Knowledge Vector of Logical Reasoning in Large Language Models

Zixuan Wang and Yuanyuan Lei
Department of Computer & Information Science and Engineering
University of Florida, Gainesville, FL, United States
{zwang10, yuanyuan.lei}@ufl.edu

Abstract

Logical reasoning serve as a central capability in LLMs and includes three main forms: deductive, inductive, and abductive reasoning. In this work, we study the knowledge representations of these reasoning types in LLMs and analyze the correlations among them. Our analysis shows that each form of logical reasoning can be captured as a reasoning-specific knowledge vector in a linear representation space, yet these vectors are largely independent of each other. Motivated by cognitive science theory that these subforms of logical reasoning interact closely in the human brain, as well as our observation that the reasoning process for one type can benefit from the reasoning chain produced by another, we further propose to refine the knowledge representations of each reasoning type in LLMs to encourage complementarity between them. To this end, we design a complementary subspace-constrained refinement framework, which introduces a complementary loss that enables each reasoning vector to leverage auxiliary knowledge from the others, and a subspace constraint loss that prevents erasure of their unique characteristics. Through steering experiments along reasoning vectors, we find that refined vectors incorporating complementary knowledge yield consistent performance gains. We also conduct a mechanism-interpretability analysis of each reasoning vector, revealing insights into the shared and specific features of different reasoning in LLMs[1].

1 Introduction

Despite the impressive capabilities of Large Language Models (LLMs) across diverse tasks (OpenAI et al., 2024; Guan et al., 2023; Yao et al., 2022), understanding their reasoning abilities, particularly logical reasoning (Pan et al., 2023; Lam et al., 2024), which is the foundation of problem solving, remains poorly understood and requires deeper investigation. Logical reasoning is commonly organized into three principal categories in standard logic theory (Peirce, 1934): abductive, deductive, and inductive reasoning. In practice, it has become a new understanding perspective being applied to different areas, such as procedural planning (Kang et al., 2025), coding (Vashishtha et al., 2025), and mathematics (Abdaljalil et al., 2025), where the model must draw reliable conclusions or hypotheses from structured information rather than relying solely on surface correlations (Bedi et al., 2025). Therefore, understanding how LLMs perform logical reasoning is essential for building models that are trustworthy, generalizable, and aligned with human expectations of rational inference.

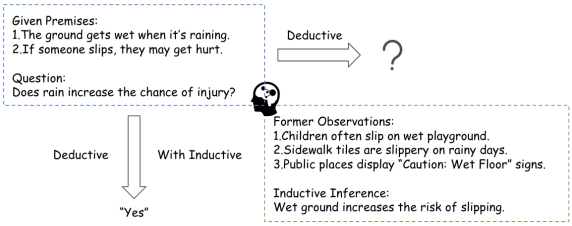


Figure 1: An example showing how multiple logical reasoning strategies cooperate in human thinking. Rather than operating in isolation, inductive reasoning contributes newly formed premises derived from former observations, which are incorporated into a deductive reasoning process to reach a final conclusion.

At the same time, knowledge vectors (Rimsky et al., 2024; Turner et al., 2023) have been proven to be meaningful under the assumption of linear representation in a model's activation space (Jorgensen et al., 2023; Zou et al., 2023). Prior work suggests that certain knowledge-related concepts can be encoded linearly, such as truthfulness (Wang et al., 2025a), instruction-following (Stolfo et al., 2024) etc. However, no prior works study whether reasoning abilities, particularly different forms of logical reasoning, can be linearly represented in the model's activation space. The most closely related

[1]The code link is: https://github.com/lei-nlp-lab/knowledge_vector_acl_2026

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 43637–43651
July 2-7, 2026

line of work is Venhoff et al. (2025), which investigates knowledge vectors for specific reasoning patterns such as backtracking, but does not address reasoning capabilities in a more general sense.

To investigate whether different types of logical reasoning can be linearly represented within LLMs, we extract a knowledge vector for each reasoning type and examine whether the knowledge-enhancement effect emerges as well (Nanda, 2023). By using these vectors to steer the model's behavior, we observe improved reasoning performance corresponding to each reasoning vector.

However, further geometric analysis reveals low pariwise cosine similarities among these naively extracted reasoning vectors, which means different types of reasoning knowledge are represented as largely distinct representation in the model's activation space. This observation contrasts with findings in human cognitive science (Holyoak and Morrison, 2013; Heit and Rotello, 2010), which suggest that different forms of logical reasoning share common components and interact in a complementary manner in human thinking process. Take Figure 1 as an illustration, inductive reasoning can support deductive reasoning by contributing newly formed premises derived from prior observations, thereby facilitating the derivation of a final conclusion.

Inspired by the above observations, we propose a complementary refinement method for enhancing reasoning knowledge, by encouraging the model to mimic human reasoning paradigm. Sparse Autoencoders (SAEs) have been shown to reliably interpret semantic features within LLMs by disentangling the complex, superimposed features into more interpreterable components (Shu et al., 2025). Leveraging this property, we introduce a complementary subspace-constrained refinement framework, consisting of (i) a complementary loss that enables each reasoning vector to incorporate complementary reasoning knowledge from the other two, thereby refining the resulting knowledge vectors, and (ii) a subspace constraint loss that restricts each reasoning vector to its corresponding SAE-induced subspace, preventing the erasure of its unique characteristics. Take Llama-3.1-8B-it as an example, by using the complementary vectors, we observed the metric boost $55.22 - 56.46$, $27.13 - 27.55$, $39.10 - 42.07$ for deductive, inductive and abductive reasoning respectively.

