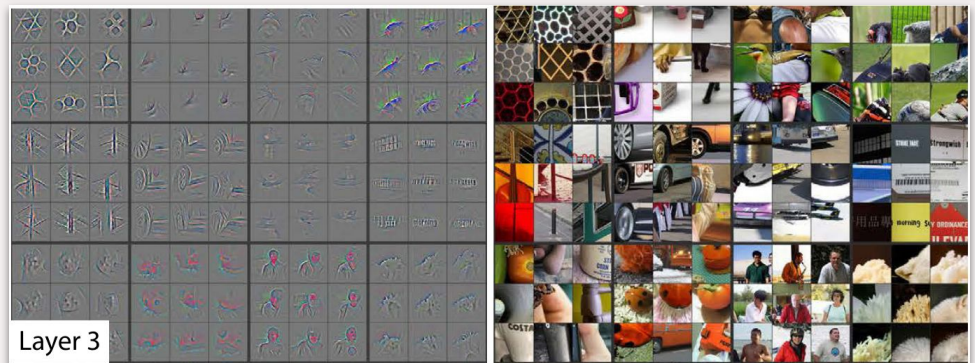
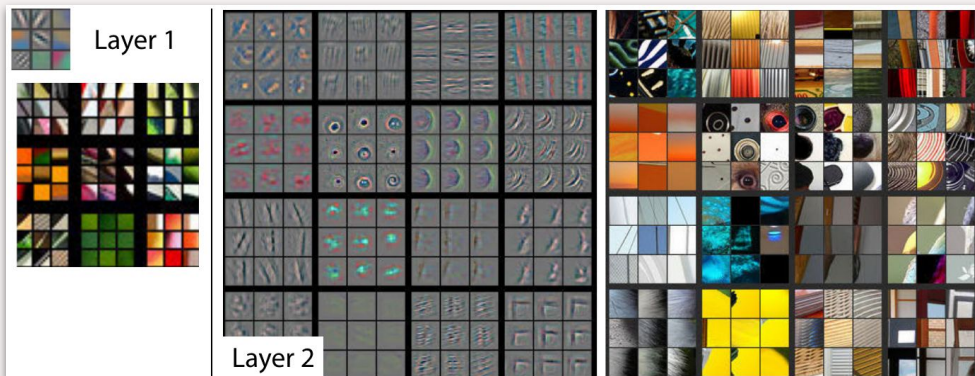
Feature visualization

Idea: Dense visual inspection of projected features across the model.



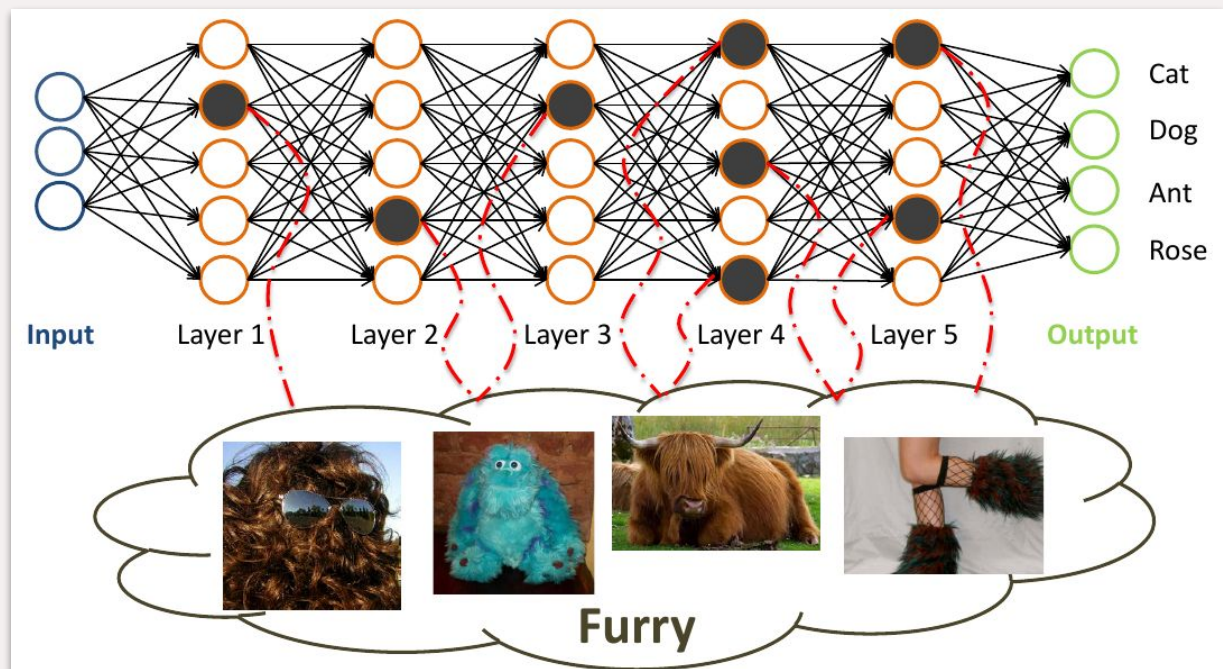
Feature visualization

Idea: Dense visual inspection of projected features across the model.



Linking neuron activations with visual attributes.

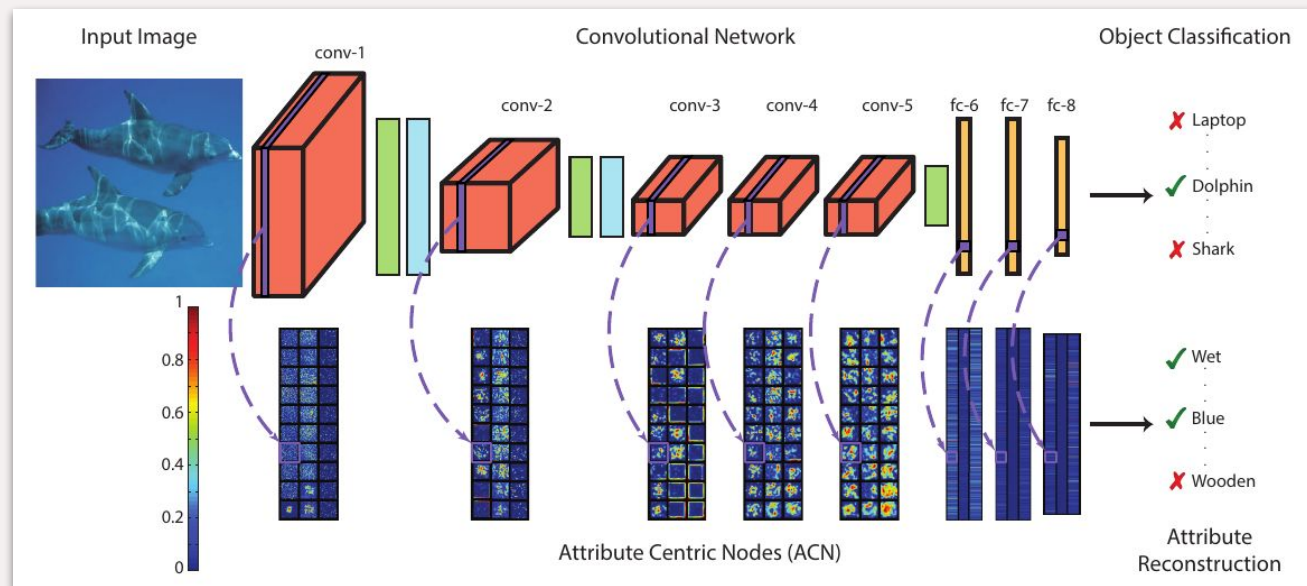
Idea: Identify relationships between activations and attributes.



Assumption: there is a set of units responsible of encoding certain information

Linking neuron activations with visual attributes.

Idea: Identify relationships between activations and attributes.



How:

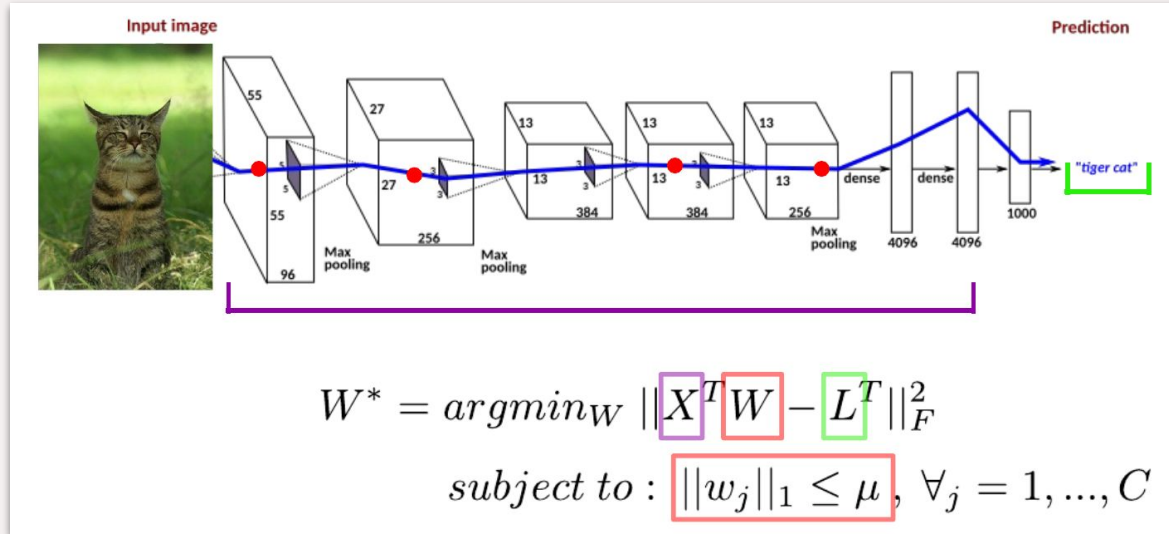
$$\mathbf{W}^* = \underset{\mathbf{W}}{\operatorname{argmin}} \|\mathbf{X}^\top \mathbf{W} - \mathbf{L}^\top\|_F^2$$

subject to: $\|\mathbf{w}_j\|_1 \leq \mu_j \quad \forall j = 1, \dots, d$

Linking neuron activations wrt. the original task.

Idea: Identify relationships between activations and the classes of interest.

Advantage: No need for additional data.

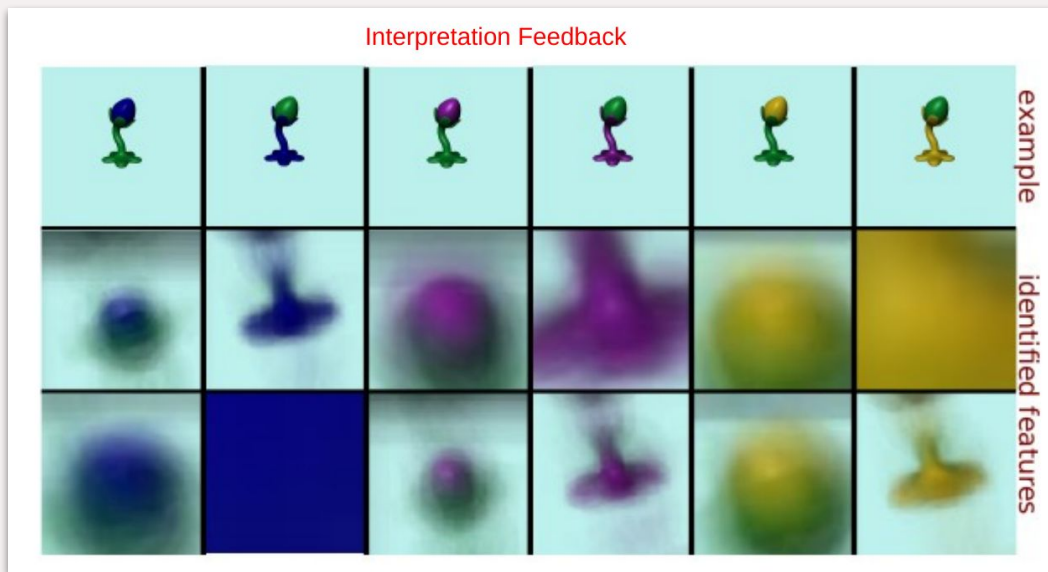


Assumption: there is a set of units responsible of encoding certain information

Linking neuron activations wrt. the original task.

Idea: Identify relationships between activations and the classes of interest.

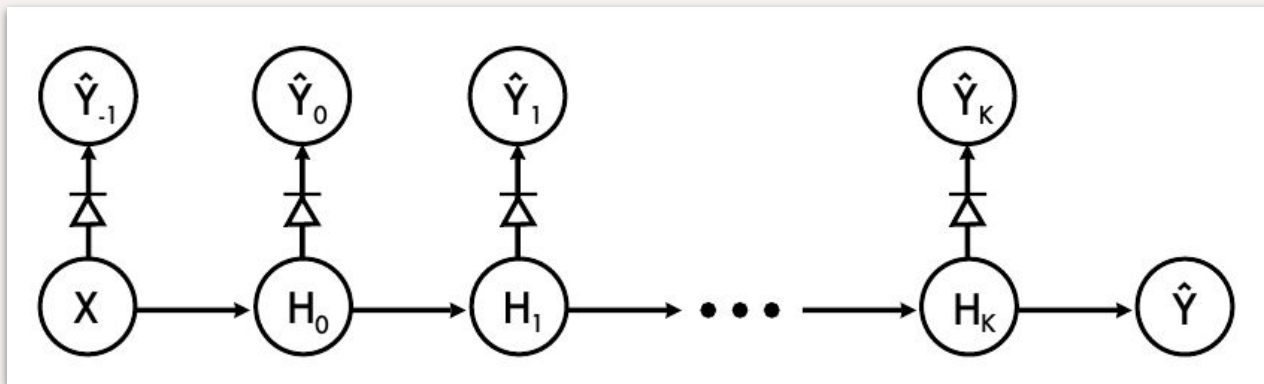
Advantage: No need for additional data.



Assumption: there is a set of units responsible of encoding certain information

Analyzing the predictive signal of neuron activations.

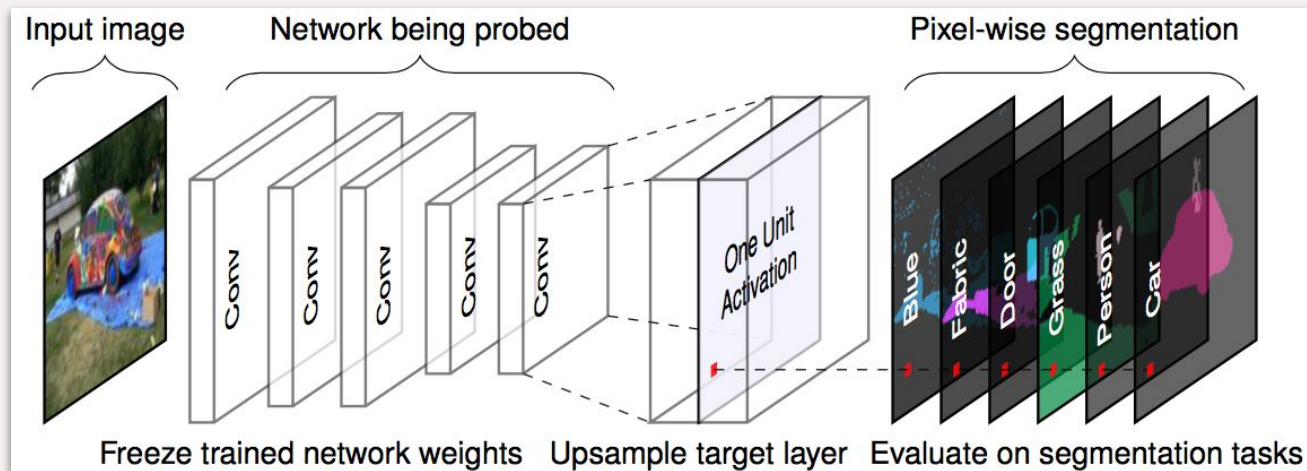
Idea: Use simple classifier (probes) to inspect the representation.



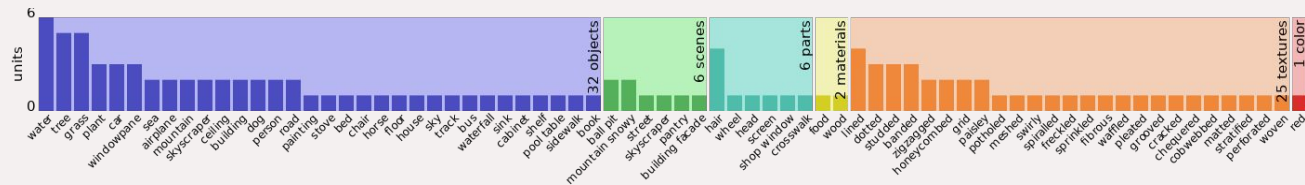
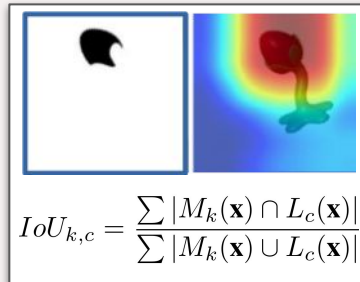
- Lightweight probes -> low capacity & expressivity
- Measure predictive performance across layers (and throughout training)
- Performance on the original task vs. on a new task

Linking neuron activations with semantic concepts.

Idea: Quantify model interpretability with segmentation overlap.

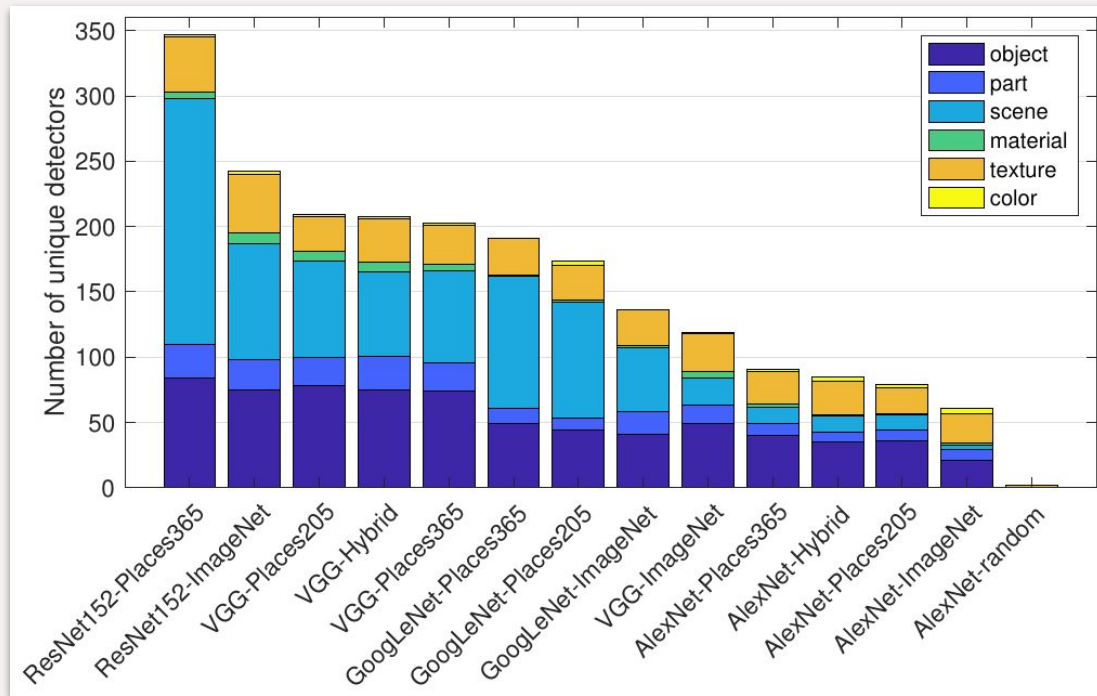


Measuring overlap



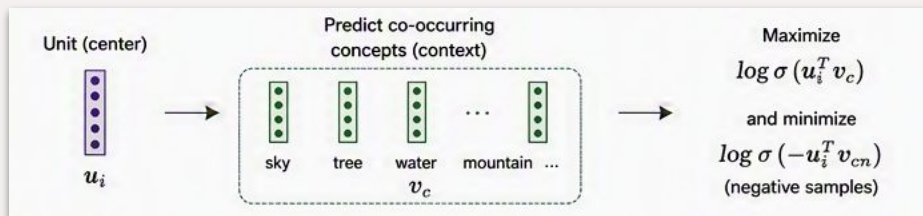
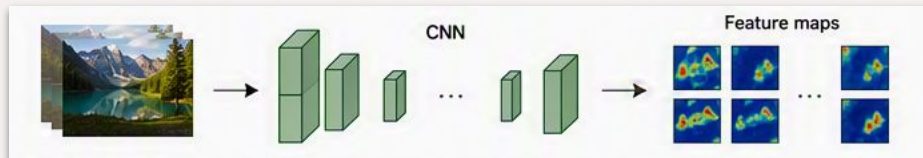
Linking neuron activations with semantic concepts.

Idea: Quantify model interpretability with concept histograms.



Linking sets of neuron activations with concepts.

Assumption: semantic representations are distributed



Units must be studied in conjunction

- Collect unit activations for inputs from a specific probe
- Learn how to weight the collected activations to perform semantic tasks

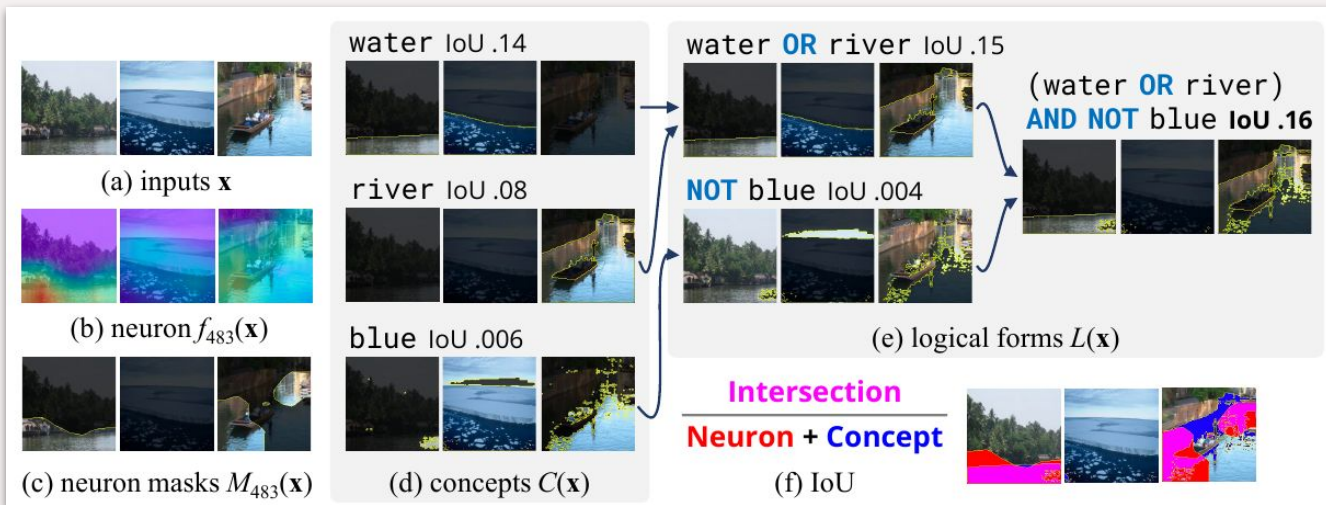
Learned weights -> Embedding

Concept-embedding Analysis

Units (channels)	Concepts					cosine similarity
	"sky"	"tree"	"water"	"mountain"	...	
1	0.12	0.63	0.07	0.21	...	
2	-0.08	0.05	0.74	0.01	...	
...	...	...	...	...	...	
K	0.03	0.36	0.02	0.58	...	

