Interpretability as Compression: Reconsidering SAE Explanations of Neural Activations

Applying the Minimum Description Length (MDL) principle to SAEs

MATS

Kola Ayonrinde, Michael Pearce

Summary

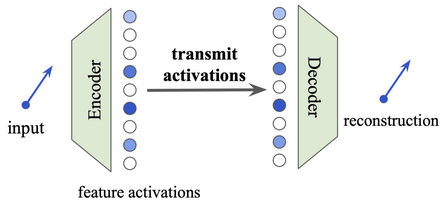
- We interpret **SAEs as lossy compression algorithms** for communicating neural activations.
- Interpretable SAEs require "**independent additivity**" so that features can be understood separately.
- The Minimal Description Length (MDL) principle finds the SAE with the **simplest explanation of activations**.
  - We demonstrate this approach for SAEs trained on MNIST
- **MDL avoids potential pitfalls** with naively maximizing sparsity such as over-splitting features and may encourage more intuitive features.

Why is this important?

- The goal of SAEs is to find interpretable features but it's **unclear how to optimize for interpretability**.
  - Sparsity is used as an imperfect proxy.
  - Human-interp and auto-interp cannot be used during training.
- Minimal Description Length principle provides an **operational proxy for interpretability** and motivate future architectures.

SAEs are feature explanations

Communication Protocol: Use SAEs to compress and send neural activations.



Description Length (DL) is the number of bits sent to reconstruct the activations. It generally has two parts:

$$DL\left(\vec{x};\mathrm{SAE}\right) = DL\left(\vec{z}\,\right) + \cancel{DL\left(\mathrm{Dec}(\cdot)\right)}$$

feature activations

Decoder (sent once), negligible for large data limit

"All else equal, prefer the simpler explanation."
The Minimal Description Length principle operationalizes Occam's Razor

Independent additivity is needed for interpretability

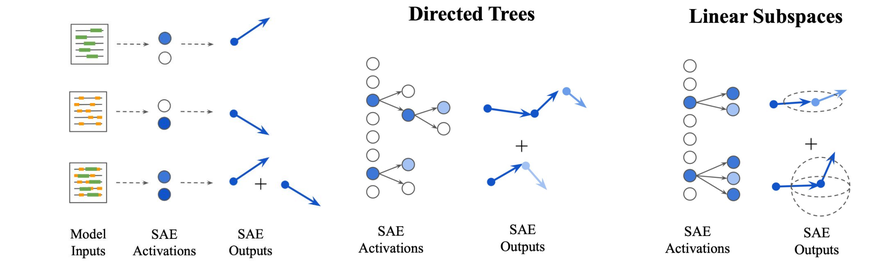
Definition: Independent additivity
A feature-based explanation is independently additive if the sum of the explanation of the features is the explanation of the sum of the features.

$$\mathrm{Dec}\left(\vec{z}\right) = \sum_{i \in \mathrm{Features}} \mathrm{Dec}\left(\vec{z}_i\right)$$

$$DL\left(\vec{z}\right) = \sum_{i \in \mathrm{Features}} DL\left(\vec{z}_i\right)$$

Allows features to be understood separately without disentangling interactions

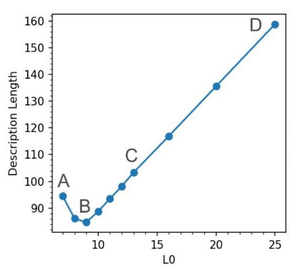
Compatible with more complex architectures that may lower DL



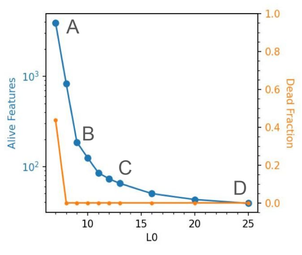
SAEs should be sparse, but not too sparse

We trained a set of SAEs to reconstruct MNIST digits with the same MSE.

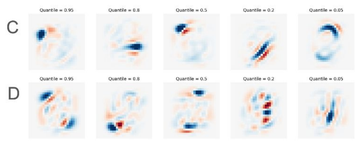
Description Length vs L0



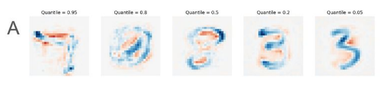
Features vs L0



- **Low Sparsity, Few Features (C, D)**: DL is large because many nonzero floats need to be sent.



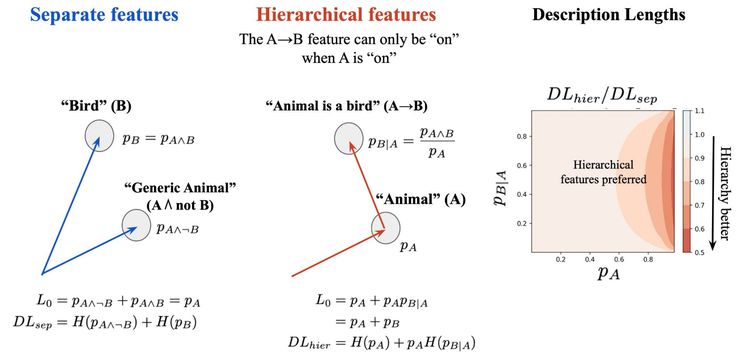
- **High Sparsity, Many Features (A)**: DL is large because of the additional bits needed to specify which features are active.



- **The MDL solution (B)**: DL balances between nonzero floats and bits needed to specify active features.

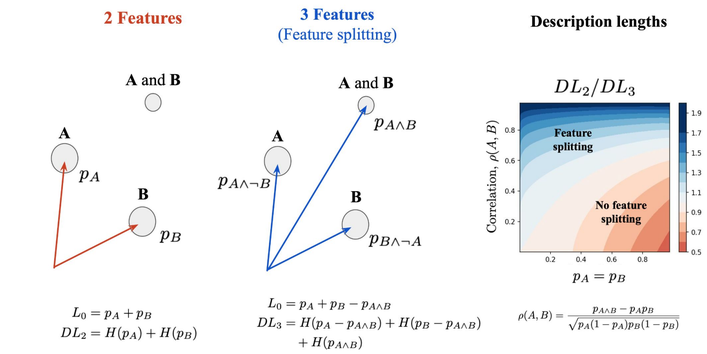
Hierarchical features can reduce DL

Hierarchical features reduce description length since the child feature activations only need to be sent if the parent feature is active. This is an example of a **variable coding scheme.**



MDL may avoid over-splitting features

Maximizing sparsity favors representing the "and" of two features as a third feature. MDL allows more nuance for when to do feature splitting.



Future Work

- Explore using the MDL principle to select optimal SAEs using existing training methods.
- Try new SAE architectures incorporating hierarchical and group structures which reduce MDL (in progress).

References

Cunningham et al. (2023). "Sparse Autoencoders Find Highly Interpretable Features in Language Models"

Bricken et al. (2023). "Towards Monosemanticity: Decomposing Language Models With Dictionary Learning"

MacKay (2003). "Information Theory, Inference, and Learning Algorithms"

Lee et al. (2001). "An Introduction to Coding Theory and the Two-Part Minimum Description Length Principle"