To better understand the model's internal reasoning mechanisms, we further conduct a fine-grained mechanism interpretation analysis. Firstly, we examine SAE-extracted feature sets across three types of logical reasoning. Our analysis reveals the emergence of complementary features acquired from other reasoning types through the refinement process. We also observe that deductive and inductive reasoning becomes more closely aligned while abductive becomes more specialized, which is consistent with the fact that deductive and inductive reasoning share a similar evidence-based reasoning style. Secondly, we employ attention patching (Heimersheim and Nanda, 2024) to analyze the model's internal circuit before and after refinement. Our finding is that core activations are largely preserved, activation patterns become more concentrated, and new activations emerge following the refinement process. Thirdly, our qualitative analysis identifies key text spans associated with each type of logical reasoning, illustrating how the model allocates attention across different reasoning processes (Han et al., 2024).

2 Preliminary Exploration

2.1 Preliminary Settings

To investigate whether different types of logical reasoning can be represented linearly in LLMs, we use three datasets-JustLogic, DEER, and ART-corresponding to deductive, inductive, and abductive reasoning, respectively (Chen et al., 2025; Yang et al., 2024; Bhagavatula et al., 2019). Details about how we use these datasets are provided in Appendix A. As for the evaluation metric, we choose accuracy for deductive and abductive reasoning and METEOR (Banerjee and Lavie, 2005) score for inductive reasoning according to the characteristics of different datasets. To ensure fairness, incomplete generations without candidate answers are excluded from accuracy calculations for deductive and abductive reasoning. For inductive reasoning, we take the last three generated sentences, compute METEOR against the ground truth, and use the highest score as the final metric.

For activation extraction and steering, we use layer 13 for both Llama-3.1-8B-it and Gemma-2-9B-it. This choice follows prior work on activation steering, which suggests that middle residual-stream layers tend to contain abstract yet consistently controllable representations, making them suitable intervention points for linear steering directions (Rimsky et al., 2024; Zhang and Viteri). Besides, to examine whether the observed pattern extends to larger models with different model

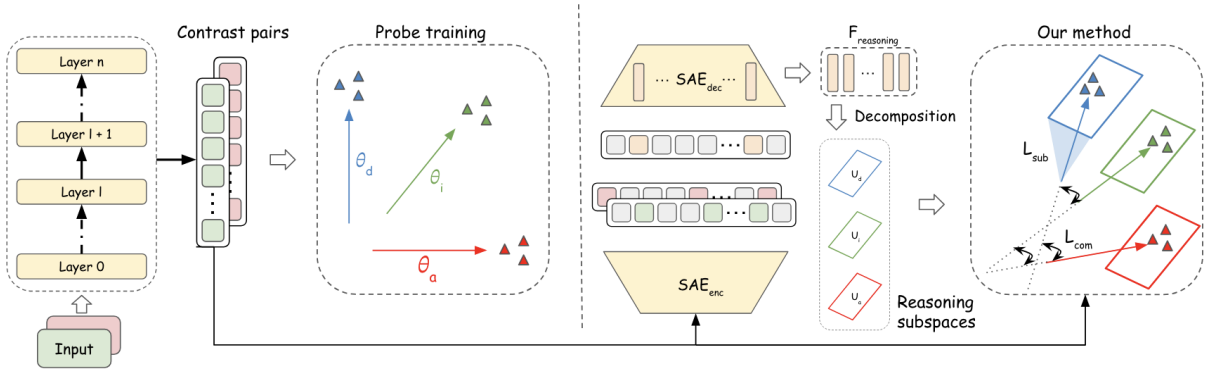


Figure 2: Comparison between the naive and complementary extraction. The left panel illustrates the naive approach, where contrast pairs are collected and used to train linear probes that serve as reasoning vectors. θ_d, θ_i and θ_a are reasoning vectors for deductive, inductive and abductive reasoning respectively. The right panel presents our complementary method, where we first extract highly related reasoning features $F_{reasoning}$ from SAE and construct a subspace for each type of logical reasoning via QR decomposition. A complementary knowledge loss L_{com} is designed, together with a subspace constraint loss L_{sub} based on SAE-induced subspace.

Method	Llama-3.1-8B-it			Gemma-2-9B-it			GPT-OSS-20B		
	Deductive	Inductive	Abductive	Deductive	Inductive	Abductive	Deductive	Inductive	Abductive
Greedy									
Unsteered	48.95	26.36	32.27	56.86	24.52	54.67	56.12	16.68	45.09
Mono Steering	55.22	27.13	39.19	57.33	26.49	55.74	57.63	17.24	47.19
Complementary Steering	**56.46**	**27.55**	**40.95**	**59.05**	**27.03**	**58.20**	**58.43**	**18.52**	**50.50**
Sampling@5									
Unsteered	47.45	24.25	39.01	49.74	23.85	54.92	51.12	22.91	41.69
Mono Steering	49.90	24.48	39.10	50.69	24.51	55.98	52.08	23.16	42.13
Complementary Steering	**51.36**	**24.78**	**42.07**	**54.64**	**25.56**	**57.53**	**53.67**	**23.92**	**46.89**

Table 1: Performance comparison under Greedy and Sampling@5 decoding settings. Greedy is the default decoding strategy, while Sampling@5 denotes averaging results over five sampled generations. Complementary steering consistently achieves the strongest performance.

structure, we additionally evaluate our method on GPT-OSS-20B (OpenAI, 2025), a larger model based on the Mixture-of-Expert (MoE) architecture. More implementation details are provided in Appendix B.

2.2 Naive Reasoning Vectors Extraction

Following prior work on contrastive activation analysis and knowledge-vector extraction (Rimsky et al., 2024; Zou et al., 2023; Wang et al., 2025a), we firstly construct contrastive activation sets that capture successful versus failed reasoning behaviors. Specifically, we design paired prompts that elicit different levels of reasoning performance: a strong version intended to support correct reasoning and a weak version that leads to failure. During generation, we record the residual stream activations at selected layers for every generated token. For each data instance, if the strong prompt yields a correct reasoning outcome while the weak prompt yields an incorrect one, we treat this as a valid contrastive pair. We then average the recorded activations across the entire generation trajectory to obtain a positive activation vector (successful reasoning) and a negative activation vector (failed reasoning). Formally, each training example provides a pair of activation representations:

$$\bar{a}^{+}_{i,l} = \frac{1}{N}\sum_{j=1}^{N} a^{+}_{j,l} \quad \text{and} \quad \bar{a}^{-}_{i,l} = \frac{1}{N}\sum_{j=1}^{N} a^{-}_{j,l} \quad (1)$$

where $\bar{a}^{+}_{i,l}$ and $\bar{a}^{-}_{i,l}$ denote the positive (successful reasoning) and negative (failed reasoning) activation vectors for instance i at layer l, and the averages are computed over all N token activations collected across the generation process.