Linking neuron activations with concept compositions.

Observation: some abstract concepts can't be covered by a single neuron



Proposal

- Expand the search space to include compositional logical forms.
- For a neuron, find a logical formula that best match its activation pattern.

How:
$$\delta(n, C) \triangleq \text{IoU}(n, C) = \left[\sum_{\mathbf{x}} \mathbb{1}(M_n(\mathbf{x}) \wedge C(\mathbf{x})) \right] / \left[\sum_{\mathbf{x}} \mathbb{1}(M_n(\mathbf{x}) \vee C(\mathbf{x})) \right]$$

Finding: concept compositions were better approximators of neuron behavior

This chapter...

Post-hoc semantic mapping of neuron activations

What does a neuron do?

Inherently interpretable methods

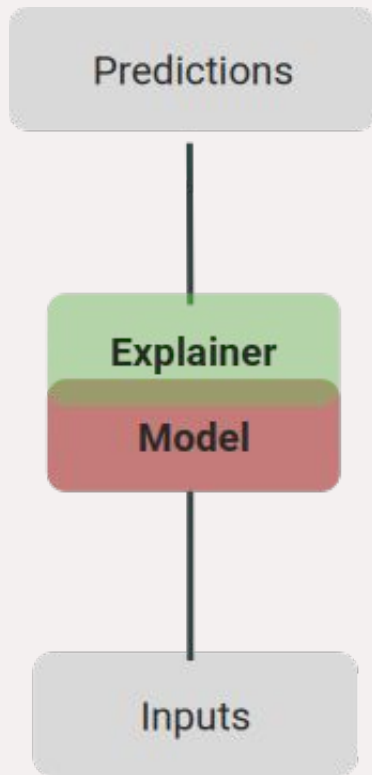
Can we make neurons understandable?

Evaluation protocols

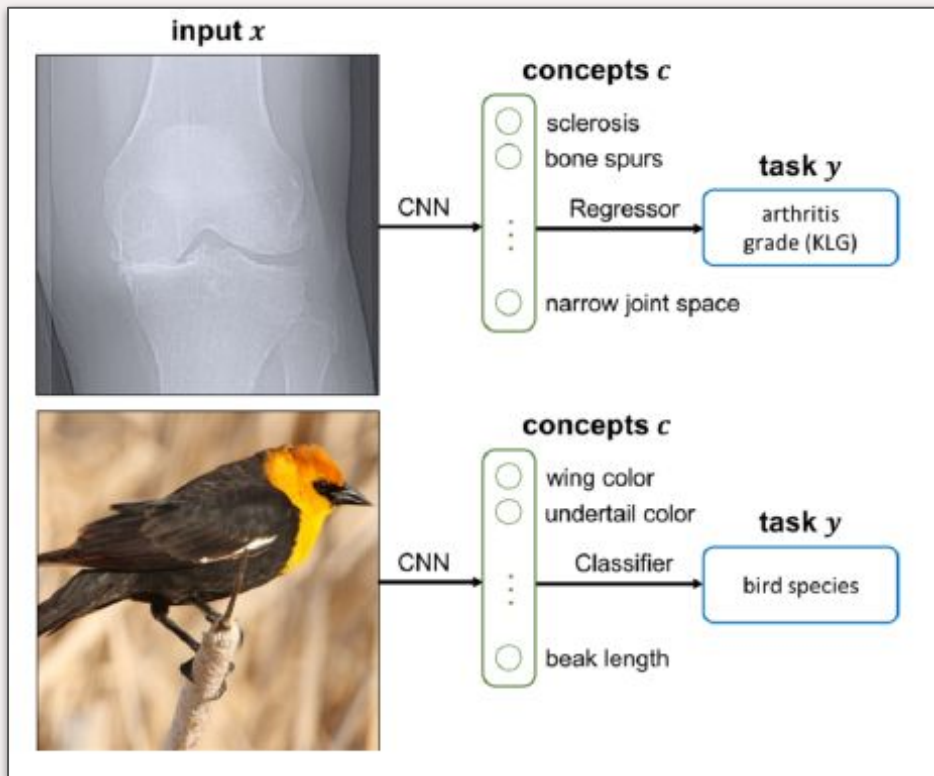
When do we understand neurons?

Inherently interpretable methods

How can we make neurons understandable?

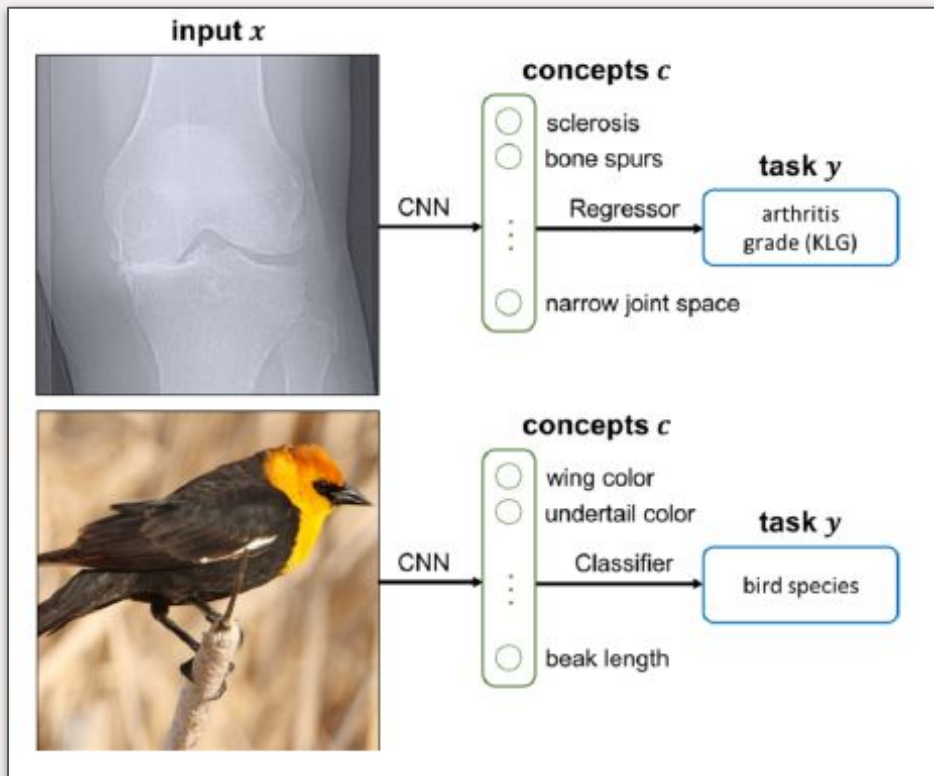


Injecting interpretable units via concept bottlenecks.



- Enforce the learning of pre-defined intermediate semantic concepts
- Make a prediction out of these concepts

Injecting interpretable units via concept bottlenecks.



- Enforce the learning of pre-defined intermediate semantic concepts
- Make a prediction out of these concepts

How:

$$\{(x^{(i)}, y^{(i)}, c^{(i)})\}_{i=1}^n \quad \text{Data}$$

$$f(g(x)) \quad \text{Target model}$$

where

$$\hat{c} = g(x)$$

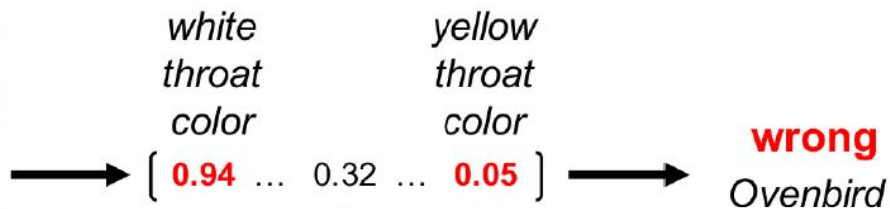
Concept prediction

$$\hat{y} = f(g(x))$$

Downstream task

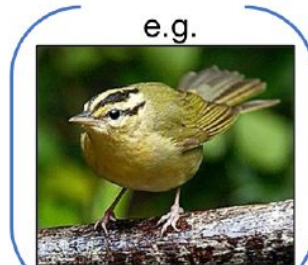
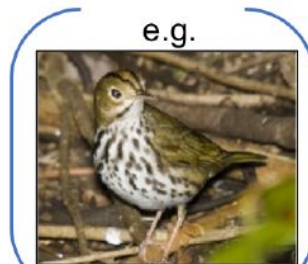
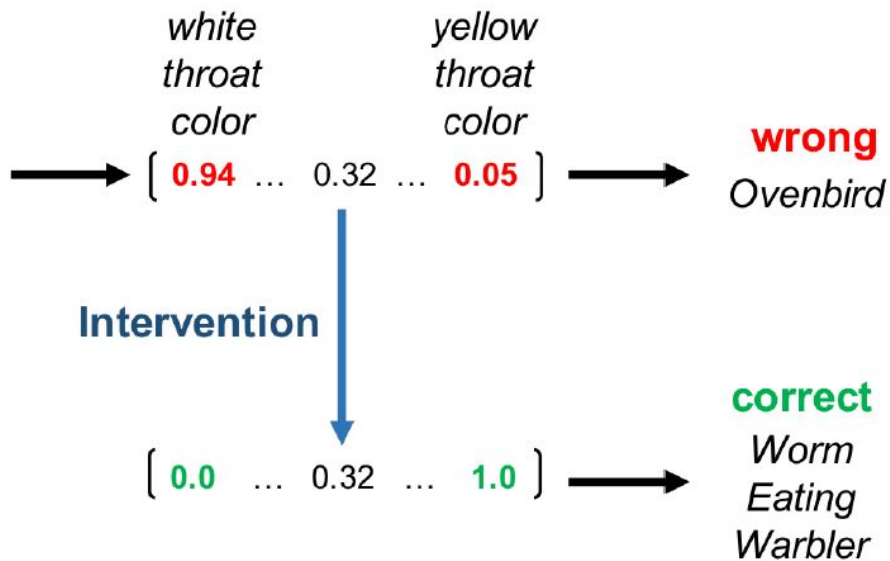
Injecting interpretable units via concept bottlenecks.

Advantage: enables manual intervention.



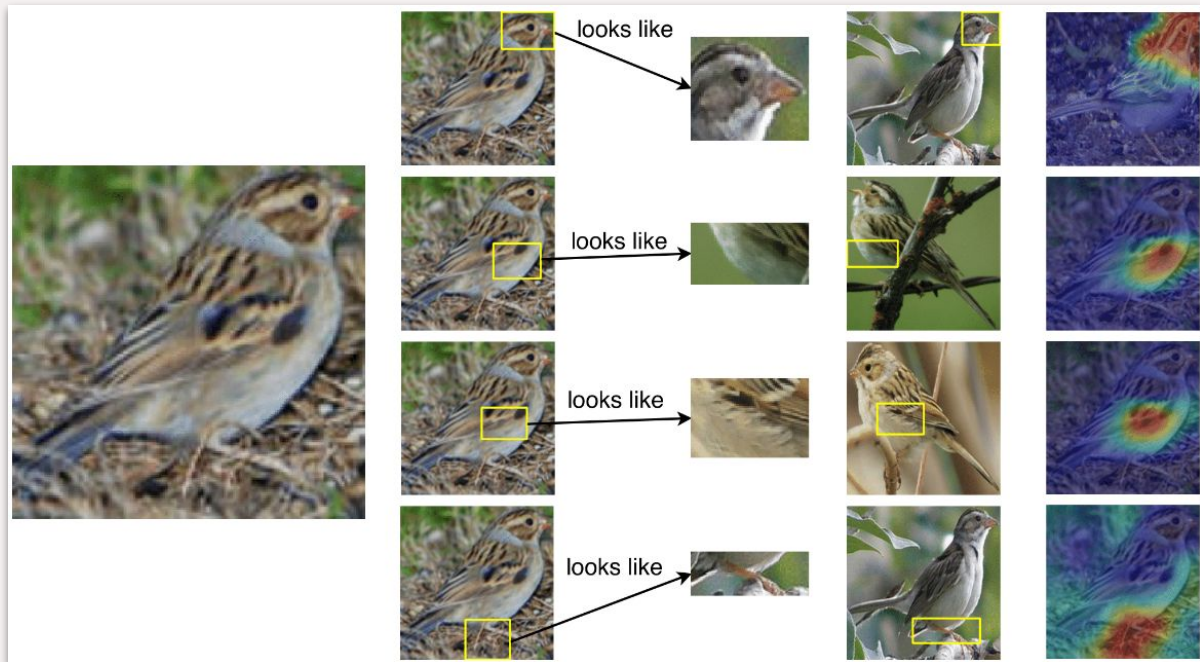
Injecting interpretable units via concept bottlenecks.

Advantage: enables manual intervention.

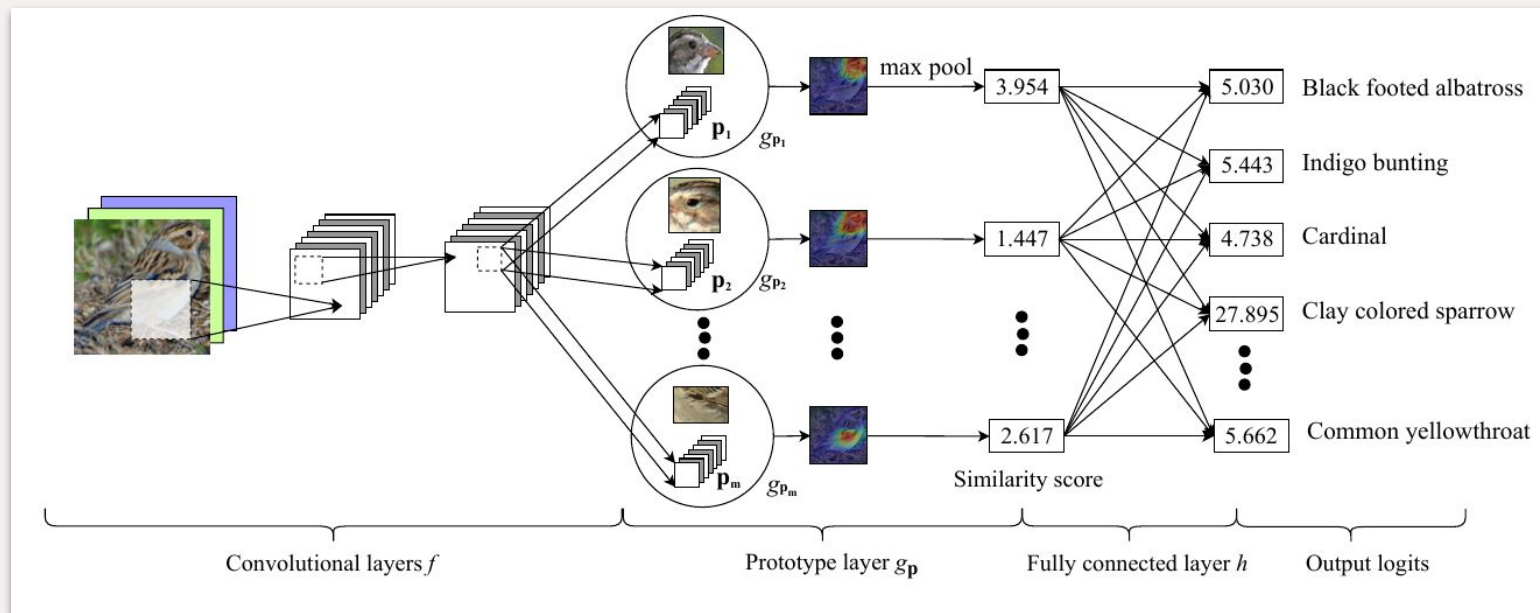


Case-based reasoning with prototype networks.

Idea: find prototypical parts and combine evidence into output.



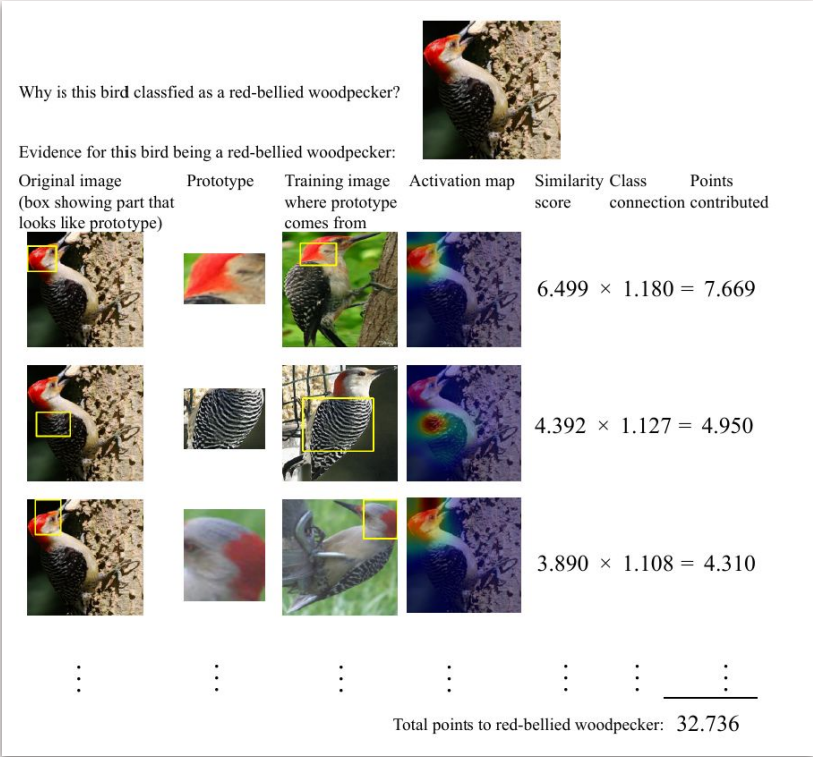
Case-based reasoning with prototype networks.



- Learn embeddings that link the representation with the task of interest (i.e. the predictions).
- Associate the embedding with intermediate concepts of interest (e.g. object parts).
- Produced explanations reflect the decision-making process

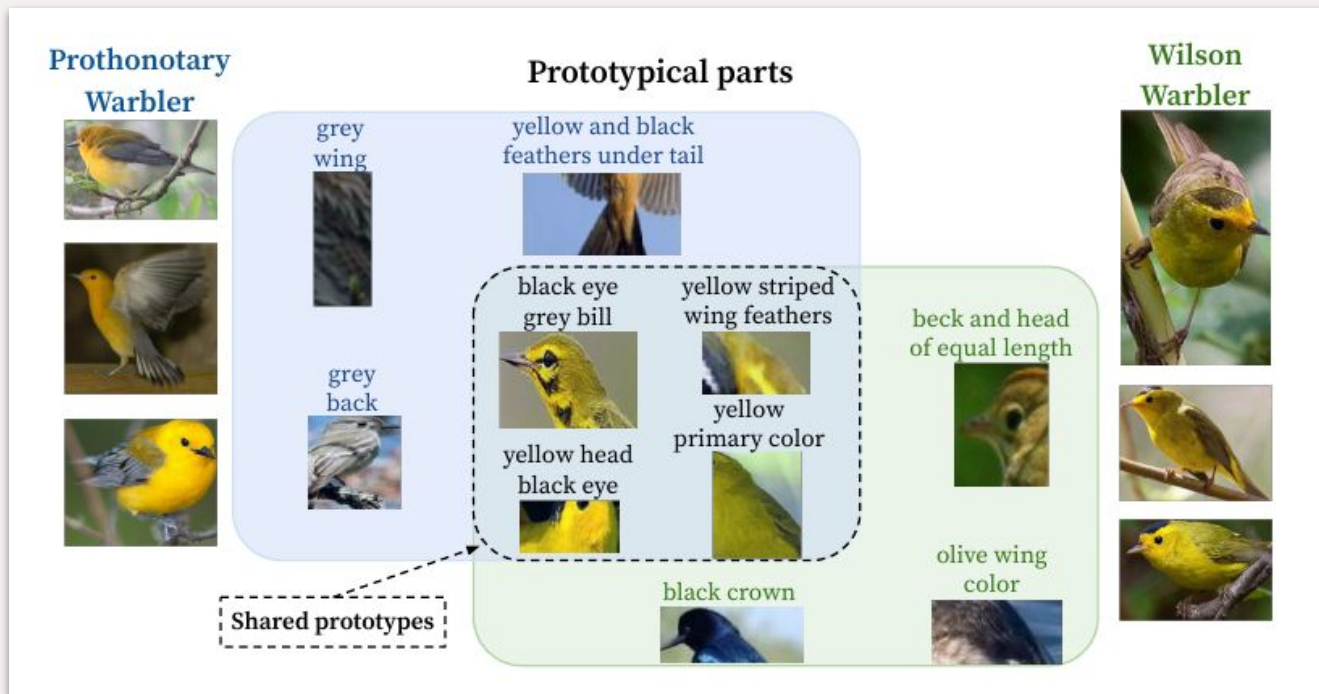
Case-based reasoning with prototype networks.

Goal: produced explanations reflect the decision-making process.



Case-based reasoning with prototype networks.

Goal: Addressing scalability by sharing prototypes across classes.



This chapter...

Post-hoc semantic mapping of neuron activations

What does a neuron do?

Inherently interpretable methods

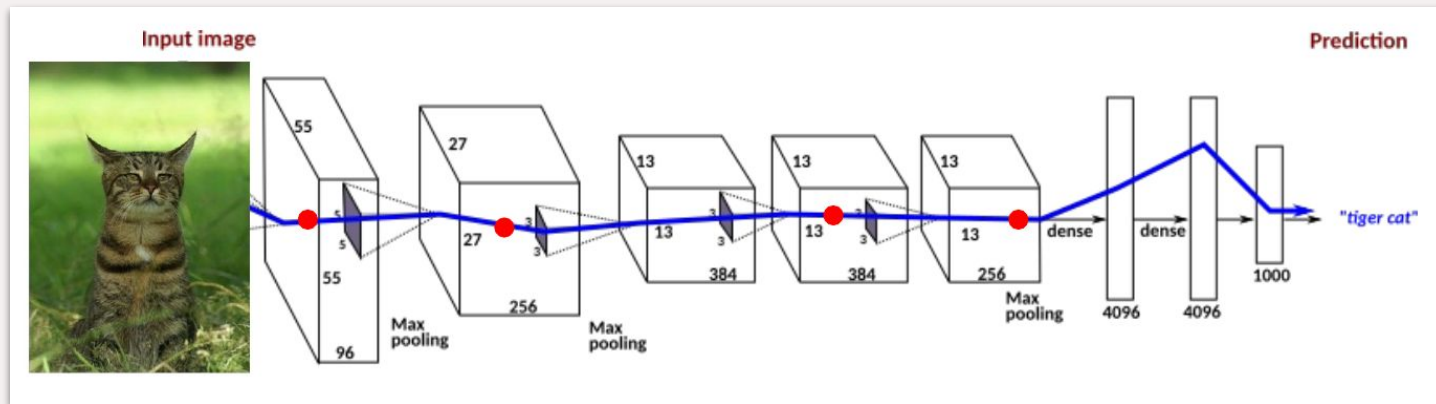
Can we make neurons understandable?

Evaluation protocols

When do we understand neurons?

Ablating the effect of relevant units/neurons in the model.