Probes help uncover the model's internal representation mechanisms (Park et al., 2023; Belinkov, 2022; Alain and Bengio, 2016). Let D_r denote the constructed contrastive activation set for reasoning r, where $r \in \mathcal{P}, \mathcal{P} =$

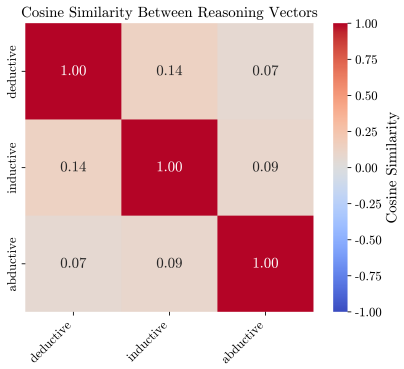


Figure 3: Cosine similarity heatmaps between knowledge vectors for different logical reasoning. The values range from -1 to 1, where negative values denote opposing directions and positive values indicate alignment. The generally near-zero scores indicate that the learned vectors encode distinct reasoning patterns in model's activation space.

$\{deductive, inductive, abductive\}$ and $D_r = \{\bar{a}^+_{0,l} ..., \bar{a}^+_{k,l}\, \bar{a}^-_{0,l} ..., \bar{a}^-_{k,l}\}$, we train a linear probe:

$$p_r = \sigma(\theta_r^\top x_{i,r} + b_r), \tag{2}$$

where $x_{i,r} \in D_r$ denotes the activation sample for reasoning type r, and θ_r is the learned reasoning knowledge vector obtained as the probe's weight parameters. The naive reasoning vector extraction is illustrated in the left panel of Figure 2.

2.3 Reasoning Vector Analysis

As shown in Table 1, mono-steering consistently outperforms the unsteered setting, which means the linear assumption works for logical reasoning and these vectors encode corresponding meaningful reasoning knowledge. However, we further investigate the geometric relationships between these vectors by measuring their pairwise cosine similarities and observe that most values are close to zero, as shown in Figure 3, which means LLMs represent different types of reasoning in largely orthogonal directions (Scalena et al., 2024). In other words, different reasoning knowledge is represented as geometrically distinct directions in the model's activation space. This is contradictory to human cognitive science understanding (Holyoak and Morrison, 2013; Heit and Rotello, 2010), which posit that deductive, inductive, and abductive reasoning arise from a common set of cognitive operations and interact in complementary ways during human inference.

3 Complementary Reasoning Refinement

3.1 Complementary Knowledge Integration

Motivated by the prior analysis, we propose a complementary subspace-constrained refinement framework, which enables each reasoning vector to learn complementary knowledge from other reasoning types while preserving its own identified features. The framework is illustrated in the right panel of Figure 2.

Complementary Knowledge Enhancement. To encourage knowledge sharing across different reasoning types, we introduce the following cosine-based objective:

$$\mathcal{L}_{\text{com}} = -\sum_{\substack{r,s\in\mathcal{P} \\ r\neq s}} \frac{\theta_r^\top \theta_s}{\|\theta_r\|_2 \, \|\theta_s\|_2}, \tag{3}$$

where θ_r and θ_s denote different reasoning knowledge vectors of type r and s. However, relying solely on this complementary loss makes it difficult to control the extent to which the reasoning vectors align. Excessive alignment can cause the vectors to collapse toward a shared direction, leading to the loss of the distinctive features that each vector originally captures.

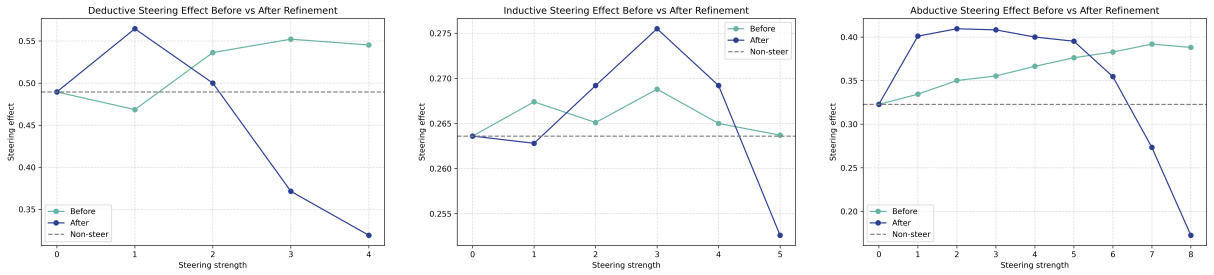
Reasoning Subspace Constraint. In order to ensure that complementary knowledge is learned without erasing their distinct characteristics, we introduce a subspace constraint for each reasoning type as well. We note the success of SAEs in extracting fine-grained semantic representations, especially the abstract concepts in LLMs (Cunningham et al., 2023; Hua et al., 2025; Wang et al., 2025b). Here, we use SAEs to construct the reasoning-specific subspace. For each reasoning type r, we feed the recorded positive and negative activations into a pretrained SAE to obtain their sparse latent representations:

$$z = \text{SAE\_encode}(h), \tag{4}$$

where h denotes the residual-stream activation and $z \in \mathbb{R}^m$ is the SAE hidden vector with high sparsity.

To identify the SAE features most predictive of reasoning success, we compute the mean squared activation of each latent unit across positive and negative generations:

$$\mu_r^+(j) = \mathbb{E}\big[z_j^2 \mid \text{pos}\big] \quad \text{and} \quad \mu_r^-(j) = \mathbb{E}\big[z_j^2 \mid \text{neg}\big] \tag{5}$$



(a) Deductive (b) Inductive (c) Abductive

Figure 4: The effect of different steering strength before and after complementary refinement. Using Llama-3.1-8B-it as an example. Before: using the naive extracted reasoning vectors to steer the model. After: using the complementary refined reasoning vectors to steer the model.

We then compute a contrastive activation ratio for each SAE feature (Muhamed et al., 2025):

$$\rho_r(j) = \frac{\mu_r^+(j)}{\mu_r^-(j) + \varepsilon}, \tag{6}$$

where ε is a small constant ensuring numerical stability. By setting a threshold τ as the α-quantile (e.g., $\alpha = 0.9$) of the ratio distribution. Only features with $\rho_r(j) \geq \tau$ are retained as the initial candidate set, ensuring that only those units that activate substantially more during successful reasoning than failures are kept. This removes features that are uniformly active or inactive across cases.

However, a high ratio does not guarantee that the feature contributes substantially in absolute magnitude. Therefore, after ratio-based filtering, we further select the top-K features with the largest mean activation strength, yielding a stable and discriminative reasoning-specific feature set denoted as $\mathcal{F}_r$.

For each feature $j \in \mathcal{F}_r$, we obtain its corresponding decoder direction w_j^{dec} from the SAE decoder matrix. Stacking these directions yields a basis matrix:

$$V_r = \left[w_j^{\text{dec}} \right]_{j \in \mathcal{F}_r} \in \mathbb{R}^{d \times K}, \tag{7}$$

Finally, we orthogonalize this matrix using QR decomposition to obtain a reasoning-specific subspace:

$$U_r = \text{QR}(V_r), \tag{8}$$

The columns of U_r form an orthonormal basis capturing the key feature directions associated with reasoning type r.

To prevent the refined reasoning vectors from shifting away from their reasoning-specific feature structure, we constrain each vector to remain close to its designated subspace. Given the orthonormal basis U_r for reasoning type r, we decompose the probe vector θ_r into its in-subspace and out-of-subspace components. The orthogonal component is penalized through the following subspace-preservation loss:

$$\mathcal{L}_{\text{sub}}^{(r)} = \left\| (I - U_r U_r^\top)\, \theta_r \right\|_2^2, \tag{9}$$

where $(I - U_r U_r^\top)\theta_r$ extracts the part of θ_r lying outside the reasoning subspace. Minimizing this term ensures that the reasoning vector learns complementary knowledge while retaining the reasoning-specific features encoded by U_r.

With the binary cross-entropy loss $\mathcal{L}_{\text{probe}}^{(r)}$ for each reasoning type $r \in \mathcal{P}$, the overall refinement objective is:

$$\mathcal{L} = \sum_{r \in \mathcal{P}} \mathcal{L}_{\text{probe}}^{(r)} \; + \; \lambda_{\text{com}}\, \mathcal{L}_{\text{com}} \; + \; \lambda_{\text{sub}} \sum_{r} \mathcal{L}_{\text{sub}}^{(r)}, \tag{10}$$

This unified objective jointly encourages: (i) accurate separation of successful and failed reasoning behaviors, (ii) controlled sharing of complementary knowledge across reasoning types, and (iii) preservation of reasoning-specific feature structure within each subspace.

3.2 Complementary Reasoning Results

Table 1 shows that our refined complementary reasoning vectors consistently outperform both the unsteered baseline and the initial reasoning vectors across deductive, inductive, and abductive tasks on Llama-3.1-8B-it and Gemma-2-9B-it, demonstrating more effective and generalizable reasoning representations via SAE-based refinement. As shown in Table 1, this trend also generalizes to GPT-OSS-20B, a larger MoE-based model, where both mono

	Deductive	Inductive	Abductive
Our method	56.46	27.55	40.95
w/o (i)	-2.81	-0.46	-3.33
w/o (ii)	-4.58	-0.29	-2.09

Table 2: Ablation study of our complementary method across three logical reasoning types on Llama-3.1-8B-it model. (i)w/o Complementary Konwledge Enhancement (ii) w/o Reasoning Subspace Constraint.

steering and complementary steering outperform the unsteered baseline, and complementary steering achieves the strongest overall performance.

It is worth noting that the absolute gains are modest, which is expected given the difficulty of the logical reasoning benchmarks considered in this work. These tasks require multi-step inference and remain challenging even for strong LLMs. We therefore interpret the consistent gains across reasoning types and and model families as evidence that the extracted reasoning vectors capture meaningful aspects of reasoning behavior. The practical significance of our method lies in providing a controllable and analyzable handle on reasoning-related representations.

Figure 4 further illustrates the effect of different steering strength on reasoning performance before and after complementary refinement. While performance exhibits a similar dependence on steering strength across all reasoning types, improving at moderate coefficients and degrading under over-steering, our refined vectors achieve higher peak performance at smaller coefficients. This suggests that they capture cleaner and better aligned reasoning directions that require less amplification.

3.3 Ablation Study

Table 2 presents the ablation study of the two key components in our complementary refinement: (i) without Complementary Konwledge Enhancement (ii) without Reasoning Subspace Constraint. In each setting, one component is removed at a time. The results show that removing either one of the components leads to performance degradation across all three types of logical reasoning, while combining them together achieves the best performance. This suggests that each type of logical reasoning benefits from auxiliary knowledge transferred from other reasoning types, while unconstrained complementary knowledge integration can negatively impact performance.

	Deductive	Inductive	Abductive
Unsteered	76.26	76.26	76.26
Mono S.	78.31	77.75	78.24
Complementary S.	79.37	78.33	78.62

Table 3: Cross-task generalization on GSM8K. We steer Llama-3.1-8B-it with deductive, inductive, and abductive reasoning vectors, respectively, and report answer accuracy. Across all three vector types, both mono steering and complementary steering improve over the unsteered baseline, with complementary steering achieving the best performance.