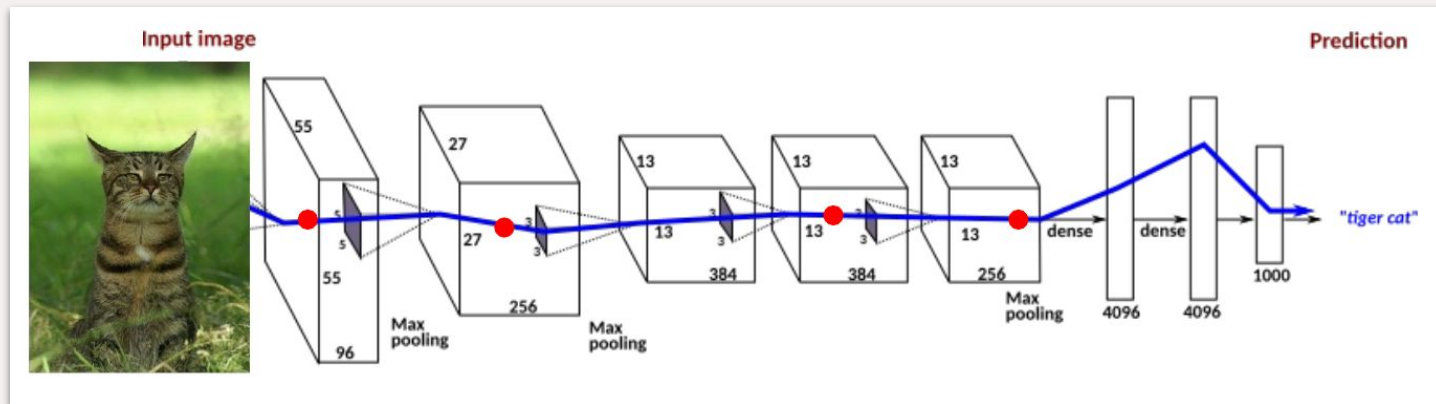
Goal: Assessing the fidelity of the units on the decision-making process.



- Define a mask
- Define a baseline

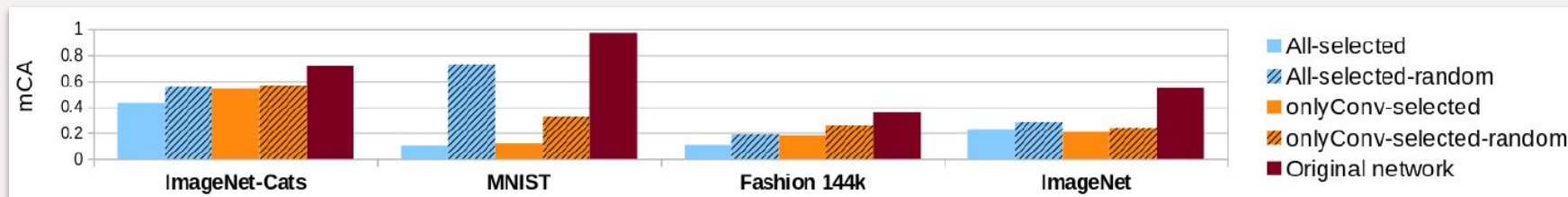
Ablating the effect of relevant units/neurons in the model.

Goal: Assessing the fidelity of the units on the decision-making process.



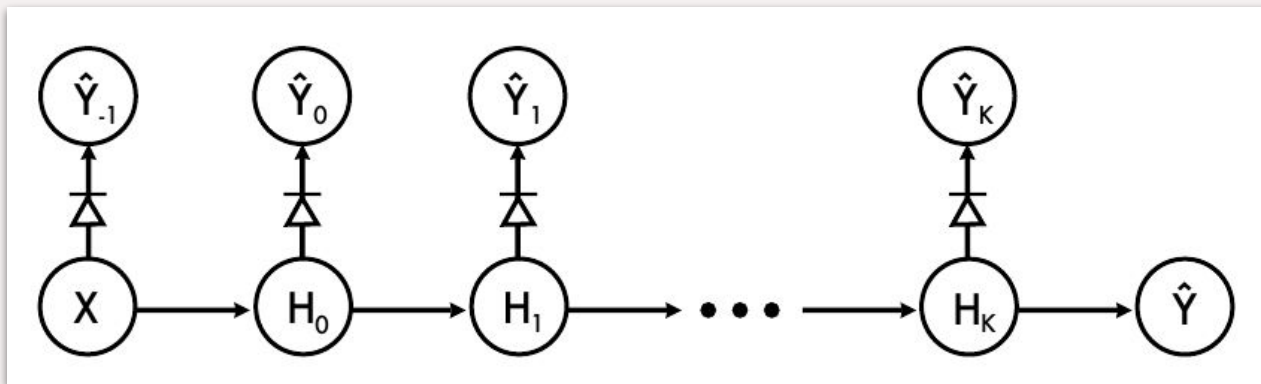
- Define a mask
- Define a baseline

... and measure the effect.



Analyzing the predictive signal of neuron activations

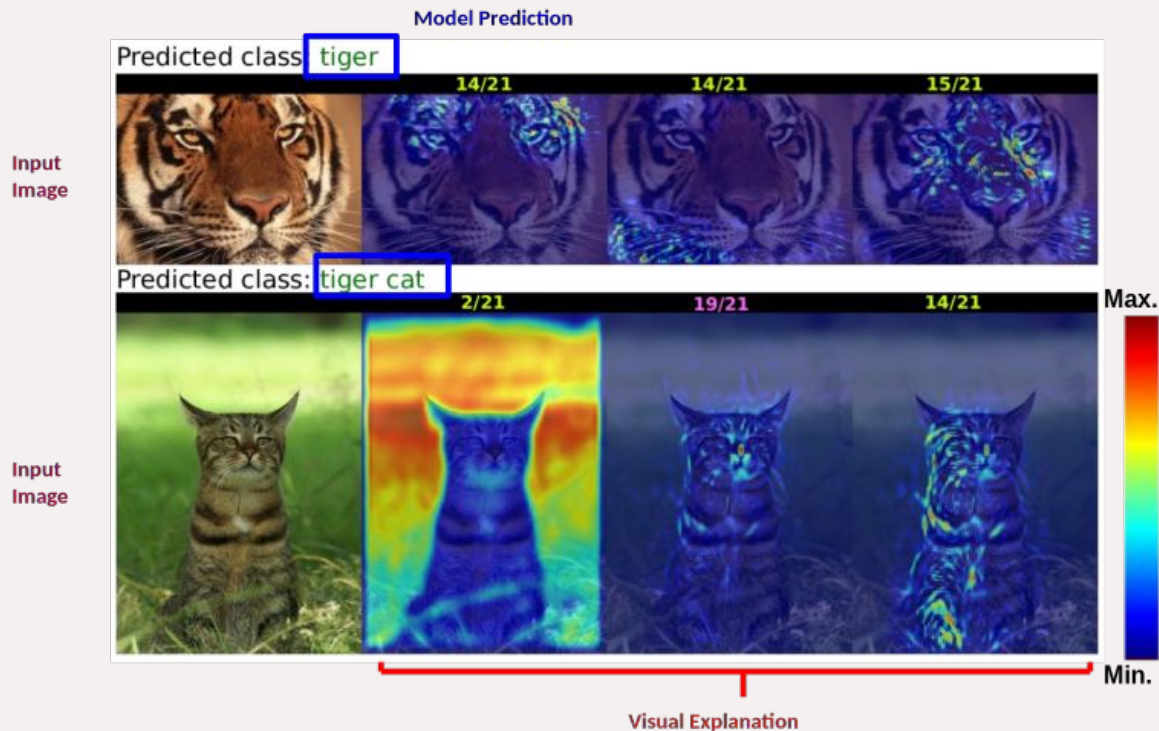
Idea: Use simple classifier (probes) to inspect the representation



- Targeted probing procedure
- Sparser analysis

(Proxy) Evaluation of derived explanations.

Idea: Evaluate the attribution maps produced from the relevant neurons.

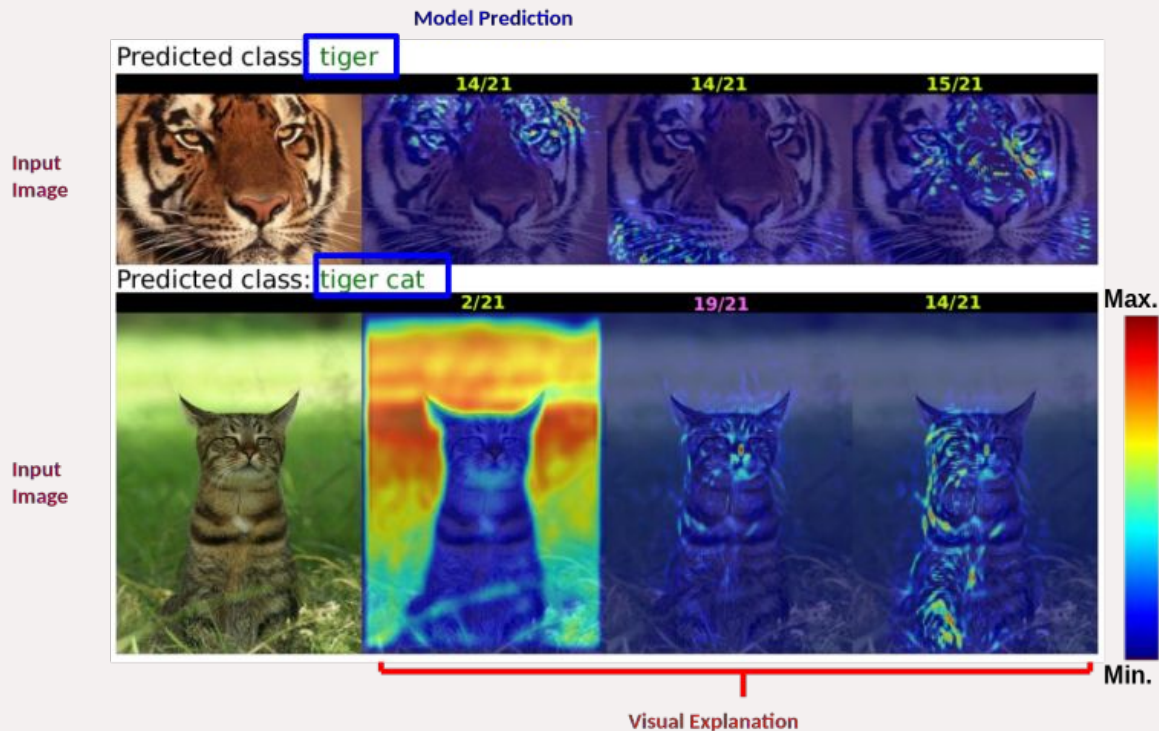


How?

- Produce an attribution map from the interpreted neurons.
- Measure performance following an attribution evaluation protocol.

(Proxy) Evaluation of derived explanations.

Idea: Evaluate the attribution maps produced from the relevant neurons.



How?

- Produce an attribution map from the interpreted neurons.
- Measure performance following an attribution evaluation protocol.

but...

- Inherit the weaknesses of those protocols

Coming up...

Early Interpretability Methods

José Oramas

From Classical to Mechanistic Interpretability

Ward Gauderis, José Oramas & Thomas Dooms

From Mechanistic to Compositional Interpretability

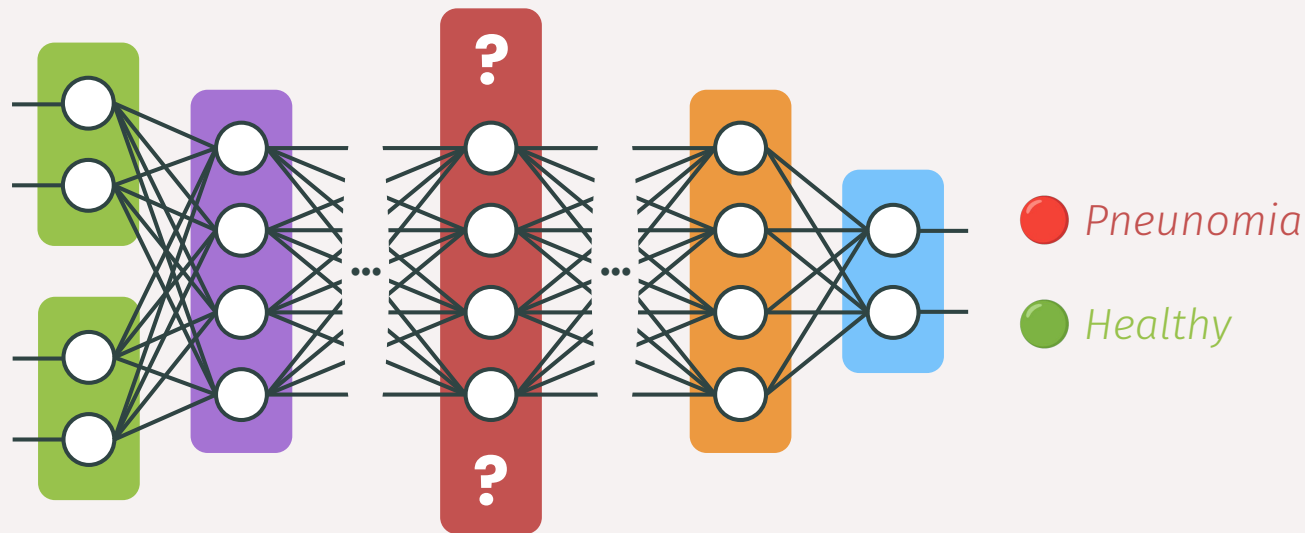
Ward Gauderis & Thomas Dooms

AIMLAI @ ECML PKDD 2026
September 7th 2026

From Classical to Mechanistic Interpretability

Ward Gauderis · José Oramas · Thomas Dooms

To control, predict, and learn from neural networks, we have to understand their *internal* working.



Monitor & steer

Align & debug

Improve design

Extract knowledge

This chapter...

Mechanistic foundations

Ward Gauderis

Methods & Models

José Oramas & Thomas Dooms

Limitations & Open Problems

Thomas Dooms

Mechanistic interpretability *reverse-engineers* learned representations into human concepts and algorithms.

Question

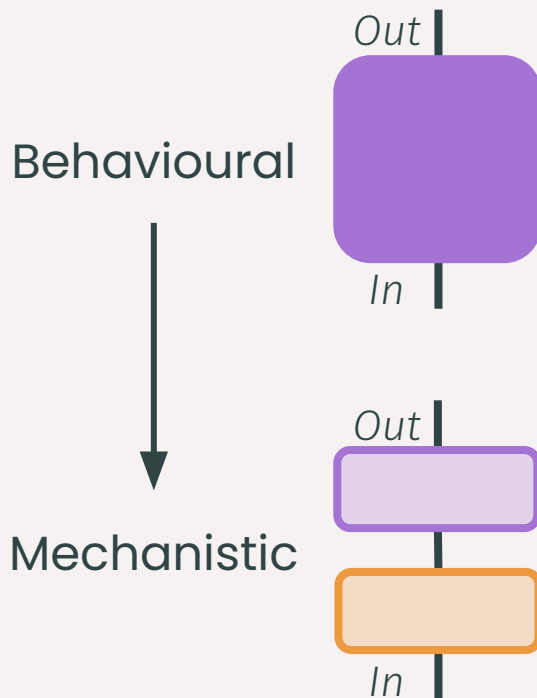
How can a model solve a **general** class of problems, not just one instance?

Premise

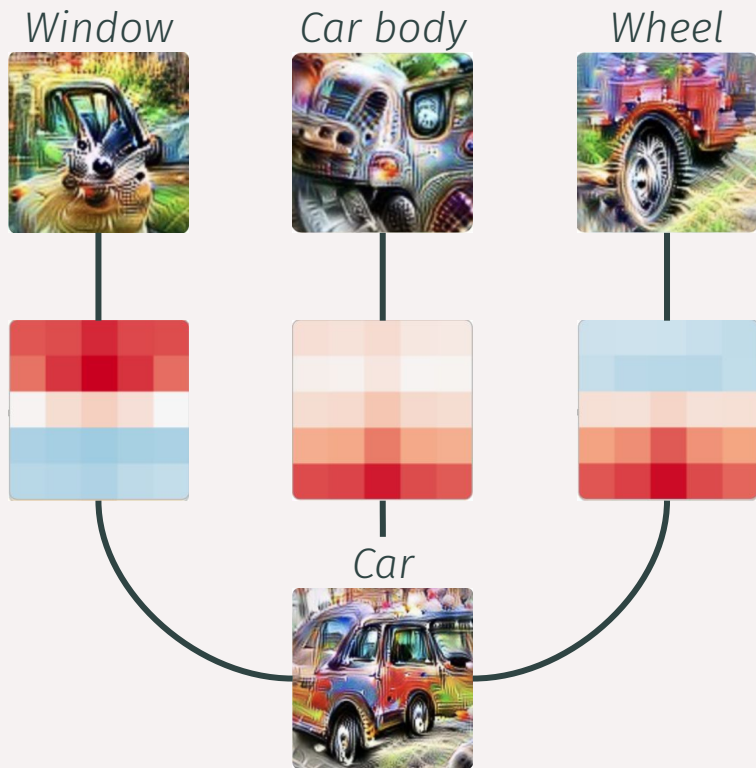
Generalisation in neural networks arises from shared computational **mechanisms**.

Scope

Any research that describes the internals of a model, including its **activations or weights**.



Axis 1. Networks are *features*, wired into *circuits*. Both can recur across models and tasks.



1. Object of study

Features

What input properties are encoded and how?

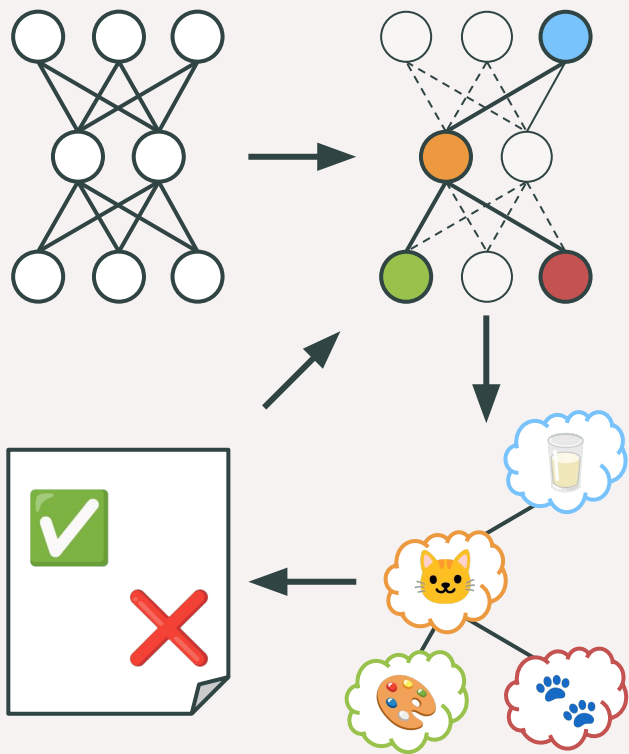
Circuits

How do features compute and interact?

Universality

Are these objects learned across models?

Axis 2. Decompose, hypothesise, and test. Reverse-engineering follows the *scientific method*.



1. Object of study

Features

Circuits

Universality

2. Process stage

Decomposition

What are the relevant components?

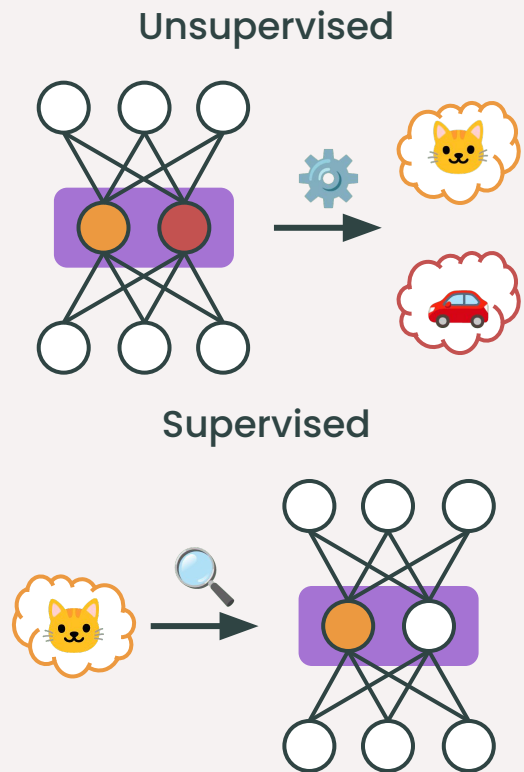
Description

What does each component do?

Validation

Is a components description accurate?

Axis 3. Either we ask what a component means, or we identify where a concept lives.



1. Object of study

Features

Circuits

Universality

2. Process stage

Decomposition

Description

Validation

3. Direction of search

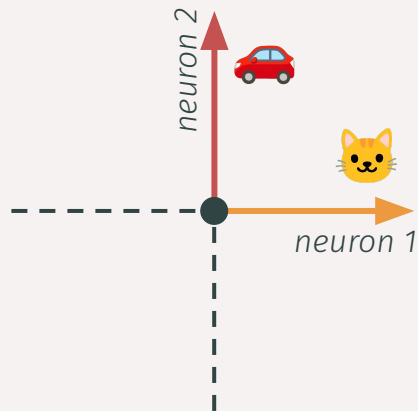
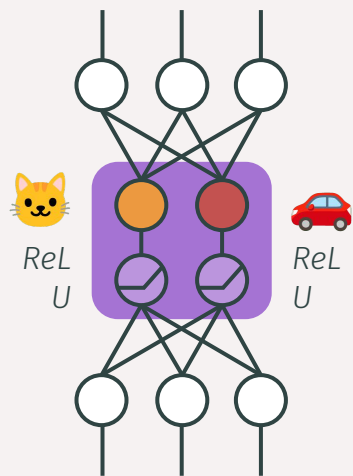
Bottom-up

Given a component, what is its role?

Top-down

Given a role, where is it represented?

We want features to align with human concepts.
Neurons form a *privileged* basis, but are *polysemantic*.

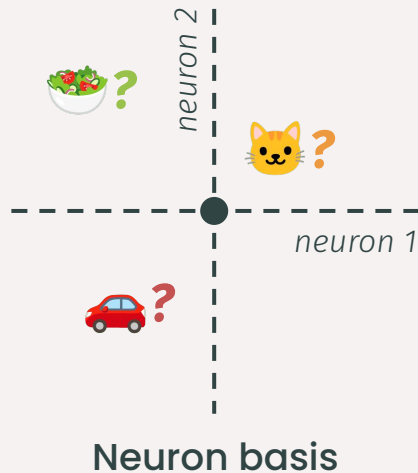
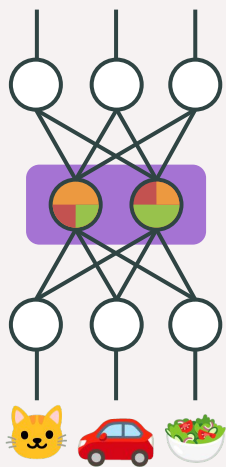


Neuron basis

If neurons are features...

A **monosemantic** neuron activates for a single, coherent concept.

We want features to align with human concepts.
Neurons form a *privileged* basis, but are *polysemantic*.



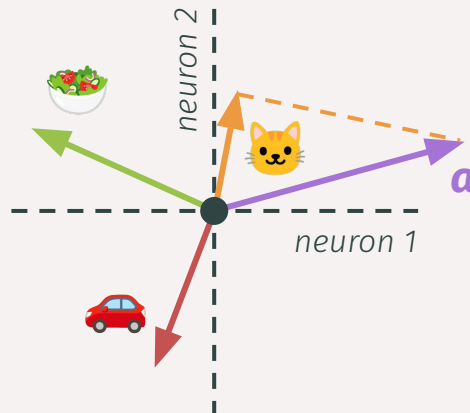
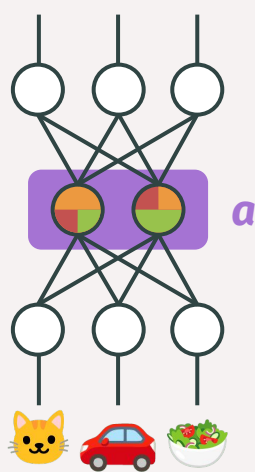
If neurons are features...

A **monosemantic** neuron activates for a single, coherent concept.

...But empirically they are not.

A **polysemantic** neuron activates for multiple unrelated concepts.

Under the *Linear Representation Hypothesis (LRH)*, features are linear directions in activation space.



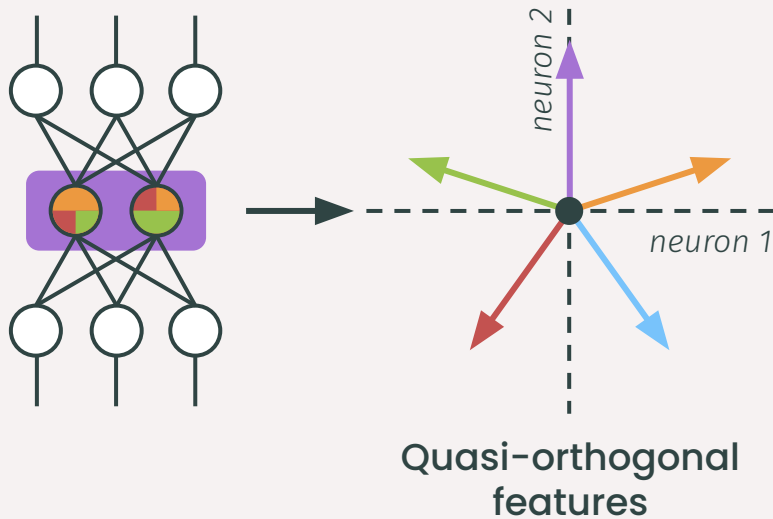
Feature presence is linear projection.

$$a_{\text{cat}} \approx \langle d_{\text{cat}} | a \rangle$$

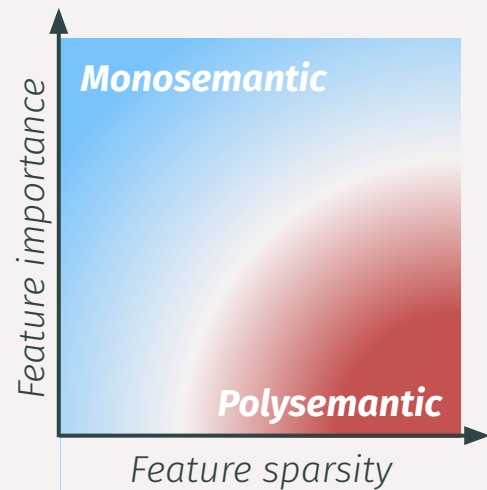
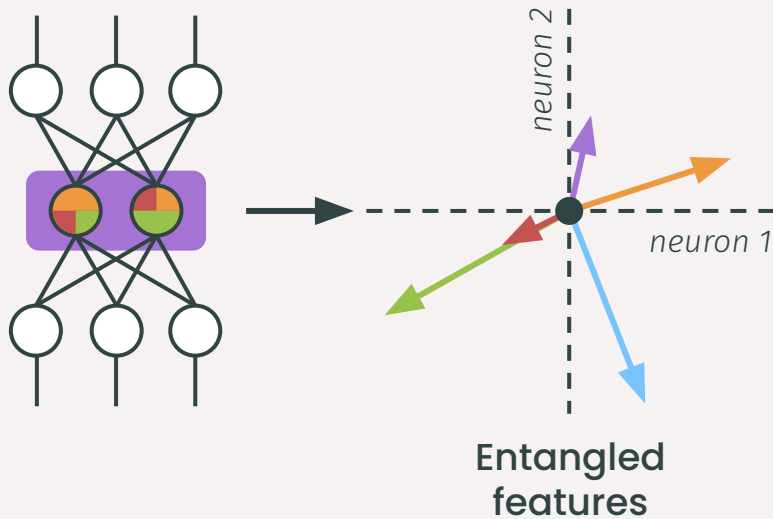
Activations are linear combinations.

$$a \approx a_{\text{cat}} \cdot \text{cat} + a_{\text{car}} \cdot \text{car} + a_{\text{salad}} \cdot \text{salad}$$

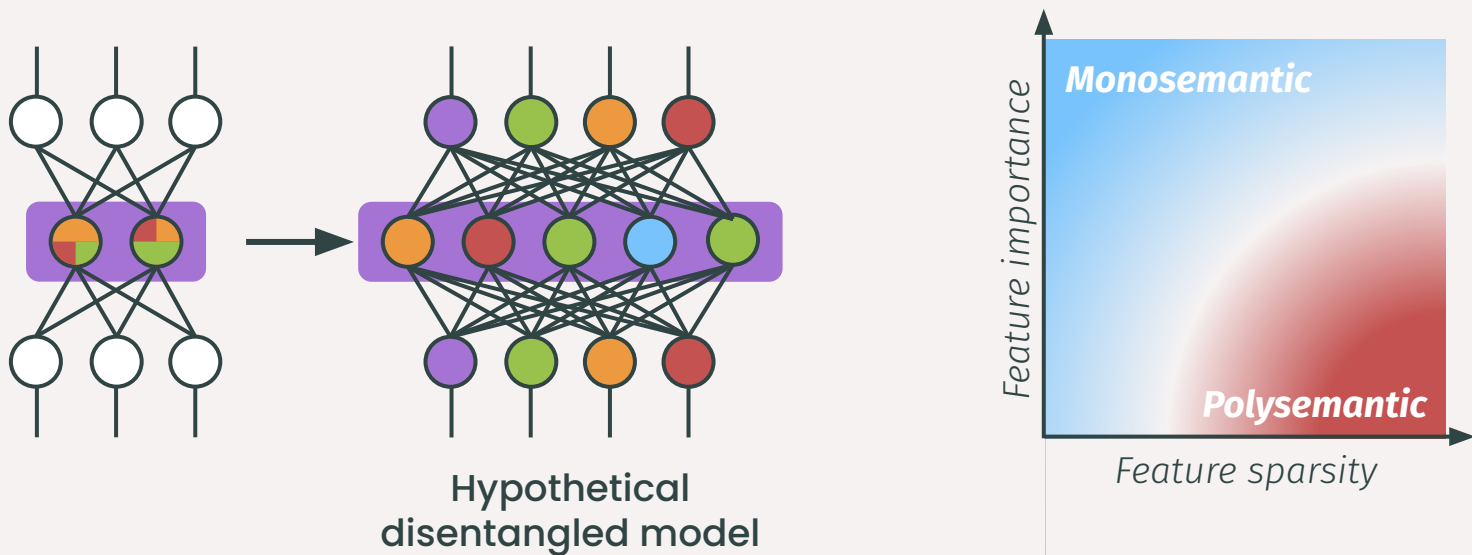
Under the LRH, *superposition* causes polysematicity.
Activations pack more concepts than neurons.



Under the LRH, *superposition* causes polysematicity.
Activations pack more concepts than neurons.



Under the LRH, *superposition* causes polysematicity.
Activations pack more concepts than neurons.



This chapter...

Mechanistic foundations

Ward Gauderis

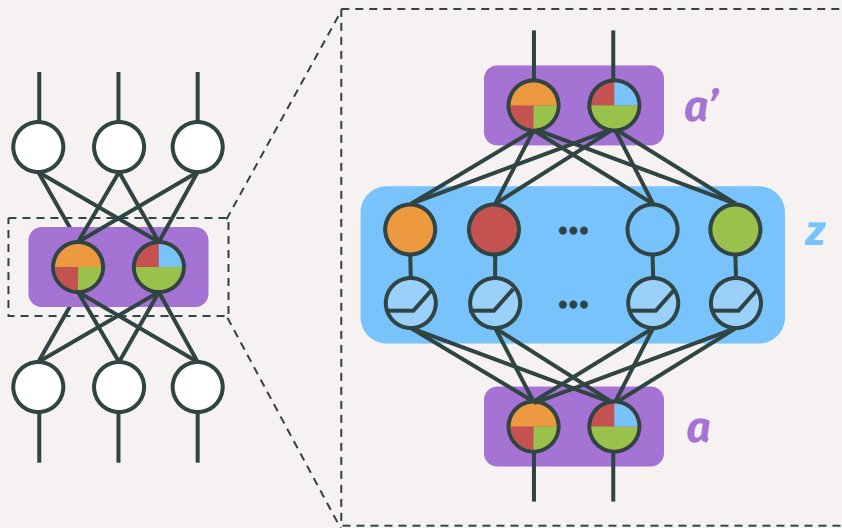
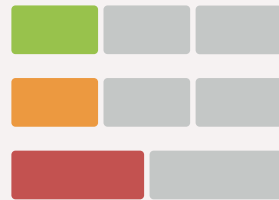
Methods & Models

Ward Gauderis & José Oramas & Thomas Doms

Limits & Open Problems

Thomas Doms

Sparse autoencoders learn an overcomplete dictionary of feature directions without labels.



Dictionary learning finds sparse features.

$$\mathbf{a} \approx D\mathbf{z}, \quad \|\mathbf{z}\|_0 \leq k$$

An encoder replaces the search for $\mathbf{z}$.

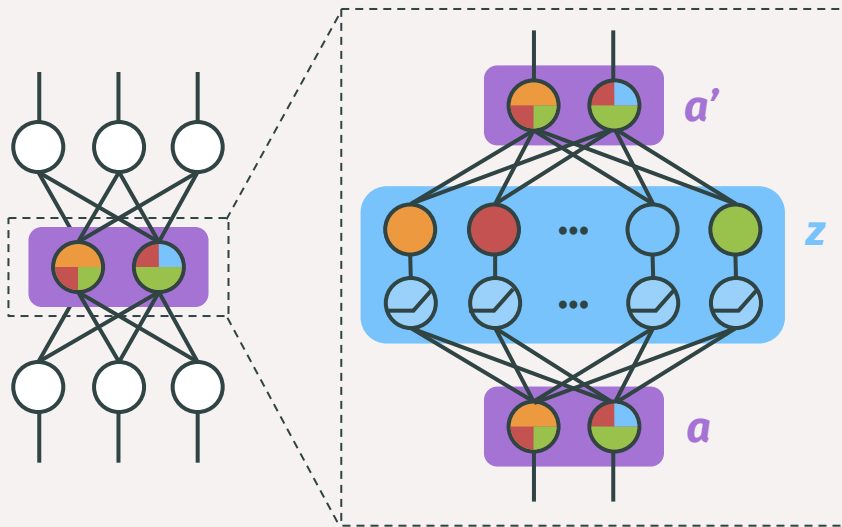
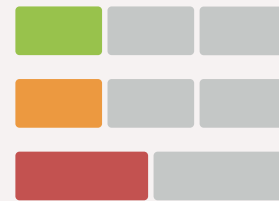
$$\mathbf{z} = \text{ReLU}(E\mathbf{a} + e)$$

$$\mathbf{a}' = D\mathbf{z} + d$$

The loss trades reconstruction and sparsity.

$$L(\mathbf{a}, \mathbf{a}') = \|\mathbf{a} - \mathbf{a}'\|_2^2 + \lambda \|\mathbf{z}\|_1$$

Sparsity is not a *proxy* for interpretability. Every feature needs a description and a test.



Features still have to be labelled.

- Decodability does not imply causality.

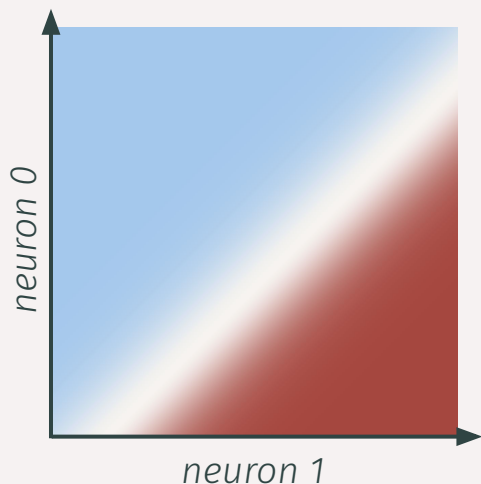
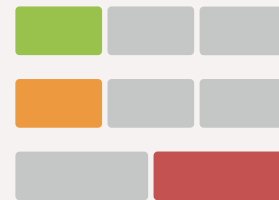
The decomposition is not unique.

- L1 penalises activation size, not count.
- Dictionary size affects feature granularity.
- Different seeds give different features.

For known concepts, supervision wins.

- Simpler baselines suffice downstream.

A *probe* uses labels to find the maximal separation between two classes.



Probing asks if a concept is linearly readable.

$$\mathbf{y} \approx \sigma(w\mathbf{a} + b), \mathbf{y} \in \{-1, 1\}$$

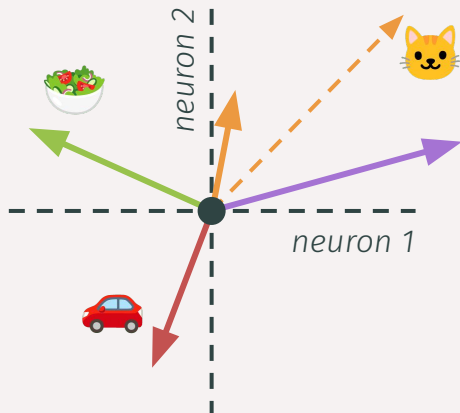
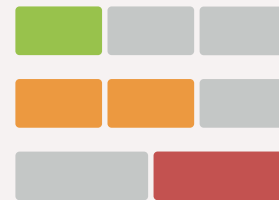
The weights carve a separating hyperplane.

$$w\mathbf{a} + b = 0$$

The loss rewards confidence in the true class.

$$L(\mathbf{y}, \mathbf{a}) = \text{softplus}(-\mathbf{y} (w\mathbf{a} + b))$$

A *probe* queries whether representations can *distinguish* a concept, not whether it is used.



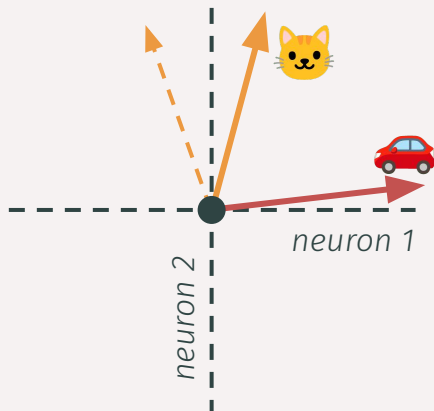
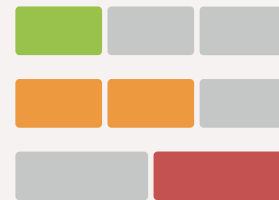
Probes can lie about concept presence.

- Decodability still does not imply causality.
- Will pick up on correlations.

Probes miss substructure.

- They answer no questions beyond presence.

A *probe* finds the maximal separation, it does not find the concept direction itself.



Probes can lie about concept presence.

- Decodability still does not imply causality.
- Will pick up on correlations.

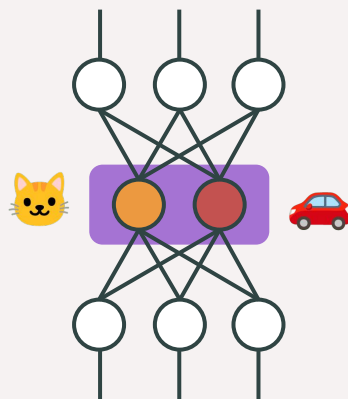
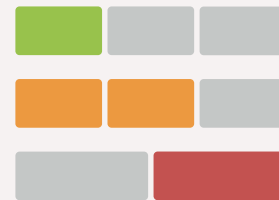
Probes miss substructure.

- They answer no questions beyond presence.

Probes find separation; not features.

- Directions don't align with features
- ... and should hence not be compared.

A *concept bottleneck model* builds a set of probes into the model.



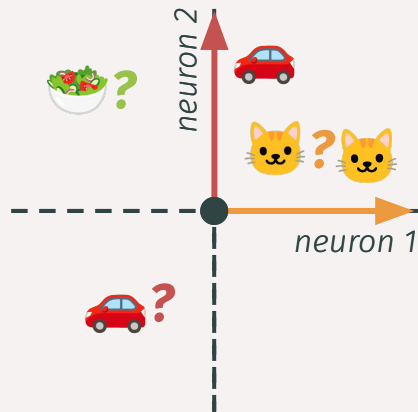
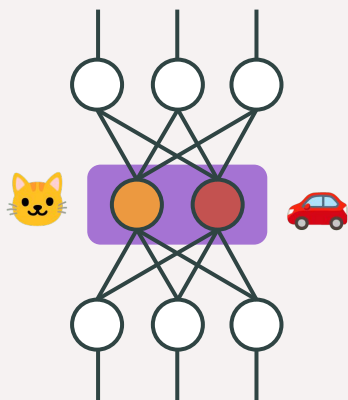
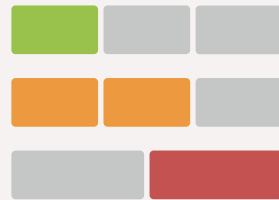
A bottleneck routes through concepts.

$$\hat{c} = g(a), \hat{y} = w\hat{c} + b$$

The loss supervises concepts and label jointly.

$$L = \ell(\hat{y}, y) + \lambda \sum \ell(\hat{c}_i, c_i)$$

Building in probes inherit probe problems. Concepts leak correlates and are incomplete.



Neuron basis

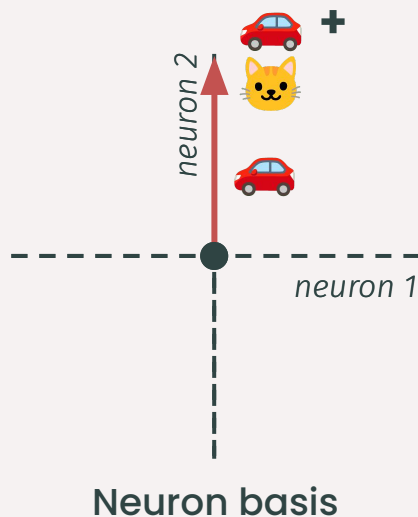
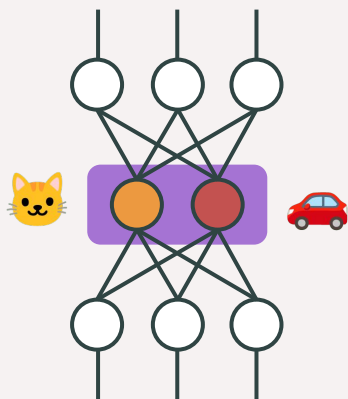
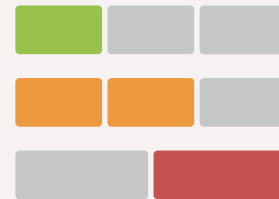
The concept set must be complete.

- Missing concepts cap accuracy.
- Residuals can interact weirdly.

Concept labels are expensive.

- Dense annotation per input
- 'label-free' outsources to an oracle.

Soft bottlenecks still allows *non-linear* signal to go through.



The concept set must be complete.

- Missing concepts cap accuracy.
- Residuals can interact weirdly.

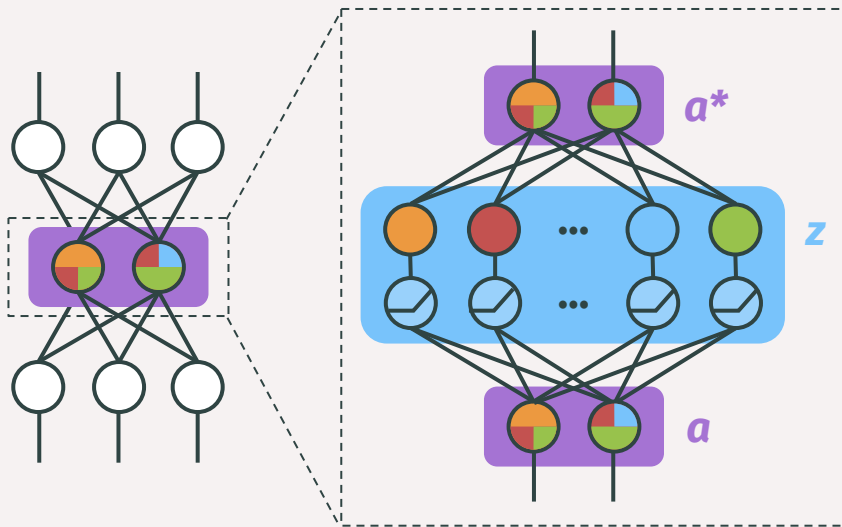
Concept labels are expensive.

- Dense annotation per input
- 'label-free' outsources to an oracle.

Soft concepts leak.

- Carry unlabelled signal past bottleneck.
- Non-linear signal is not penalized

Transcoders sparsify the computation, they decompose the model into *circuits*.



The dictionary is the same as an autoencoder.

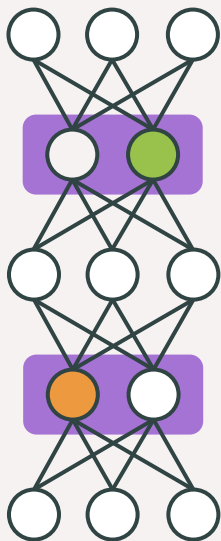
$$\mathbf{z} = \text{ReLU}(E\mathbf{a} + e)$$

$$\mathbf{a}^* = \text{MLP}(\mathbf{a}) \approx D\mathbf{z} + d$$

But it reconstructs the layer's output.

$$L = \|\mathbf{a}^* - D\mathbf{z} - d\|^2 + \lambda \|\mathbf{z}\|_1$$

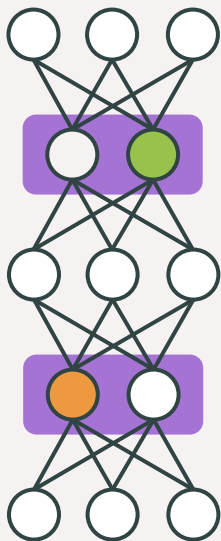
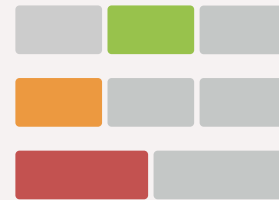
Understanding *circuits* allows tracing how a feature is formed.



Circuits describe feature interactions.

- Follow how features are computed.
- Test the proposed connections.

Understanding *circuits* allows tracing how a feature is formed.



Circuits describe feature interactions.

- Follow how features are computed.
- Test the proposed connections.

Approximation errors can compound.

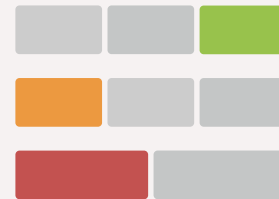
- Layer-wise fits can miss model behaviour.
- Error nodes retain unexplained signal.

SAE caveats still apply.