3.4 Cross-Task Generalization on GSM8K

To evaluate cross-task generalization, we further test the extracted reasoning vectors on GSM8K, a widely used benchmark for multi-step mathematical reasoning. As shown in Table 3, steering with each type of reasoning vector consistently improves over the unsteered baseline, and complementary steering yields further gains over mono steering. Although GSM8K does not isolate a single reasoning type, these results suggest that the learned vectors transfer beyond the original task formulations and retain useful reasoning-related information on an out-of-domain benchmark.

4 Analysis

4.1 Activated Feature Analysis

To understand how reasoning vectors alter internal model representations, we analyze changes in SAE-activated features before and after complementary refinement. Take Llama-3.1-8B-it as an example, we use LlamaScope layer14 residual stream SAE for further fine-grained analysis. We choose layer 14 for analysis because the reasoning vectors are applied at layer 13 to steer model activation (Section 2.1), so we expect the immediate subsequent residual stream to provide the clearest view of the direct representational changes induced by steering.

Core features Analysis. Given a set of residual stream activations collected from layer 14, we encode each activation sequence into the SAE latent space following Equation (4). For each reasoning type, we compute the mean SAE activation vectors before and after refinement and obtain their feature-wise difference:

$$\Delta_r = \bar{z}_r^{\text{ref}} - \bar{z}_r^{\text{orig}}$$

Then, we identify the Top-5 SAE features exhibiting the largest difference in activation for each

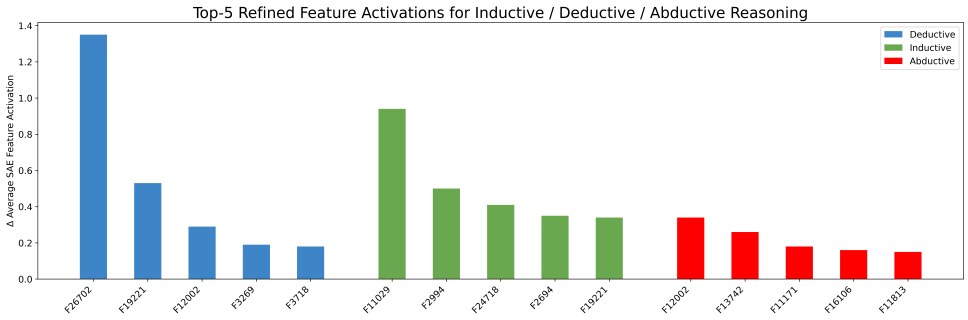


Figure 5: Top-5 refined features for each type of logical reasoning measured by Δ_r based on SAE.

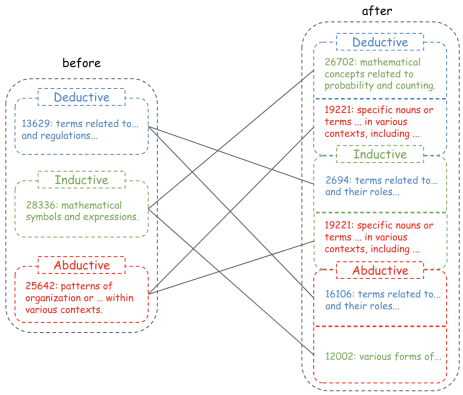


Figure 6: The learned features during complementary refinement process. Take deductive reasoning as an example, $F26702$ and $F19221$ are 2 out of 5 top boosted features after the refinement process from Figure 5. They have similar semantic explanation with $F28336$ and $F25642$, which are two top features for original inductive and abductive reasoning without refinement.

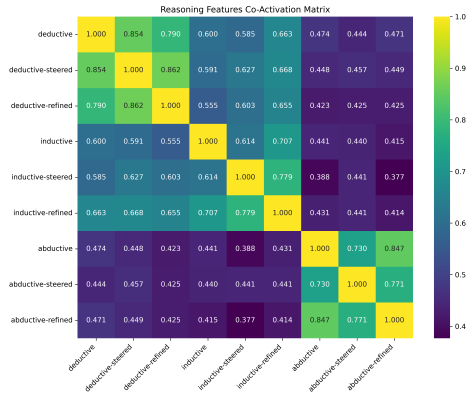


Figure 7: Reasoning Features Co-Activation Matrix.

reasoning type. These features are visualized in Figure 5 to illustrate how refinement shifts the underlying semantic subspaces associated with deductive, inductive, and abductive reasoning. We use Neuronpedia (Lin and Bloom, 2023) to retrieve the feature explanations. For each type of logical reasoning, we observe that the top boosted features share similar semantic relevance to the top features activated in the other two reasoning types. This indicates that our complementary refinement enables each reasoning type to effectively leverage meaningful features learned from the others. The related feature explanations are shown in Figure 6.

Broader features Analysis. While the core-feature analysis focuses on the features most strongly affected by refinement, we additionally investigate how the broader distribution of reasoning-related SAE feature activations changes as the steering or refined-steering intervention added. For each reasoning dataset and model variant (unsteered, mono-steered, and refined-steered), we encode all residual stream activations from layer 14 following Equation (4) and compute the mean activation vector for each variant setting as x. To quantify how similarly two variant settings activate their respective top-k SAE features (k=100 in our experiments), we compute the symmetric co-activation similarity between variant settings i and j as:

$$S(i,j) = \frac{2\sum_k \min(x_i(k),\, x_j(k))}{\sum_k x_i(k) + \sum_k x_j(k)}$$

This measures how strongly the two top-feature sets align in expectation over the SAE space; a value near 1 indicates heavy overlap in activated feature directions, while lower values indicate divergence. We compute this similarity across all nine variant settings (three reasoning types times three model versions), producing a 9×9 co-activation matrix visualized in Figure 7.