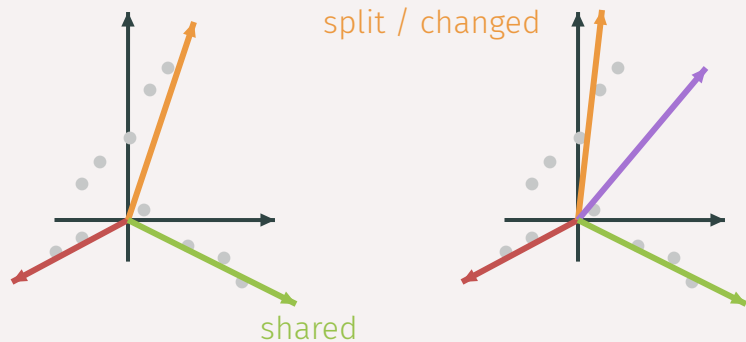
- Check width, sparsity and stability.
- Sparse activations $\neq$ sparse circuits

A decomposition need not be unique.

Identifiability requires assumptions.



Same activation data



SAE seed 1

SAE seed 2

Schematic dictionaries

Identifiability asks for uniqueness.

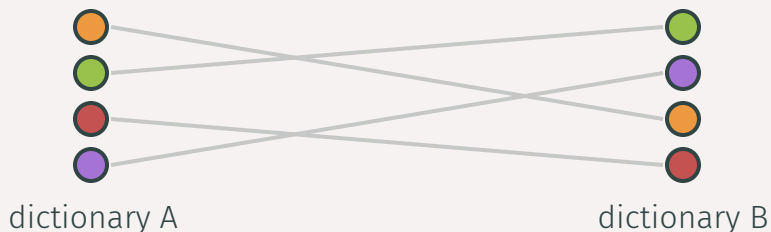
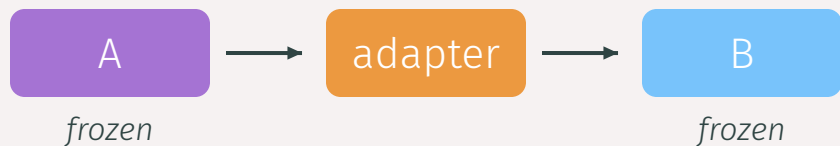
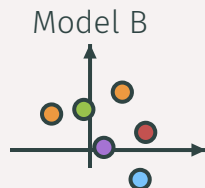
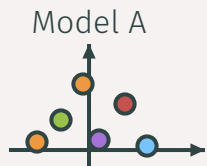
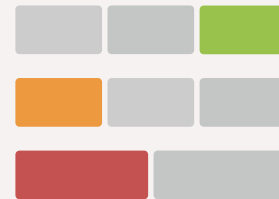
- Allow rescaling and permutation.

Stability tests the fitted solution.

- Vary seeds, data, width, and sparsity.

Universality depends on the comparison.

Features, geometry or function?



Do models organise inputs similarly?

- Compare the geometry of their activations.

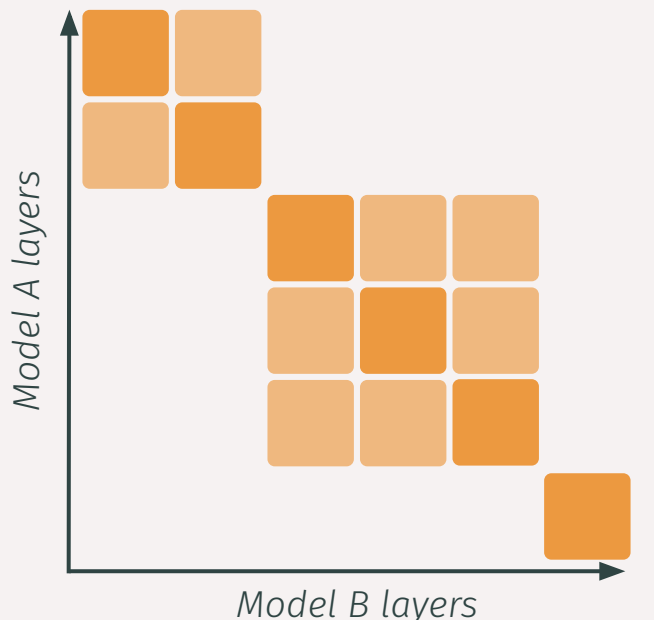
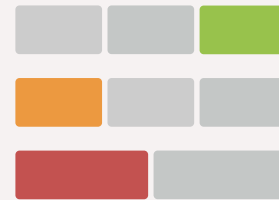
Can their representations work together?

- A learned adapter joins two frozen models.

Do they share candidate components?

- Compare features or fit a shared dictionary.

Centered kernel alignment compares layers. High similarity does not prove shared circuits.



Light to dark: low to high CKA (schematic)

Use matched input examples.

- Collect and centre activations; feature counts may differ.

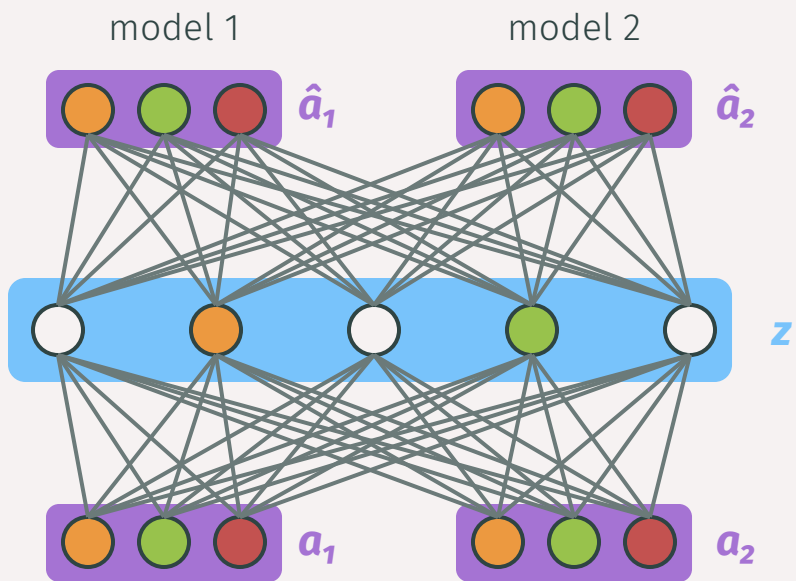
Read the layer-by-layer comparison.

- Darker cells indicate greater similarity; blocks reveal layer groupings.

Similar geometry leaves causal questions.

- Rotations and uniform scaling are ignored; shared roles need separate tests.

Crosscoders learn one shared code. Each site keeps its own decoder.



Paired activations for the same input

One code combines multiple sites.

$$\mathbf{z} = \text{ReLU}(E_1 \mathbf{a}_1 + E_2 \mathbf{a}_2 + e)$$

Each site has its own decoder.

$$\hat{\mathbf{a}}_1 = D_1 \mathbf{z} + d_1$$

$$\hat{\mathbf{a}}_2 = D_2 \mathbf{z} + d_2$$

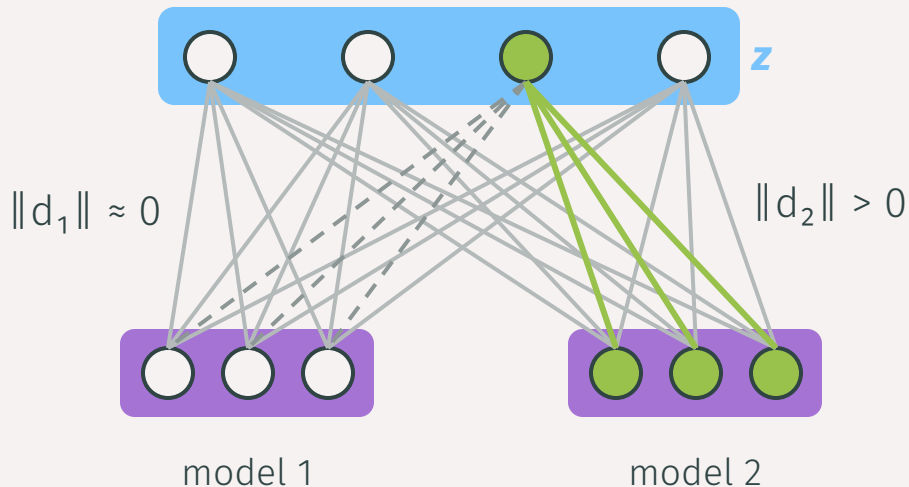
The loss trades reconstruction and sparsity.

$$L = \|\mathbf{a}_1 - \hat{\mathbf{a}}_1\|^2 + \|\mathbf{a}_2 - \hat{\mathbf{a}}_2\|^2 + \lambda R$$

Sparsity can create apparent differences. Model-only latents need targeted checks.



An apparent model-only latent



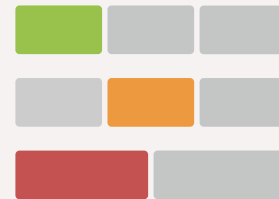
L1 can create false exclusivity.

- Shrinkage can erase useful decoder weights.

A shared code leaves causal questions.

- Joint encoding does not establish a circuit.

Examples suggest what a feature detects. An LLM turns the pattern into a *hypothesis*.



Examples + activations

The cat slept.

8

A sleepy cat purred.

9

The cat stretched.

7



Explainer LLM



“Feline-related text”

Schematic activations

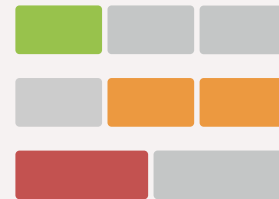
Input evidence suggests a description.

- Inspect activating contexts and token-level attribution.

Output evidence suggests a role.

- Vocabulary projections are clues; interventions test effects.

A label must survive new examples. *Counterexamples* reveal its scope.



Hypothesis: feline words

The lion roared.

Any activation? Yes

The lion roared.

Correct

The lion roared.

Incorrect

Detection checks the context.

- Does any token activate the feature?

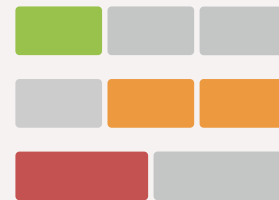
Fuzzing checks the marked tokens.

- Are these the positions where it activates?

Compare against a null model.

- Detection and fuzzing can score randomized models well.

Surprisal tests how much a label helps. It should help most on activating contexts.



$$\text{Gain} = \text{NLL}(\text{generic}) - \text{NLL}(\text{label})$$

Context	Generic	Label	Gain
cat	12	8	4
lion	13	10	3
dog	11	10.8	0.2

Score the description.

- Measure gain over a generic description.

Rank held-out contexts.

- Use the reduction to separate active from inactive examples.

The animal is a ...

Can you spot the *intruder*?

Coherence can be tested without a label.



Five held-out contexts

1. The ferry crossed the bay.
2. He took the tram home.
3. She caught the bus today.
4. The apple tasted sweet.
5. The train arrived.

Four positives, one negative.

- Choose the example that does not belong.

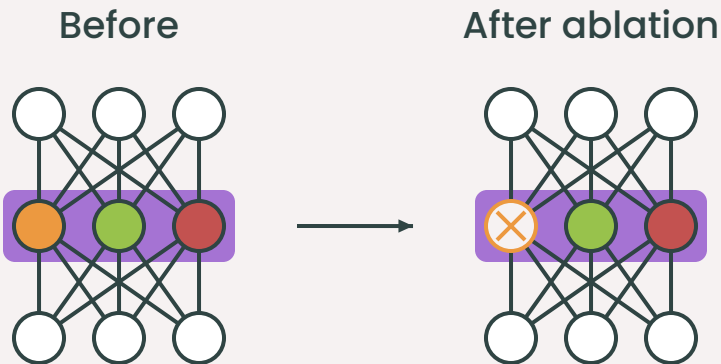
Score accuracy across examples.

- Chance accuracy is 20%.

Schematic intruder test

Causal claims predict intervention effects.

The setup defines the scope of the test.



Predicted effect



Specify the causal hypothesis.

- Set variables, basis, and intervention rules.

Distributed Alignment Search (DAS).

- Align to a fixed causal model.

Test predictions against controls.

- Report test scope and uncertainty.

Validation needs more than one setting. Known mechanisms and real tasks differ.



Known program



Model organism



Practical task



Known programs test recovery.

- Tracr provides a computation to recover.

Model organisms test failure detection.

- Can an audit find the hidden objective?

Real tasks test usefulness.

- Compare strong baselines for this question.

This chapter...

Mechanistic foundations

Ward Gauderis

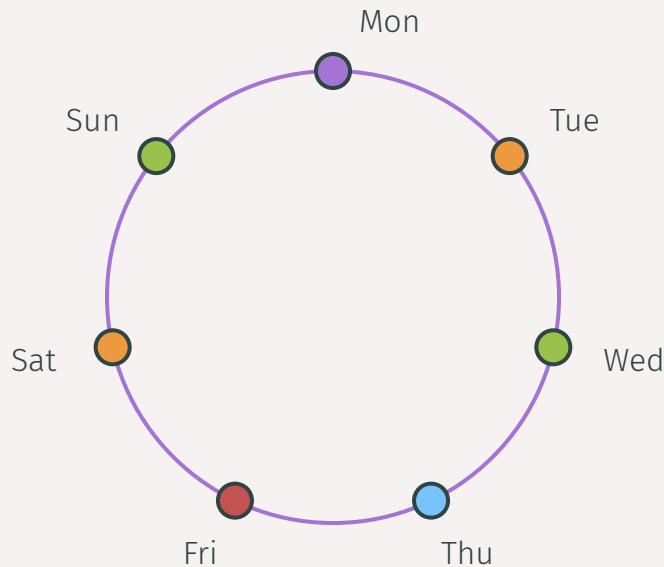
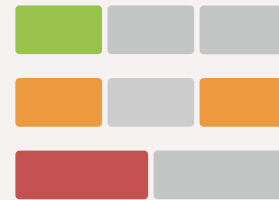
Methods & Models

José Oramas & Thomas Dooms

Limits & Open Problems

Thomas Dooms

Several directions can encode one concept. Interventions test their computational role.



One structured representation

Schematic weekday circle

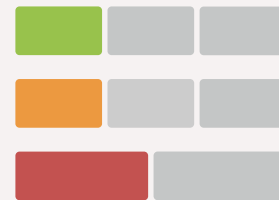
A feature can have internal geometry.

- Weekdays can trace a circle in a plane.

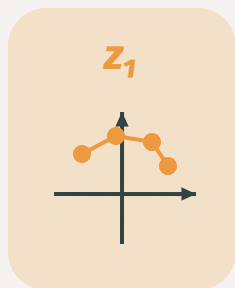
PCA alone does not identify a mechanism.

- Intervene on the subspace; test its role.

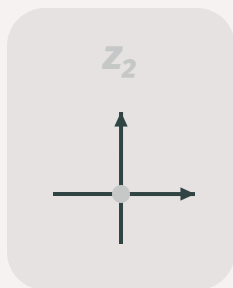
Block-sparse featurizers select vectors. Sparsity is imposed across whole blocks.



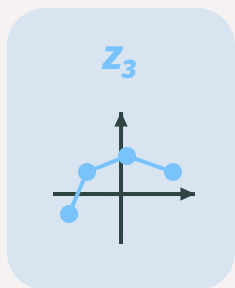
Select whole vector blocks



active



inactive



active

$$\hat{x} = D_1 z_1 + D_2 z_2 + D_3 z_3$$

Schematic coordinates across inputs

Select a few vector blocks.

- Top-k keeps the blocks with largest norms.

Keep structure within each block.

- Signed coordinates can vary together.

Blocks are a modelling assumption.

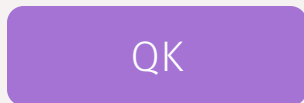
- A block need not match one concept.

Weights also admit a decomposition.

APD searches for reusable components.



Attention routes OV outputs

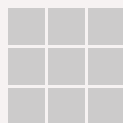


APD: additive weight components

$$\theta \approx \theta_1 + \theta_2 + \theta_3$$



active



inactive



active

QK and OV expose weight circuits.

- QK scores where to attend.
- OV maps what is written.

APD adds reusable components.

- Rebuild weights; use few pieces per input.

Interactions still matter.

- Parameters add; outputs need not.

APD depends on costly fitting and tuning. Its importance scores are approximations.



More components cost more to fit



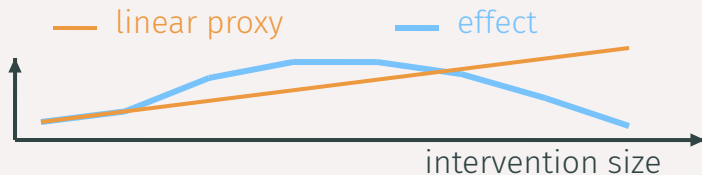
Fitting costs memory and computation.

- More parameter components add overhead.

Sparsity changes the decomposition.

- Tune top-k and simplicity penalties.

A local estimate can miss curvature



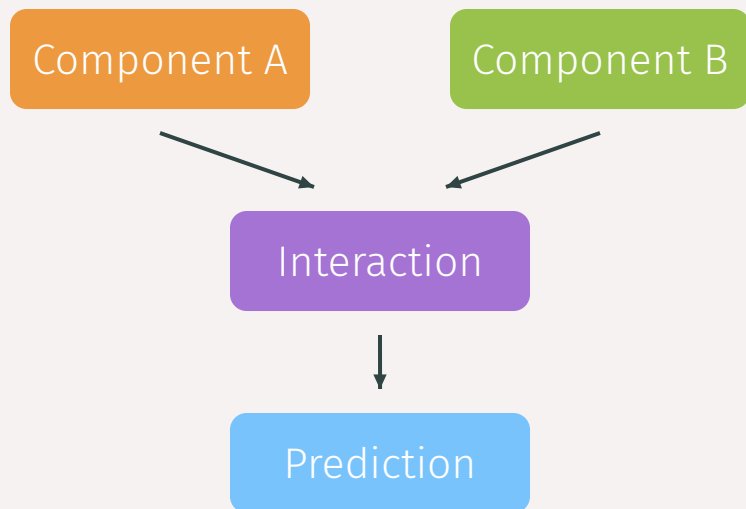
Attribution is a causal proxy.

- Gradients can misestimate ablation effects.

We can find and test the parts.

When do they explain the *computation*?

Parts and interactions



Account for parts and interactions.

- How do they combine into the computation?

Specify fidelity and what is missing.

- Which effects are preserved or unexplained?

Compare explanations for a purpose.

- Which helps answer the actual question?

Coming up...

Early Interpretability Methods

José Oramas

From Classical to Mechanistic Interpretability

Ward Gauderis, José Oramas & Thomas Dooms

From Mechanistic to Compositional Interpretability

Ward Gauderis & Thomas Dooms

AIMLAI @ ECML PKDD 2026
September 7th 2026

From Mechanistic to Compositional Interpretability

Ward Gauderis • Thomas Dooms

This chapter...

Interpretability Paradigms

Ward Gauderis

Compositional Interpretations

Ward Gauderis & Thomas Dooms

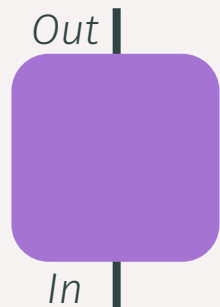
Architectures & Algorithms

Thomas Dooms

As AI adoption grows, regulation asks questions our interpretability tools *cannot* yet answer.

Post-hoc
interpretability

“What information was used to make this particular decision?”



Behavioural faithfulness

Decisions, but no mechanisms

As AI adoption grows, regulation asks questions our interpretability tools *cannot* yet answer.

Post-hoc interpretability

"What information was used to make this particular decision?"

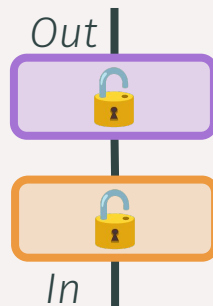


Behavioural faithfulness

Decisions, but no mechanisms

Intrinsic interpretability

"How do we design a model that explains all its behaviour?"



Inherent faithfulness

Guarantees, but constrained

As AI adoption grows, regulation asks questions our interpretability tools *cannot* yet answer.

Post-hoc interpretability

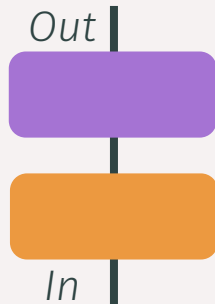
"What information was used to make this particular decision?"



Behavioural faithfulness
Decisions, but no mechanisms

Mechanistic interpretability

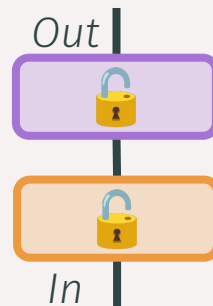
"How does the model solve this general class of problems?"



Explanatory faithfulness
Heuristic, but no foundations

Intrinsic interpretability

"How do we design a model that explains all its behaviour?"



Inherent faithfulness
Guarantees, but constrained

Interpretability Paradigms

Ward Gauderis

Compositional Interpretations

Ward Gauderis & Thomas Dooms

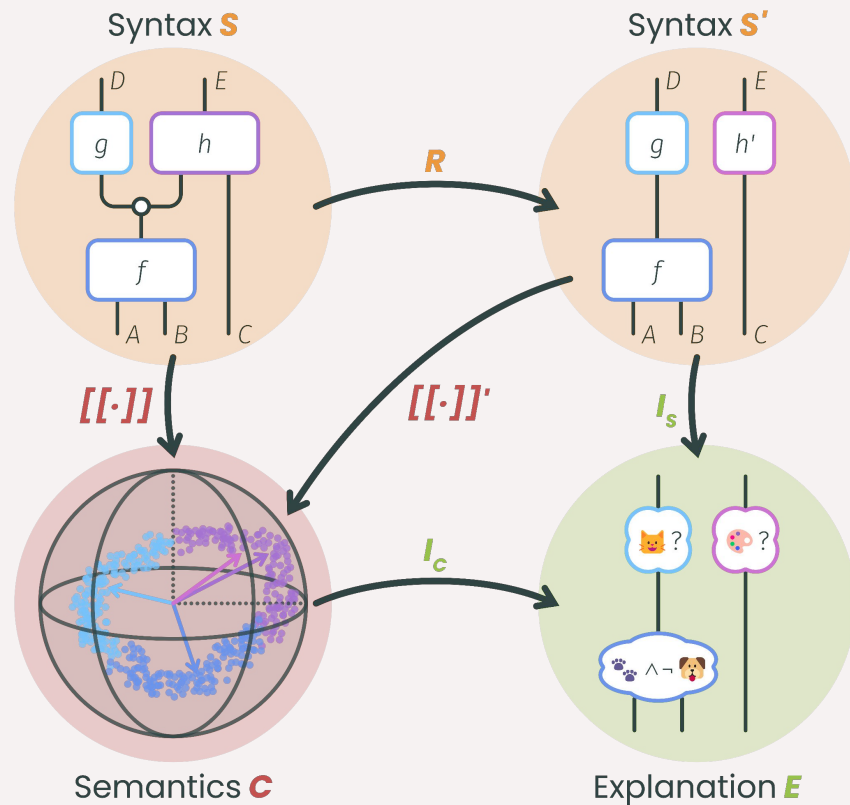
Architectures & Algorithms

Thomas Dooms

Compositional interpretability is a formal framework to evaluate, compare and improve explanations.

Two model-agnostic principles, formalised,

cast interpretability as **constrained optimisation**.

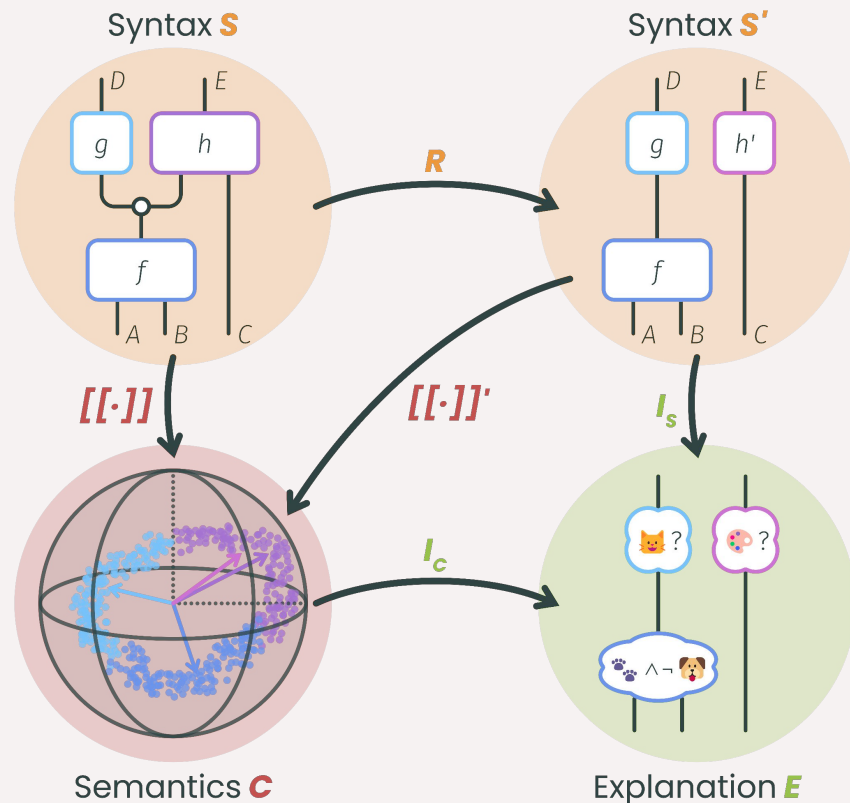


Compositional interpretability is a formal framework to evaluate, compare and improve explanations.

Two model-agnostic principles, formalised,

- **Compositionality**
 - Category Theory

cast interpretability as **constrained optimisation**.

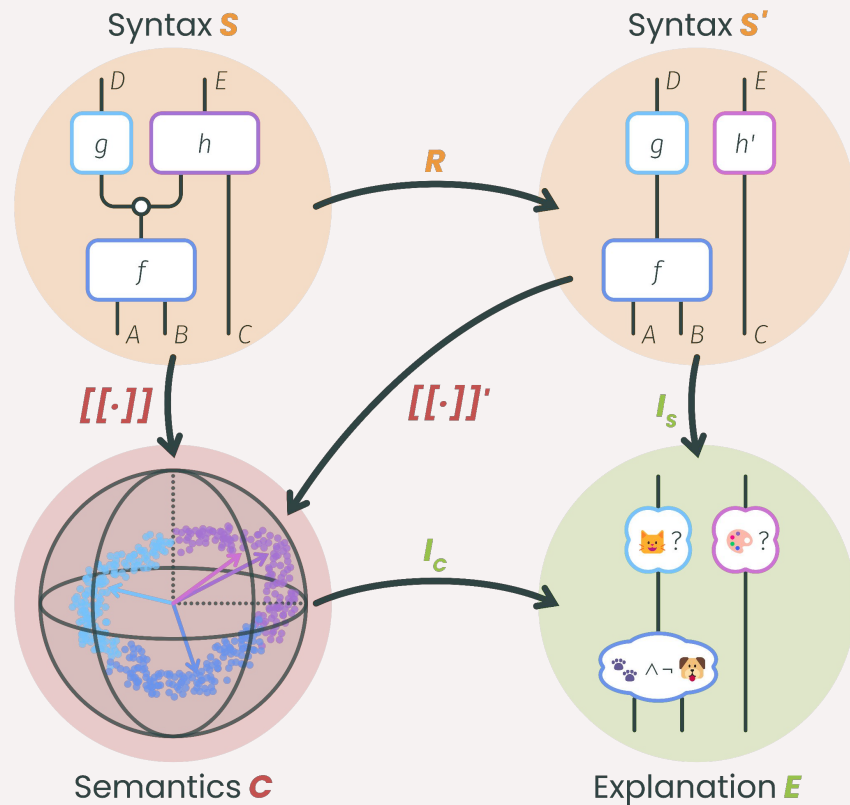


Compositional interpretability is a formal framework to evaluate, compare and improve explanations.

Two model-agnostic principles, formalised,

- **Compositionality**
 - Category Theory
- **Occam's Razor**
 - Minimum Description Length

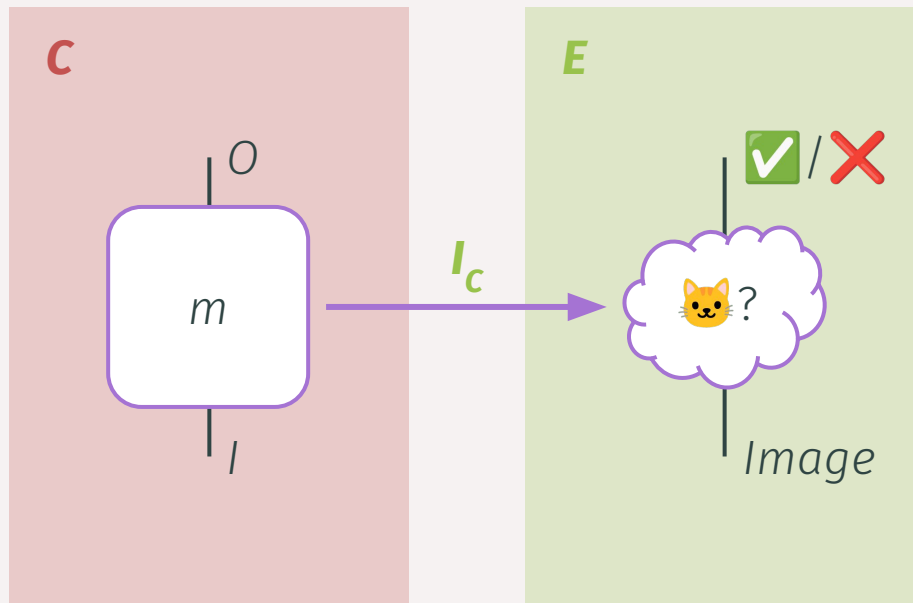
cast interpretability as **constrained optimisation**.



Coming up...

1. **What are compositional interpretations?**
2. What makes explanations good?
3. How do we find better explanations?

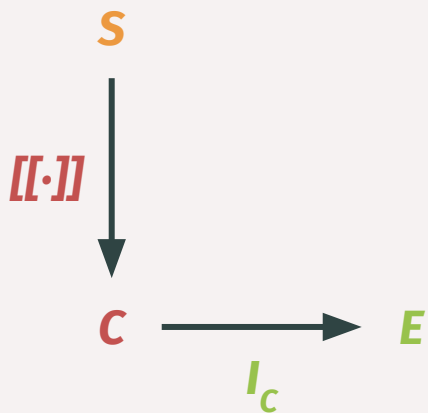
Post-hoc interpretations only map input-output behaviour to explanations.



C Semantics
 E Explanation

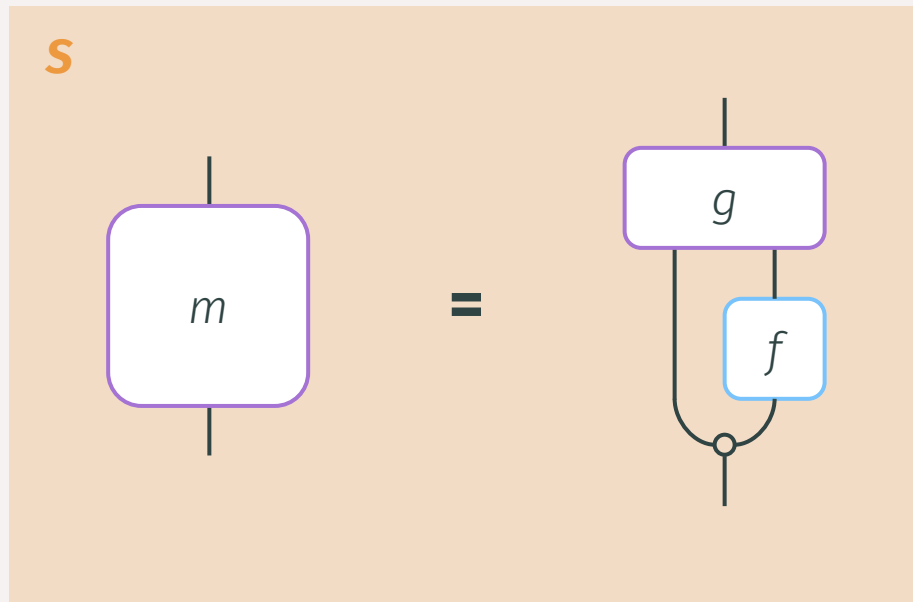
I_C Semantic interpretation

Compositional models abstract high-level syntax from low-level semantics.

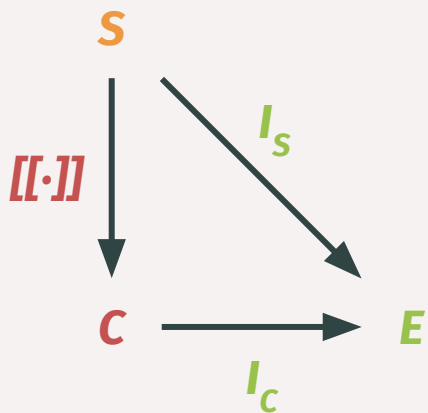


S Syntax
C Semantics
E Explanation

[[·]] Representation
I_C Semantic interpretation

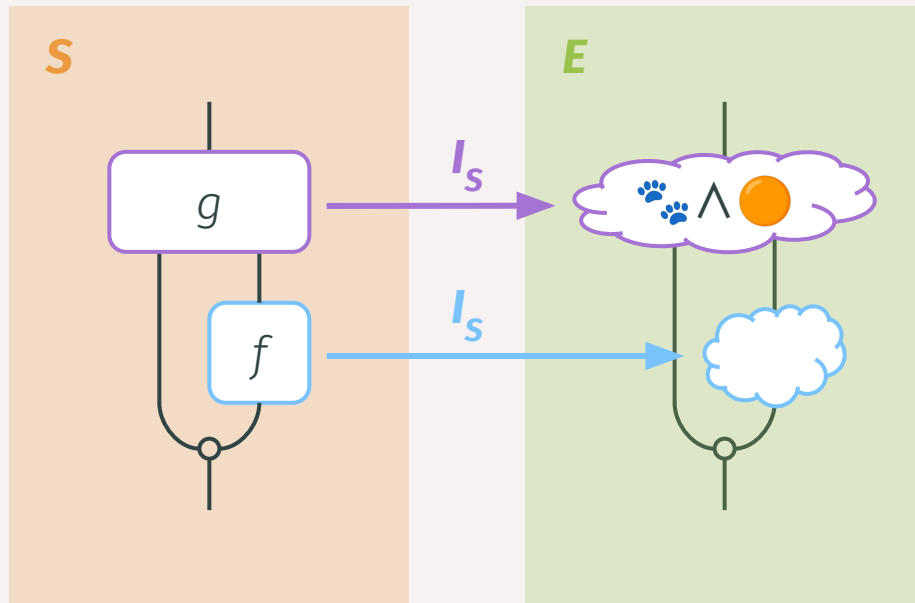


Compositional interpretations mechanistically explain how a model works, step-by-step.



S Syntax
 C Semantics
 E Explanation

$[[\cdot]]$ Representation
 I_c Semantic interpretation
 I_s Syntactic interpretation



Category theory and string diagrams are the mathematics of compositional structure.

A **compositional model** $M = (D, G, \llbracket - \rrbracket, C)$ consists of:

- G , a signature inducing $S = \text{Free}(G)$, an abstract syntactic category
- $D \in \text{Hom} \square(I, O)$, a composite morphism in S
- C , a concrete semantic category
- $\llbracket - \rrbracket : S \rightarrow C$, a representation functor

A **compositional interpretation** $I(I_a, I_c, I_c^*, H)$ for M into H consists of:

- $I_a : S \rightarrow H$, an abstract interpretation functor
- $I_c : C \rightarrow H$, a concrete interpretation functor s.t.

$$I_a = I_c \circ \llbracket - \rrbracket$$

- $I_c^* : H \rightarrow C$, the left adjoint to I_c ($I_c^* \dashv I_c$) defined by the natural isomorphism

$$\text{Hom}_c(I^*(h), c) \cong \text{Hom} \square(h, I_c(c))$$

A **Category** C consists of:

- $ob(C)$, a class of objects
- $hom_c(A, B)$, a class of morphisms for $A, B \in ob(C)$
- $id_a \in hom_c(A, A)$, an identity morphism for $A \in ob(C)$
- $\circ : hom_c(B, C) \times hom_c(A, B) \rightarrow hom_c(A, C)$, an associative operator that respects identity morphisms

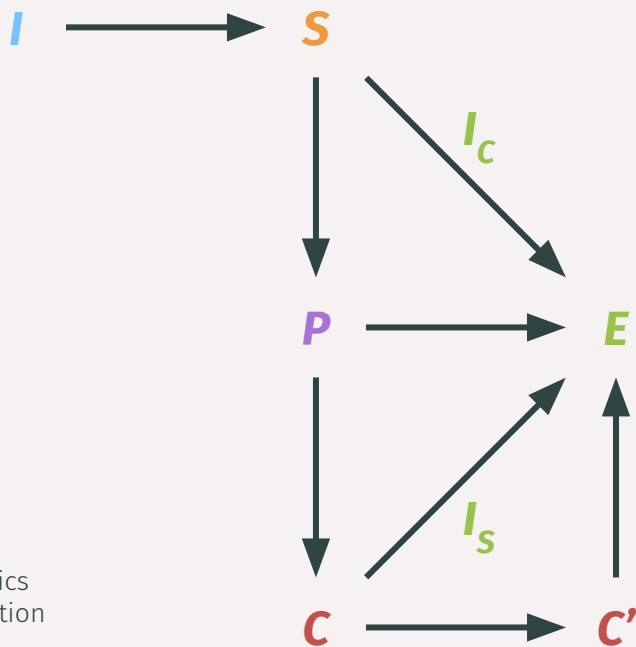
A **string diagram** is a visual representation of a composite morphism in a **symmetric monoidal category** C equipped with:

- $I \in ob(C)$, the monoidal unit object
- $\otimes : C \times C \rightarrow C$, a bifunctor for the tensor product
- Natural isomorphisms for the associator, symmetry, left and right unitors (left as an exercise to the reader)

Examples include:

- **CartSp** of Euclidean spaces and continuous functions
- **Stoch** of finite sets and probability channels
- **Quant** of finite-dimensional Hilbert spaces and CP-maps

This language lets us formalise and structurally compare interpretability methods.



S Syntax
C Semantics
E Explanation

I_c Semantic interpretation
 I_s Syntactic interpretation

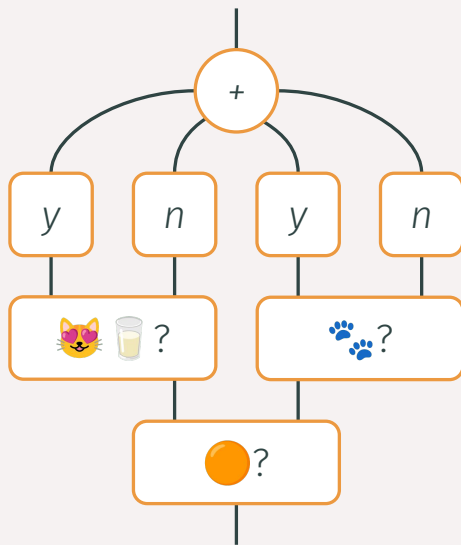
I Input-dependent structure

P Parameter space

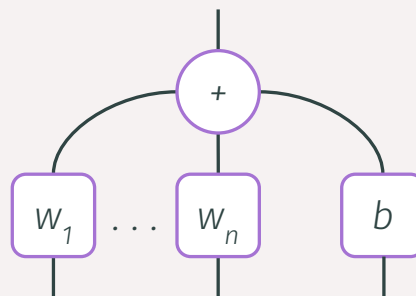
C' Proxy semantics

The objective of interpretability is to search for complete compositional interpretations.

Decision tree



Linear regression



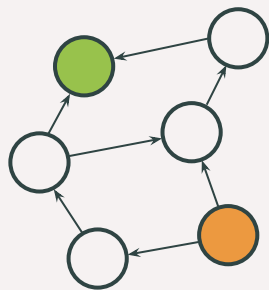
Intrinsically interpretable models have compositional interpretations.

Coming up...

1. What are compositional interpretations?
- 2. What makes explanations good?**
3. How do we find better explanations?

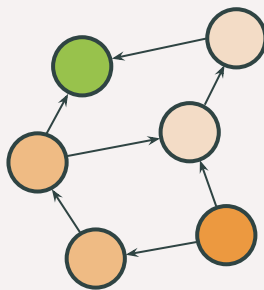
What makes a fastest path algorithm good?

Brute force



exact

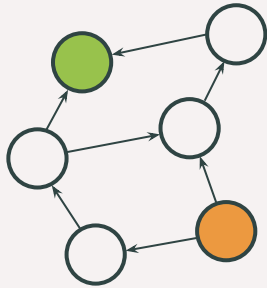
Dijkstra



exact

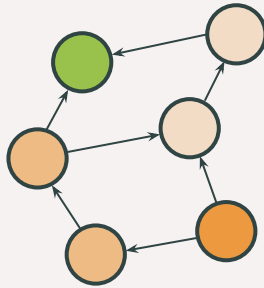
What makes a fastest path algorithm good?

Brute force



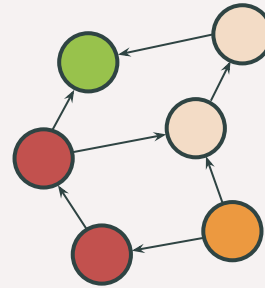
exact

Dijkstra



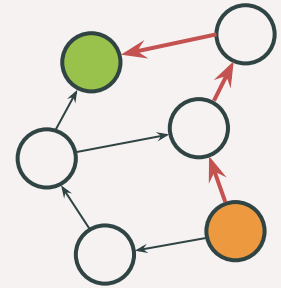
exact

Weighted A*



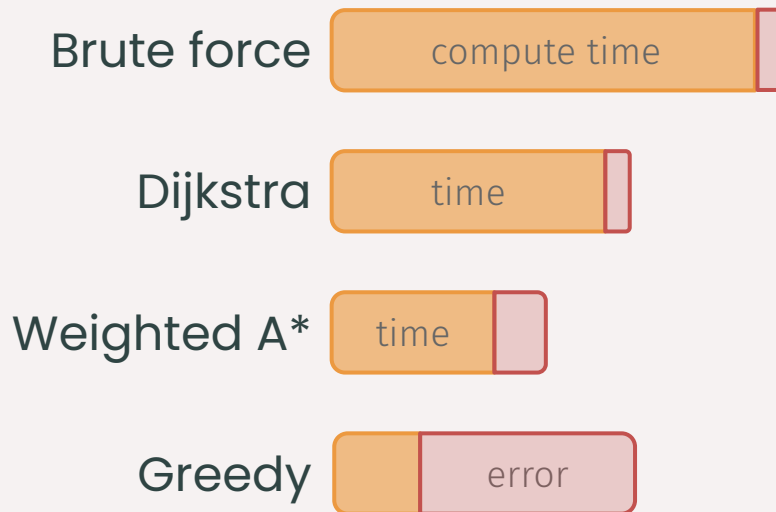
bounded error

Greedy

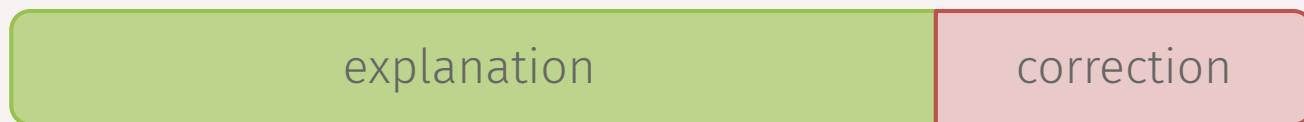


unbounded error

We can rank an algorithm goodness based on time and error.



Interpretability is a communication problem: the best explanations are the most compressed.



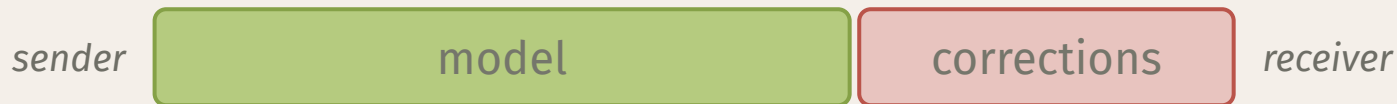
$L(model)$

$L(data | model)$

description length

$$\mathbf{L(data) = L(model) + L(data | model)}$$

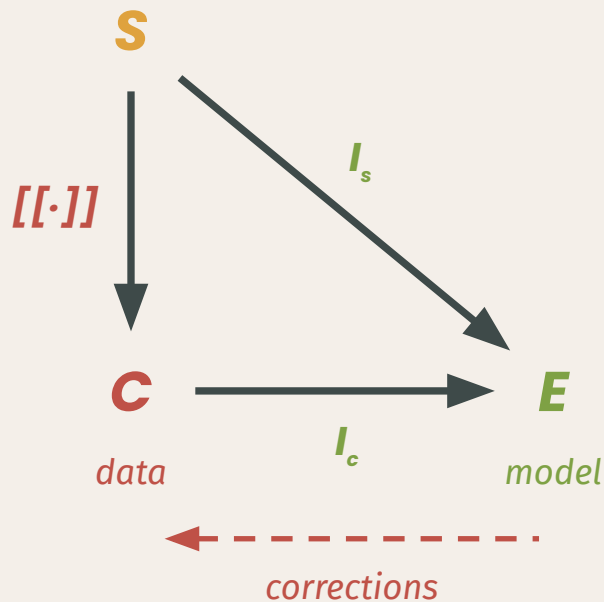
Interpretability is a communication problem: the best explanations are the most compressed.



description length

$$L(\mathbf{Data}) = L(\mathbf{Model}) + L(\mathbf{Data} \mid \mathbf{Model})$$

Describing the model through an explanation has two different costs.



$$L([[D]]) \leq L(I_s(D)) + L([[D]] | I_s(D))$$

For interpretations, we formalise this ranking with the notion of description length.