As the result shows to us, deductive and inductive reasoning become more aligned in their overall activation distributions, whereas abductive reasoning becomes more differentiated. This trend is reflected in the co-activation scores: on one hand, it is apparent that the deductive–inductive reason-

ing region displays higher co-activation values; on the other hand, from a fine-grained perspective, deductive–inductive similarity rises from 0.600 to 0.655 across variants, while abductive–deductive similarity (0.474 → 0.457 → 0.425) and abductive–inductive (0.441 → 0.441 → 0.414) similarities consistently decline. These patterns are consistent with the fact that deductive and inductive reasoning share an evidence-based inferential structure, whereas abductive reasoning relies more on explanation-based inference.

Former core feature analysis shows that the Top-5 feature shifts of each refined reasoning vector often include semantic cues associated with other reasoning types. This does not contradict the broader divergence observed above because LLM representations are compositional: complementary refinement shapes global reasoning structure while still allowing shared cross-reasoning features that supports complementary reasoning behaviors.

4.2 Reasoning Causal Tracing

In order to better understand the impact of the naive and complementary refined reasoning knowledge vectors, besides the feature level analysis, we further conduct mechanism analysis in models' activation space to trace the differentiated causal traits before and after our complementary refinement.

Activation patching is a mechanistic interpretability technique that measures the causal contribution of specific model components by replacing their activations and observing the resulting change in model behavior (Zhang and Nanda, 2023; Conmy et al., 2023). Here, we following Heimersheim and Nanda (2024), which overwrite specific activations during a model run with cached activations from a previous run and observe how this affects the model's output. We focus on patching the activations of attention heads. Detailed operation and results are provided in Appendix C. We have following observations from Figure 8:

Core activations reserved. The attention heads that were strongly aligned with current reasoning remain consistently active after complementary refinement, such as layer 27 head 29 for abductive reasoning and layer 17 head 24 for inductive reasoning. This persistence indicates that the refinement procedure preserves the core representational components necessary for the targeted reasoning behavior. Notably, certain heads such as layer 31 head 14 remain relatively highly-activated across all three reasoning types, suggesting their role as general-purpose reasoning facilitators.

...Since both conditions support the statement, we can conclude that the statement is true.
...is equivalent to saying "buffalo do not roam the pr airie." Since the paragraph states that the notion "buffalo roam the prairie" can be considered false,...
...but does not mention insecticides or industries that sell them. Therefore, the truth value of the statement is uncertain.
...has three main body parts - head, thorax, and abdomen, then it has three pairs of legs.
...is the deepest part of the ocean and the deepest location on Earth. It is 11,034 meters (36,201 feet) deep, which is almost 7 miles...
...can have two types of sales assistants: those that help the customer and those that keep the space and clothes organized...
...Hypothesis B "Our founder Rachel only uses the PC" seems more plausible because it is possible that...
...have time to wash the existing ones . Hypothesis A, which states that Lina received a pile of adult clothes, seems...
...has to close again tomorrow. This implies that his work schedule is likely a result of his preference for...

Table 4: Representative text spans extracted for deductive, inductive and abductive reasoning.

Activation profile concentrated. The refined-steered model exhibits a more concentrated activation profile: the most influential heads become more localized, while diffuse non-essential activations diminish. In several cases, such as layer 16 head 30 and layer 31 head 14 for inductive reasoning, refinement increases activation strength, implying improved sparsity, specialization, and more efficient utilization of reasoning-relevant circuits.

New activations appear. Refinement gives rise to newly activated attention heads that are not prominent in the naive-steered model. For example, layer 17 head 24 becomes salient for deductive reasoning after refinement, while layer 24 head 27 shows increased activation for both deductive and abductive reasoning. These new heads suggests that refinement not only preserves existing reasoning circuits but also recruits additional components that encode complementary reasoning features.

4.3 Qualitative Analysis

Since prior work reveals that causal connectives such as 'therefore', 'thus', and 'because' explicitly encode inferential relations in text (Sanders and

Stukker, 2012), we investigate whether meaningful knowledge in reasoning vectors actually highlighted specific keywords or phrases in text distributions. Following the text span extraction procedure in Han et al. (2024), we quantify the influence of a reasoning vector by computing the log-likelihood shift for each token by $\Delta_i = \log P_r(v_i \mid v_{<i}) - \log P_0(v_i \mid v_{<i})$ and apply a dynamic programming algorithm to identify the contiguous token span with the highest cumulative shift under length 5, yielding a concise summary of the textual patterns most affected by the reasoning vector. Several extracted representative spans are presented in Table 4. Deductive reasoning vectors tend to emphasize causal connectives such as 'therefore', 'since' and explicit conclusion markers, which are characteristic of rule-based logical reasoning. Inductive reasoning vectors frequently attend to generalization-related phrases, including quantifiers, categorical properties, and statistical regularities, reflecting reasoning from repeated observations. In contrast, abductive reasoning vectors highlight plausibility-oriented expressions such as 'more plausible' and 'likely', which are indicative of hypothesis selection under uncertainty. These findings suggest that the learned reasoning vectors capture semantically meaningful linguistic cues associated with distinct reasoning processes, providing qualitative evidence that the vectors encode interpretable reasoning-related knowledge rather than arbitrary activation patterns.

5 Related Work

Logical Reasoning in LLMs. Logical reasoning includes multiple forms of inference that enable humans and machines to draw conclusions from some premises or observations (Johnson-Laird, 2010). Classical cognitive science distinguishes three primary types of reasoning: deductive, inductive, and abductive (Peirce, 1934), which have increasingly been adopted as frameworks for analyzing the behavior of LLMs (Li et al., 2025). Several works enhance reasoning via strategy selection or hybrid deductive–inductive designs (Cheng et al., 2024; Cai et al., 2025; Wang et al., 2024). However, most of current work focus on external reasoning paths, with limited analysis of internal representations in activation space (Tan et al., 2024; Dumas et al., 2025). In contrast, we study whether logical reasoning can be linearly manipulated to enable controllable and compositional reasoning behaviors (Scalena et al., 2024; Nguyen et al., 2025).