Syntactic interpretation

complexity

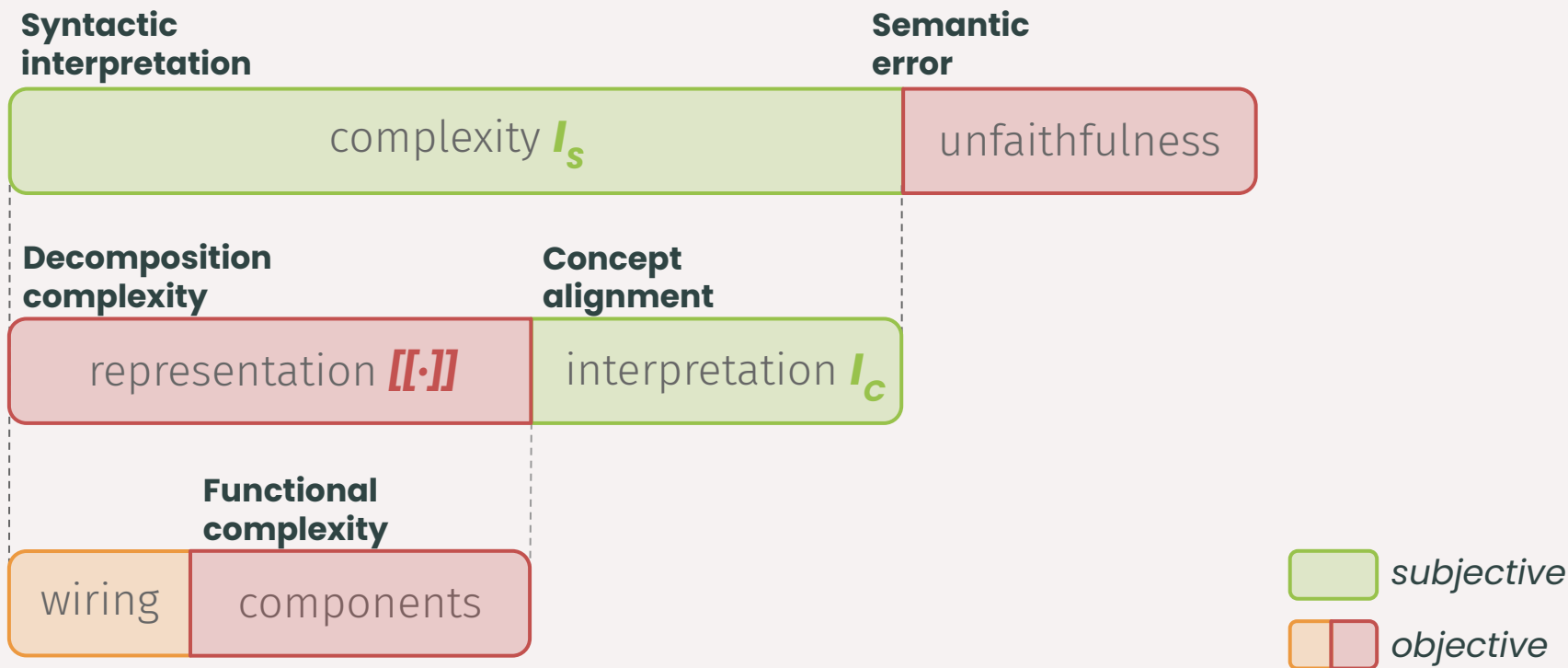
$$L(I_s(D))$$

Semantic error

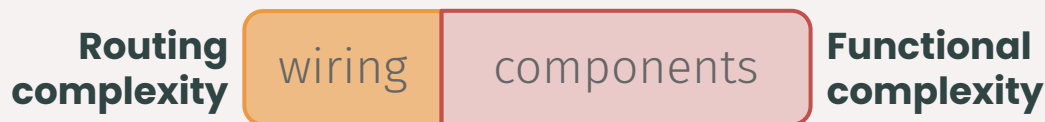
unfaithfulness

$$L([D] \mid I_s(D))$$

Since an interpretation is subjective, we decompose into parts.



Description length gives us a framework to assess when interpretations cheat.



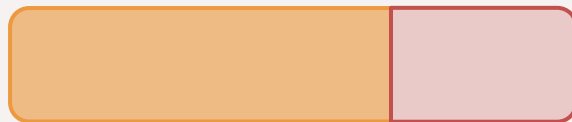
$$L(\text{ReLU}) + L(\text{Matrix})$$

=

=

?

...

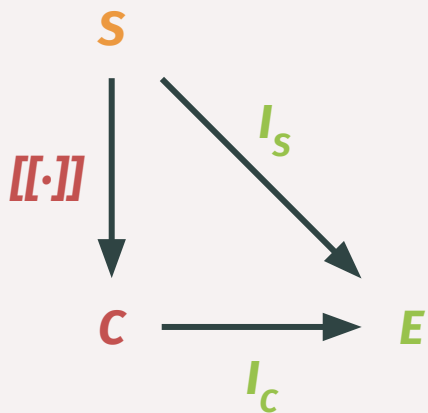


Syntactic complexity can degenerate if not measured

Coming up...

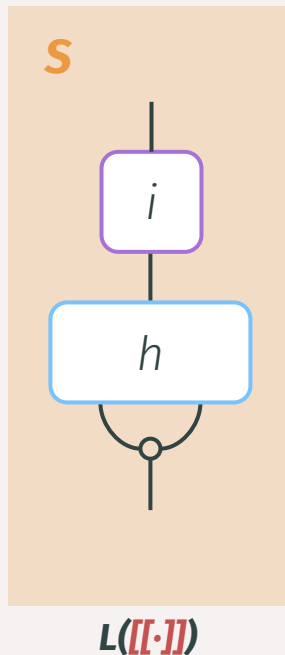
1. What are compositional interpretations?
2. What makes explanations good?
3. **How do we find better explanations?**

Compressive refinements can simplify representations without changing model behaviour.

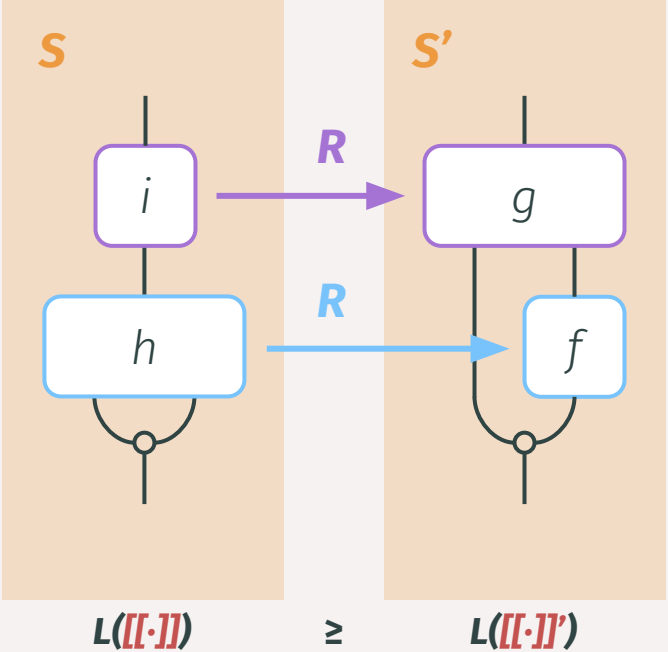
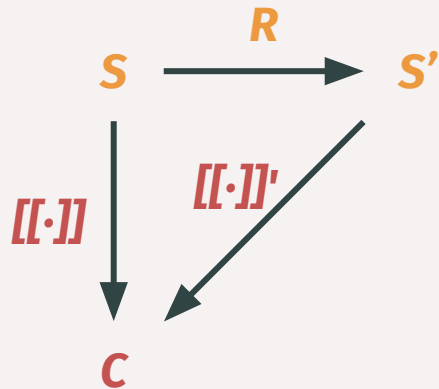


S Syntax
 C Semantics
 E Explanation

$[[\cdot]]$ Representation
 I_c Semantic interpretation
 I_s Syntactic interpretation

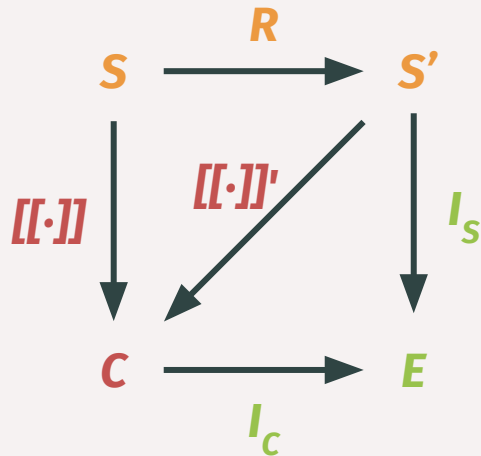


Compressive refinements can simplify representations without changing model behaviour.



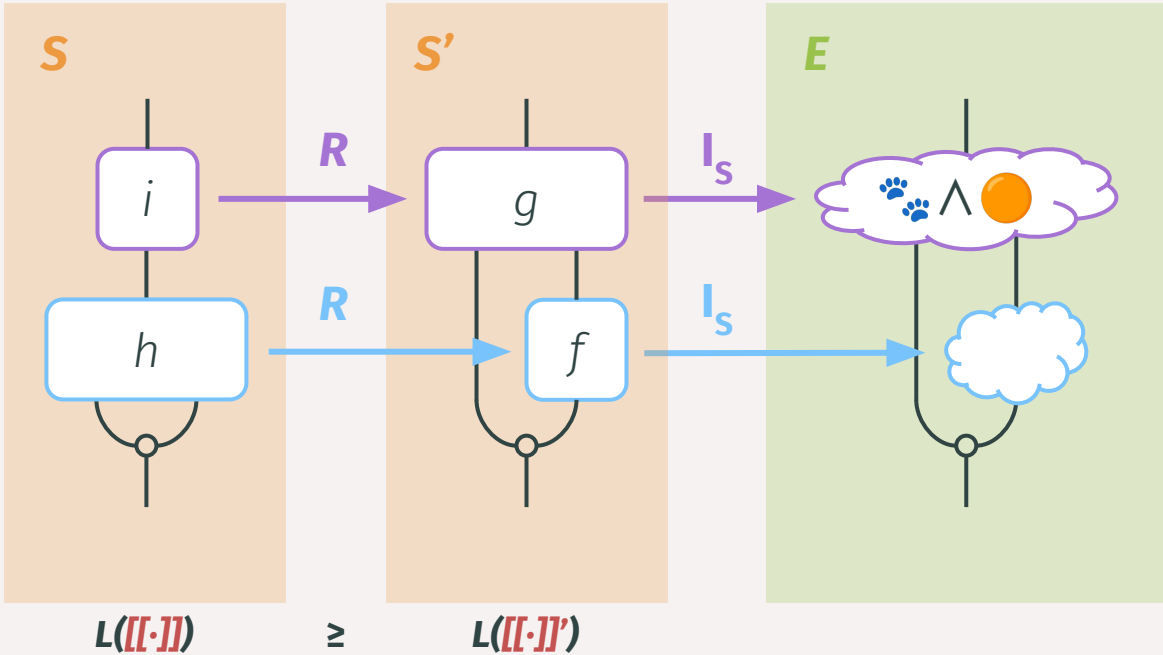
- S Syntax
- C Semantics
- E Explanation
- R Syntactic refinement
- $[[\cdot]]$ Representation
- I_C Semantic interpretation
- I_S Syntactic interpretation

Compressive refinements can simplify representations without changing model behaviour.



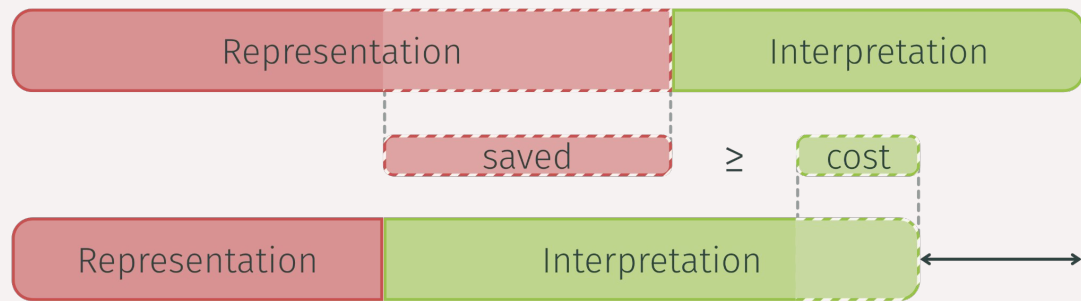
S Syntax
C Semantics
E Explanation

R Syntactic refinement
[[·]] Representation
I_C Semantic interpretation
I_S Syntactic interpretation



Simpler does not always mean more interpretable...
But it does under the *parsimony criterion*.

Complex syntax **S**

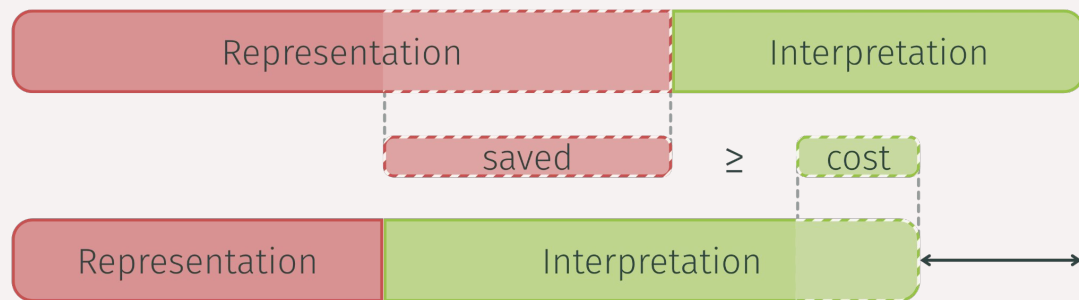


Refined syntax **S'**

Simpler does not always mean more interpretable...

But it does under the *parsimony criterion*.

Complex syntax **S**



Refined syntax **S'**

Comonotonic coding suffices.

A coding distribution is a prior.
Encode *explanatory optimism* in it,
and both complexities fall together.

$$L([[\cdot]]) := -\log_2 P([[\cdot]])$$

Make faithfulness and complexity *measurable*, and interpretability becomes *constrained optimisation*.

Search space
refinement in **S**

Constraint
faithful in **C**

Objective
complexity of **[[·]]**

Make faithfulness and complexity *measurable*, and interpretability becomes *constrained optimisation*.

Search space
refinement in **S**

Constraint
faithful in **C**

Objective
complexity of **[·]**

Interpretability is a spectrum, not a constraint.

- Including post-hoc, mechanistic and intrinsic



Make faithfulness and complexity *measurable*, and interpretability becomes *constrained optimisation*.

Search space
refinement in **S**

Constraint
faithful in **C**

Objective
complexity of **[·]**

Interpretability is a spectrum, not a constraint.

- Including post-hoc, mechanistic and intrinsic

Interpretability is relative to assumptions on **E.**

- To compare across **E**s, use description length.
- Choosing **E** is also a question for cognitive science.



Make faithfulness and complexity *measurable*, and interpretability becomes *constrained optimisation*.

Search space
refinement in **S**

Constraint
faithful in **C**

Objective
complexity of **[·]**

Interpretability is a spectrum, not a constraint.

- Including post-hoc, mechanistic and intrinsic

Interpretability is relative to assumptions on **E.**

- To compare across **E**s, use description length.
- Choosing **E** is also a question for cognitive science.

If description length stays high, design new models.



Interpretability Paradigms

Ward Gauderis

Compositional Interpretations

Ward Gauderis & Thomas Dooms

Architectures & Algorithms

Thomas Dooms

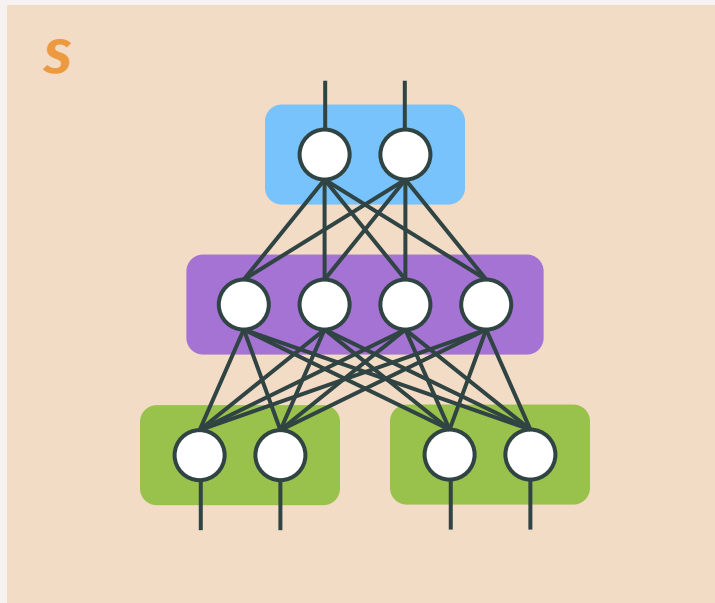
Neural networks are hard to refine and to measure. Syntactic refinement becomes approximation.

No interesting way to rewrite structure

A **neural model** is a compositional model with

- the syntax of a **copy-discard** string diagram
- semantics in **CartSp** (continuous maps between Cartesian spaces)

A box can be any function (in theory)



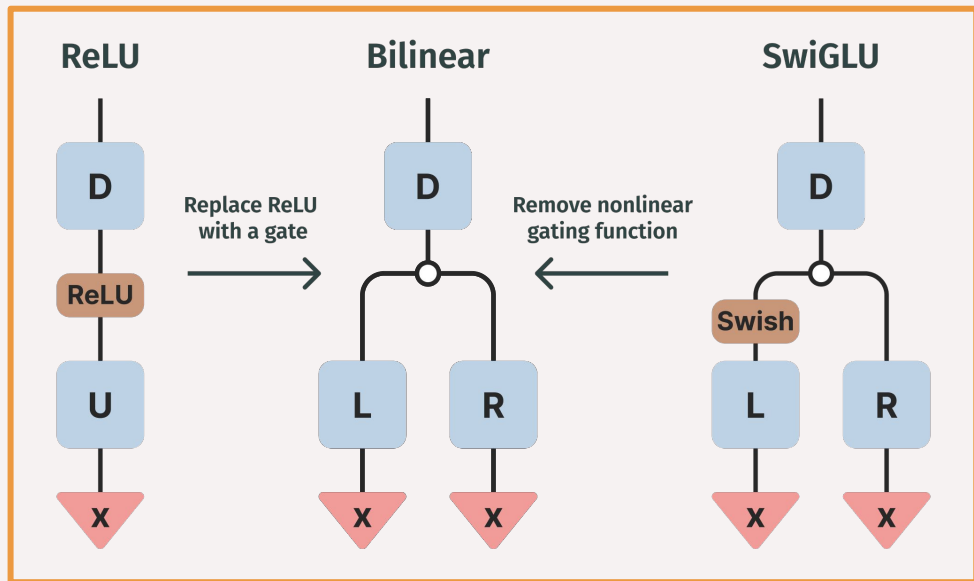
Tensor models are tractable to refine and measure. Their structure resembles neural architectures.

A rich framework for decompositions

A **tensor model** is a compositional model with

- the syntax of an **additive hypergraph**
- semantics in **FdHilb $\mathbb{R}$** (linear maps and finite-dimensional real Hilbert spaces)

A box is a higher-order interaction



*The appealing properties of **compositional understanding** can be scaled to **state-of-the-art networks** using tensor-based models.*

Coming up...

1. **How do we operationalise compositional interpretability?**
2. What can we do with these models?
3. How are these used in practice?

A curious table from a curious paper.

Training Steps	65,536	524,288
FFN_{ReLU} (<i>baseline</i>)	1.997 (0.005)	1.677
FFN_{GELU}	1.983 (0.005)	1.679
$\text{FFN}_{\text{Swish}}$	1.994 (0.003)	1.683
FFN_{GLU}	1.982 (0.006)	1.663
$\text{FFN}_{\text{Bilinear}}$	1.960 (0.005)	1.648
$\text{FFN}_{\text{GEGLU}}$	1.942 (0.004)	1.633
$\text{FFN}_{\text{SwiGLU}}$	1.944 (0.010)	1.636
$\text{FFN}_{\text{ReGLU}}$	1.953 (0.003)	1.645

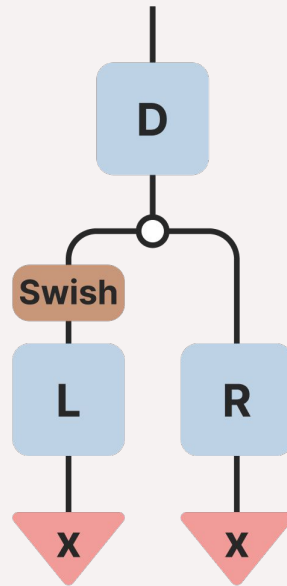
"We offer no explanation as to why these architectures seem to work; we attribute their success, as all else, to divine benevolence."

A bilinear layers is a tensor,
which can be interpreted compositionally.

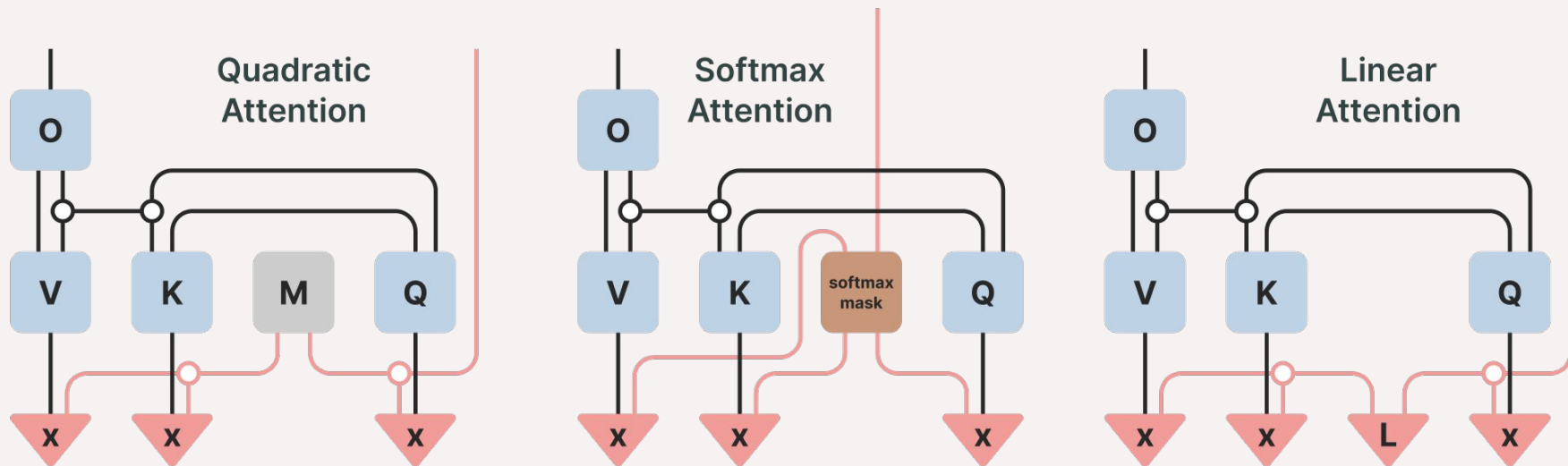
ReLU



SwiGLU



Attention can also be easily converted.
We can use a similar trick by making it quadratic.



Tensor models are performant across tasks. Training to the same loss is equally fast.



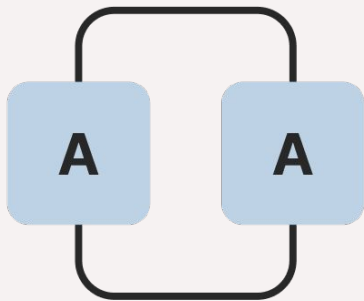
Coming up...

1. How do we operationalise compositional interpretability?
- 2. What can we do with these models?**
3. How are these used in practice?

Many operations on matrices are tensor networks.

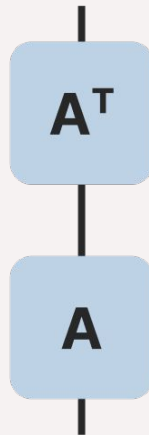
Frobenius Norm

$$\|A\|_F^2 = \sum_{ij} A_{ij} A_{ij}$$



Gram Matrix

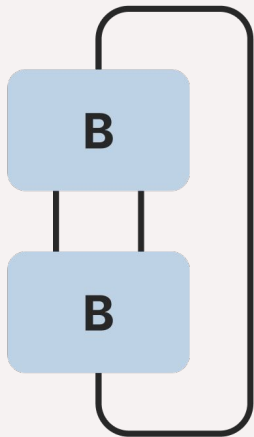
$$(A^\top A)_{jk} = \sum_i A_{ij} A_{ik}$$



These operations intuitively generalise to tensors.
This enables efficient computation of metrics.