Knowledge Vector. Knowledge vectors (KVs) in LLMs refer to linear representations that encode semantic information in model parameters or activation space (Elhage et al., 2022). Early work on model editing shows that factual knowledge can be localized and manipulated via linear interventions (Meng et al., 2022a,b), and more recent studies identify latent directions corresponding to high-level behaviors such as sentiment and instruction-following (Turner et al., 2023; Stolfo et al., 2024). Despite these advances, reasoning-related vectors remain largely unexplored. Compared to prior work that focuses on isolated behaviors such as chain-of-thought stages or backtracking patterns (Venhoff et al., 2025; Zhang and Viteri), we investigate whether general logical reasoning types correspond to separable directions in activation space.

6 Conclusion

In this work, we show that logical reasoning can be linearly represented as distinct knowledge vectors in LLM activation space and geometric analysis reveals low pairwise cosine similarity among these vectors, suggesting that different reasoning abilities rely on distinct representational mechanisms. Motivated by above observation, we hypothesize that reasoning vectors can be improved by incorporating complementary knowledge across reasoning types and propose a complementary subspace–constrained refinement framework based on SAEs, which preserves each vector's core features while selectively integrating complementary information. Experiments show that the refined vectors consistently outperform both the unsteered and mono-steering baselines, and further multi-perspective analyses reveals the shared and specific features among these reasoning knowledge vectors.

Limitations

One limitation of this work is that our study focuses on three logical reasoning datasets. Our experiments do not explore how the proposed approach generalizes to other reasoning benchmarks, nor do we examine potential cross-domain transfer effects between datasets. In addition, we do not investigate more fine-grained reasoning forms, such as analogical reasoning and counterfactual reasoning, which may involve different representational structures or interaction patterns in model's activation space.

References

Samir Abdaljalil, Hasan Kurban, Khalid Qaraqe, and Erchin Serpedin. 2025. Theorem-of-thought: A multi-agent framework for abductive, deductive, and inductive reasoning in language models. *arXiv preprint arXiv:2506.07106*.

Kingma DP Ba J Adam and 1 others. 2014. A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 1412(6).

Guillaume Alain and Yoshua Bengio. 2016. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Suhana Bedi, Yixing Jiang, Philip Chung, Sanmi Koyejo, and Nigam Shah. 2025. Fidelity of medical reasoning in large language models. *JAMA Network Open*, 8(8):e2526021–e2526021.

Yonatan Belinkov. 2022. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219.

Chandra Bhagavatula, Ronan Le Bras, Chaitanya Malaviya, Keisuke Sakaguchi, Ari Holtzman, Hannah Rashkin, Doug Downey, Scott Wen-tau Yih, and Yejin Choi. 2019. Abductive commonsense reasoning. *arXiv preprint arXiv:1908.05739*.

Chengkun Cai, Xu Zhao, Haoliang Liu, Zhongyu Jiang, Tianfang Zhang, Zongkai Wu, Jenq-Neng Hwang, and Lei Li. 2025. The role of deductive and inductive reasoning in large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16780–16790.

Michael K Chen, Xikun Zhang, and Dacheng Tao. 2025. Justlogic: A comprehensive benchmark for evaluating deductive reasoning in large language models. *arXiv preprint arXiv:2501.14851*.

Kewei Cheng, Jingfeng Yang, Haoming Jiang, Zhengyang Wang, Binxuan Huang, Ruirui Li, Shiyang Li, Zheng Li, Yifan Gao, Xian Li, Bing Yin, and Yizhou Sun. 2024. Inductive or deductive? rethinking the fundamental reasoning abilities of llms. *Preprint*, arXiv:2408.00114.

Arthur Conmy, Augustine Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. 2023. Towards automated circuit discovery for mechanistic interpretability. *Advances in Neural Information Processing Systems*, 36:16318–16352.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. 2023. Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600*.

Clément Dumas, Chris Wendler, Veniamin Veselovsky, Giovanni Monea, and Robert West. 2025. Separating tongue from thought: Activation patching reveals language-agnostic concept representations in transformers. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 31822–31841.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, and 1 others. 2022. Toy models of superposition. *arXiv preprint arXiv:2209.10652*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Lin Guan, Karthik Valmeekam, Sarath Sreedharan, and Subbarao Kambhampati. 2023. Leveraging pretrained large language models to construct and utilize world models for model-based task planning. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Chi Han, Jialiang Xu, Manling Li, Yi Fung, Chenkai Sun, Nan Jiang, Tarek Abdelzaher, and Heng Ji. 2024. Word embeddings are steers for language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16410–16430.

Zhengfu He, Wentao Shu, Xuyang Ge, Lingjie Chen, Junxuan Wang, Yunhua Zhou, Frances Liu, Qipeng Guo, Xuanjing Huang, Zuxuan Wu, and 1 others. 2024. Llama scope: Extracting millions of features from llama-3.1-8b with sparse autoencoders. *arXiv preprint arXiv:2410.20526*.

Stefan Heimersheim and Neel Nanda. 2024. How to use and interpret activation patching. *arXiv preprint arXiv:2404.15255*.

Evan Heit and Caren M Rotello. 2010. Relations between inductive reasoning and deductive reasoning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 36(3):805.

Keith J Holyoak and Robert G Morrison. 2013. *The Oxford handbook of thinking and reasoning*. Oxford University Press.

Zhenglin Hua, Jinghan He, Zijun Yao, Tianxu Han, Haiyun Guo, Yuheng Jia, and Junfeng Fang. 2025. Steering lvlms via sparse autoencoder for hallucination mitigation. *arXiv preprint arXiv:2505.16146*.

Philip N Johnson-Laird. 2010. Mental models and human reasoning. *Proceedings of the National Academy of Sciences*, 107(43):18243–18250.

Ole Jorgensen, Dylan Cope, Nandi Schoots, and Murray Shanahan. 2023. Improving activation steering in language models with mean-centring. *arXiv preprint arXiv:2312.03813*.

Jiabao Kang, Xinye Li, Liyan Xu, Qingbin Liu, Xi Chen, Zhiying Tu, Dianhui Chu, and Dianbo Sui. 2025. Exploring deductive and inductive reasoning capabilities of large language models in procedural planning. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 16320–16341.