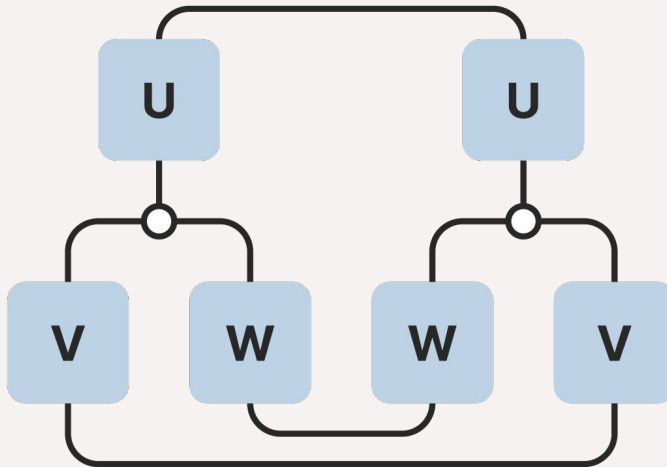
3-Tensor Norm

$$\|B\|_F^2 = \sum_{ijk} B_{ijk} B_{ijk}$$

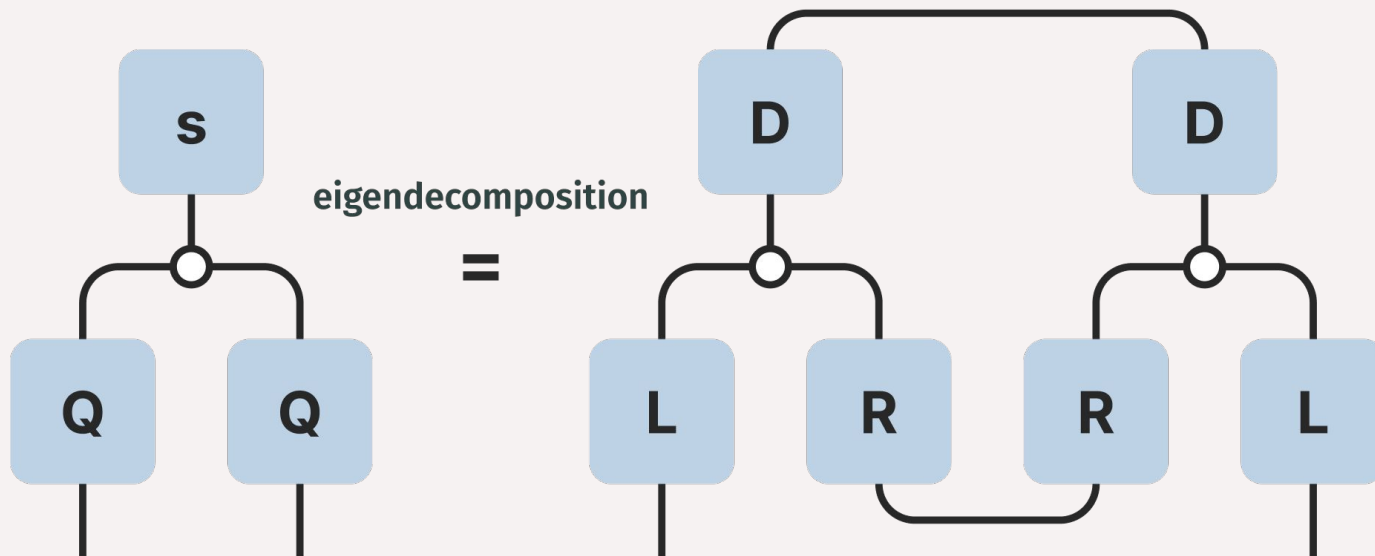


3-Tensor Norm

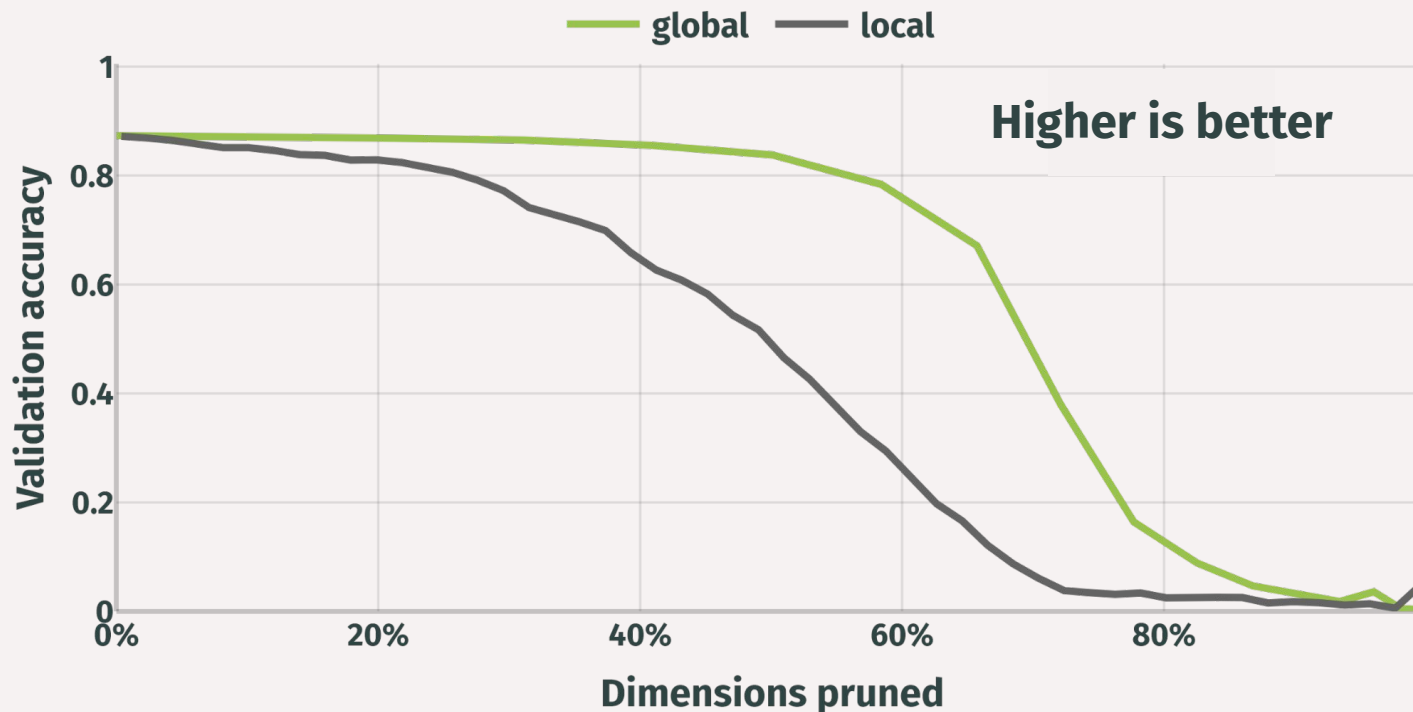
$$\|\cdot\|_F^2 = \sum_{ijknm} U_{in} V_{jn} W_{kn} U_{im} V_{jm} W_{km}$$



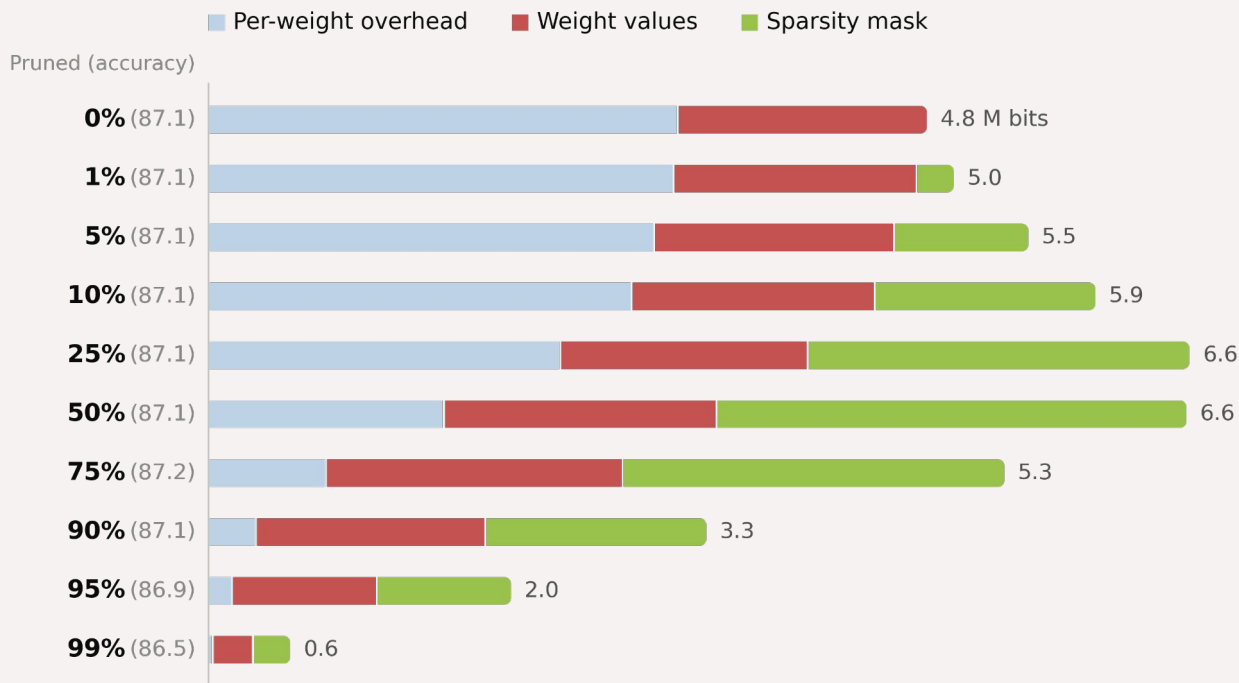
This enables decomposing models analytically while keeping into account all present structure.



Global decompositions outperforms local versions.
Models retain accuracy under strong compression.



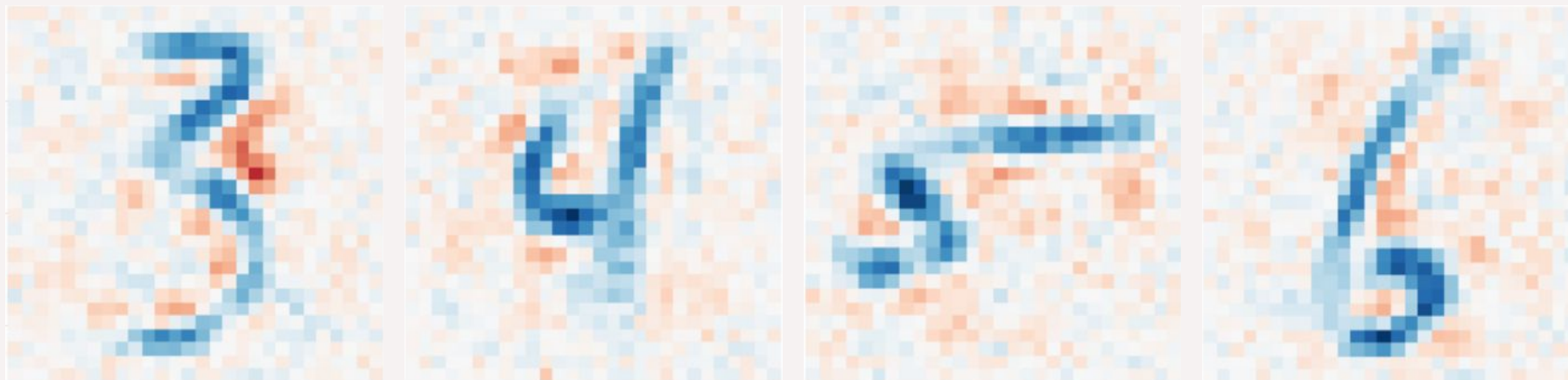
We can use weight sparsity to prune networks. This allows even lower description length.



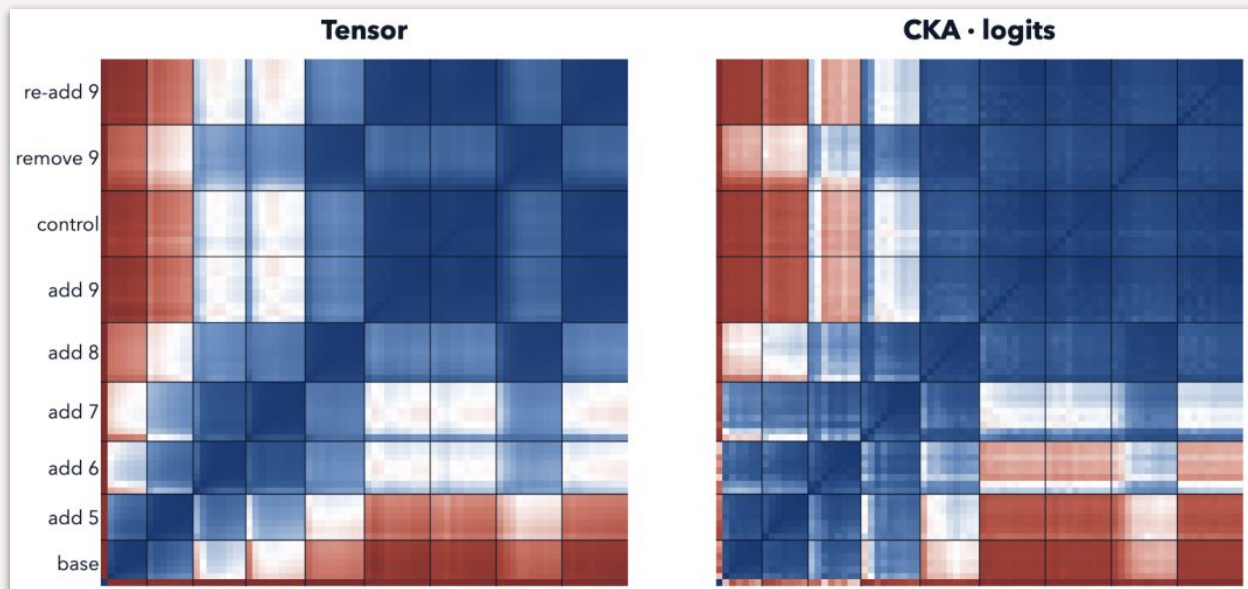
Coming up...

1. How do we operationalise compositional interpretability?
2. What can we do with these models?
3. **How are these used in practice?**

Rank decompositions also reveal what models learn, purely from their weights.

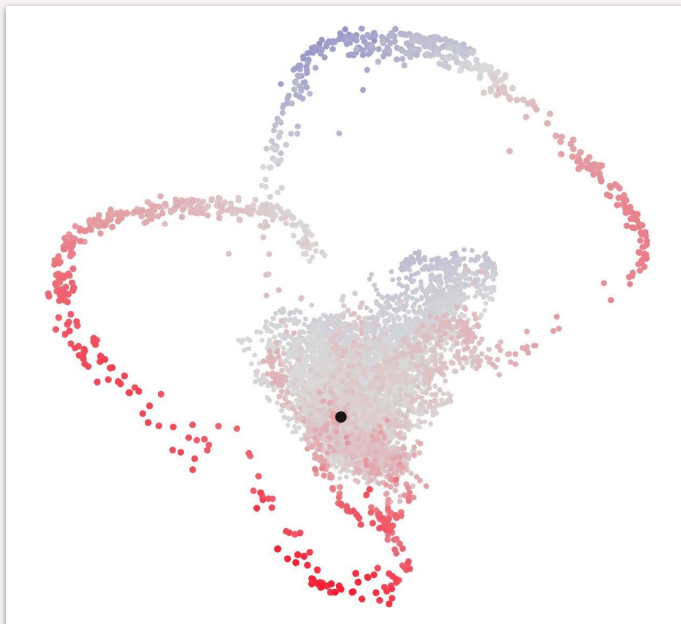


Tensors are not only useful for decomposing,
but can also measure similarity across training.

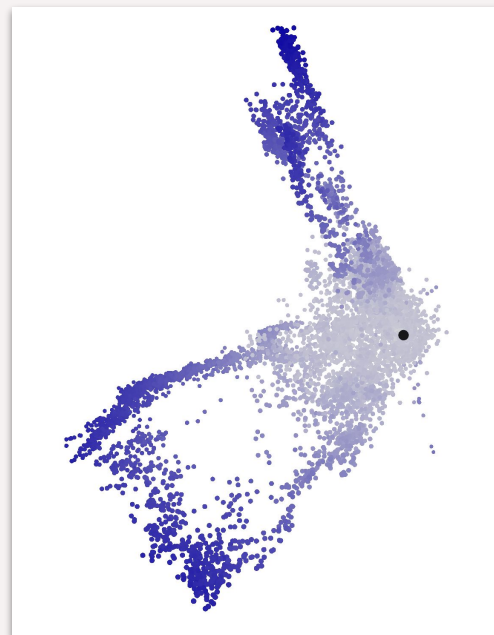


Interpretability aims to find the geometry of concepts inside neural models.

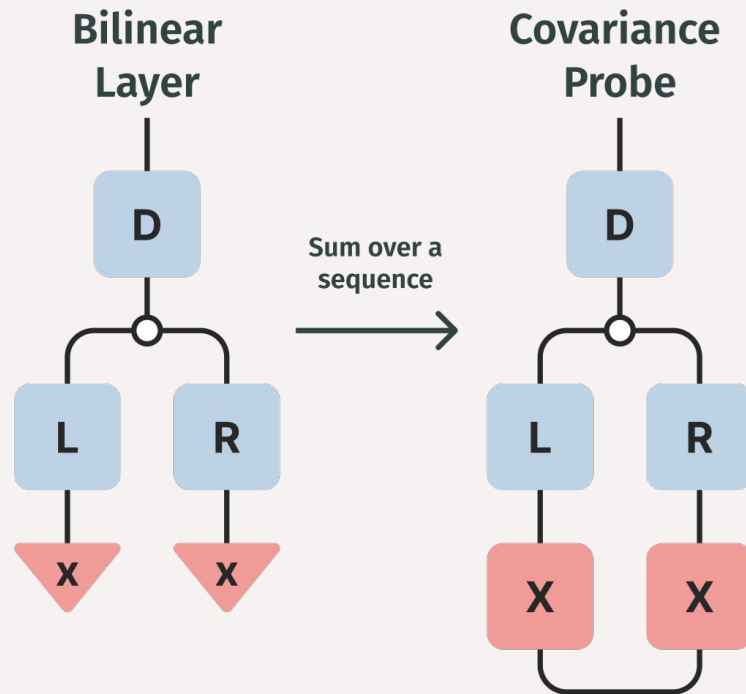
Sparse composition



Sparse activation



These tools are useful outside of language models, they can become drivers of scientific discovery.



These tools are useful outside of language models, they can become drivers of scientific discovery.



EVEE Evo Variant Eff

Search gene, rsID, or ClinVar ID (BRCA1, rs80357906, 15126)

GOODFIRE

SDHA c.217G>A (p.Gly73Ser)
ClinVar:2933037 | rs2477286087 | chr5:224426 G>A | exon 3/15 | Missense

PREDICTED PATHOGENICITY
98%

VUS ★★☆☆
Hereditary cancer-predisposing syndrome | gnomAD AF: not observed

ClinVar gnomAD dbSNP UCSC UniProt Ensembl OMIM GeneCards

VARIANT INTERPRETATION ?

Gly73Ser disrupts beta-strand geometry in the FAD-binding domain of SDHA while also perturbing a nearby splice acceptor context, collectively impairing proper protein folding and potentially pre-mRNA processing of the SDHA transcript.

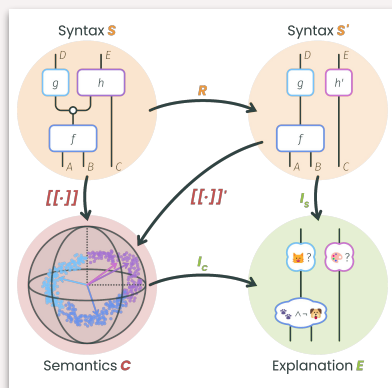
This glycine-to-serine substitution at position 73 of SDHA generates a high-confidence pathogenicity prediction of 98%, consistent with the calibration data showing 97% of variants in this score range are pathogenic. The most prominent disruptions are at a splice acceptor region approximately 945-952bp upstream, where polypyrimidine tract identity drops by 0.49, intron identity by 0.42, intron-to-exon transition signal by 0.41, and splice acceptor probability by 0.34, suggesting the variant propagates a conformational or sequence context change that destabilizes a nearby intronic regulatory element. At the variant site itself, the P-loop containing nucleoside triphosphate hydrolase domain signal increases by 0.34, which reflects a gain of anomalous domain-like character rather than preservation of normal architecture, and is accompanied by a gain of disorder (+0.27) and loss of beta-strand integrity (-0.27) in the immediate vicinity. SDHA encodes the flavoprotein subunit of succinate dehydrogenase (Complex II), where the FAD-binding domain near the N-terminus depends on precise beta-strand geometry for cofactor coordination and electron transfer; disruption of beta-strand structure at this position would impair FAD binding and catalytic function. The combination of local structural disruption and upstream splice region perturbation produces a broad disruption profile consistent with a pathogenic variant causing mitochondrial Complex II dysfunction.

KEY EVIDENCE

- Polypyrimidine tract disruption of -0.49 at -952bp, with correlated splice acceptor loss of -0.34 at -945bp
- Beta-strand identity loss of -0.27 at -20bp from variant, consistent with structural disruption in the FAD-binding domain
- Gain of disorder signal (+0.27) near the variant site, indicating local structural destabilization
- Anomalous P-loop hydrolase domain signal gain (+0.34) at the variant position, reflecting aberrant sequence context
- Calibration data: 97% of variants scoring 90-101% are pathogenic, and this variant scores 98%

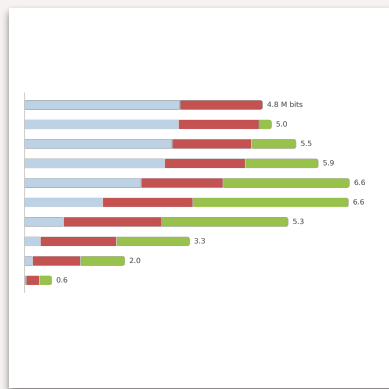
Interpretability is a constrained optimisation problem.

Optimize simplicity under compositional constraints.



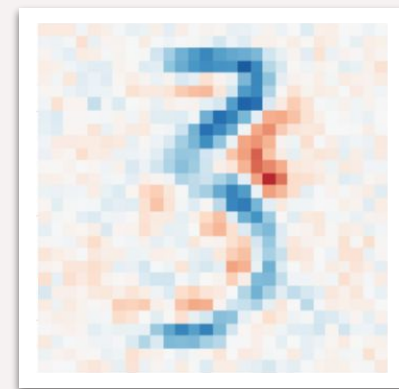
Composition

+



Compression

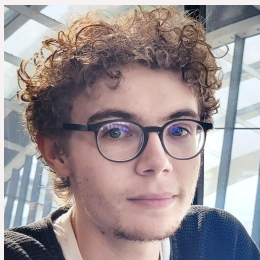
=



Interpretability*

*Left as exercise to the reader.

Want to know more or reach out?



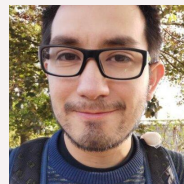
Ward Gauderis



Geraint Wiggins



Thomas Doods



Jose Oramas