Connor Kissane, robertzk, Arthur Conmy, and Neel Nanda. 2024. SAEs (usually) Transfer Between Base and Chat Models. https://www.lesswrong.com/posts/fmwk6qxrpW8d4jvbd/saes-usually-transfer-between-base-and-chat-models.

Long Hei Matthew Lam, Ramya Keerthy Thatikonda, and Ehsan Shareghi. 2024. A closer look at tool-based logical reasoning with llms: The choice of tool matters. In *Proceedings of the 22nd Annual Workshop of the Australasian Language Technology Association*, pages 41–63.

Chunyang Li, Weiqi Wang, Tianshi Zheng, and Yangqiu Song. 2025. Patterns over principles: The fragility of inductive reasoning in llms under noisy observations. *arXiv preprint arXiv:2502.16169*.

Tom Lieberum, Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Nicolas Sonnerat, Vikrant Varma, János Kramár, Anca Dragan, Rohin Shah, and Neel Nanda. 2024. Gemma scope: Open sparse autoencoders everywhere all at once on gemma 2. *arXiv preprint arXiv:2408.05147*.

Johnny Lin and Joseph Bloom. 2023. Neuronpedia: Interactive reference and tooling for analyzing neural networks. *Software available from neuronpedia. org*.

Samuel Marks, Adam Karvonen, and Aaron Mueller. 2024. Dictionary learning. https://github.com/saprmarks/dictionary_learning. GitHub repository.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022a. Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35:17359–17372.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. 2022b. Mass-editing memory in a transformer. *arXiv preprint arXiv:2210.07229*.

Aashiq Muhamed, Jacopo Bonato, Mona T Diab, and Virginia Smith. 2025. Saes can improve unlearning: Dynamic sparse autoencoder guardrails for precision unlearning in llms. In *ICML 2025 Workshop on Reliable and Responsible Foundation Models*.

Neel Nanda. 2023. Actually, othello-gpt has a linear emergent world representation. https://www.neelnanda.io/mechanistic-interpretability/othello.

Neel Nanda and Joseph Bloom. 2022. Transformerlens. https://github.com/TransformerLensOrg/TransformerLens.

Duy Nguyen, Archiki Prasad, Elias Stengel-Eskin, and Mohit Bansal. 2025. Multi-attribute steering of language models via targeted intervention. *arXiv preprint arXiv:2502.12446*.

OpenAI. 2025. gpt-oss-120b & gpt-oss-20b model card. *Preprint*, arXiv:2508.10925.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, and 262 others. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Liangming Pan, Alon Albalak, Xinyi Wang, and William Wang. 2023. Logic-lm: Empowering large language models with symbolic solvers for faithful logical reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 3806–3824.

Kiho Park, Yo Joong Choe, and Victor Veitch. 2023. The linear representation hypothesis and the geometry of large language models. *arXiv preprint arXiv:2311.03658*.

Charles Sanders Peirce. 1934. *Collected papers of charles sanders peirce*, volume 5. Harvard University Press.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. 2024. Steering llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15504–15522.

Ted Sanders and Ninne Stukker. 2012. Causal connectives in discourse: A cross-linguistic perspective. *Journal of pragmatics*, 44(2):131–137.

Daniel Scalena, Gabriele Sarti, and Malvina Nissim. 2024. Multi-property steering of large language models with dynamic activation composition. *arXiv preprint arXiv:2406.17563*.

Dong Shu, Xuansheng Wu, Haiyan Zhao, Daking Rai, Ziyu Yao, Ninghao Liu, and Mengnan Du. 2025. A survey on sparse autoencoders: Interpreting the internal mechanisms of large language models. *arXiv preprint arXiv:2503.05613*.

Alessandro Stolfo, Vidhisha Balachandran, Safoora Yousefi, Eric Horvitz, and Besmira Nushi. 2024. Improving instruction-following in language models through activation steering. *arXiv preprint arXiv:2410.12877*.

Daniel Tan, David Chanin, Aengus Lynch, Brooks Paige, Dimitrios Kanoulas, Adrià Garriga-Alonso, and Robert Kirk. 2024. Analysing the generalisation and reliability of steering vectors. *Advances in Neural Information Processing Systems*, 37:139179–139212.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, and 1 others. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J Vazquez, Ulisse Mini, and Monte MacDiarmid. 2023. Steering language models with activation engineering. *arXiv preprint arXiv:2308.10248*.

Aniket Vashishtha, Qirun Dai, Hongyuan Mei, Amit Sharma, Chenhao Tan, and Hao Peng. 2025. Executable counterfactuals: Improving llms' causal reasoning through code. *arXiv preprint arXiv:2510.01539*.

Constantin Venhoff, Iván Arcuschin, Philip Torr, Arthur Conmy, and Neel Nanda. 2025. Understanding reasoning in thinking language models via steering vectors. *arXiv preprint arXiv:2506.18167*.

Danqing Wang, Jianxin Ma, Fei Fang, and Lei Li. 2024. Typedthinker: Diversify large language model reasoning with typed thinking. *arXiv preprint arXiv:2410.01952*.

Tianlong Wang, Xianfeng Jiao, Yinghao Zhu, Zhongzhi Chen, Yifan He, Xu Chu, Junyi Gao, Yasha Wang, and Liantao Ma. 2025a. Adaptive activation steering: A tuning-free llm truthfulness improvement method for diverse hallucinations categories. In *Proceedings of the ACM on Web Conference 2025*, pages 2562–2578.

Xu Wang, Zihao Li, Benyou Wang, Yan Hu, and Difan Zou. 2025b. Model unlearning via sparse autoencoder subspace guided projections. *arXiv preprint arXiv:2505.24428*.

Zonglin Yang, Li Dong, Xinya Du, Hao Cheng, Erik Cambria, Xiaodong Liu, Jianfeng Gao, and Furu Wei. 2024. Language models as inductive reasoners. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 209–225.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2022. React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*.

Fred Zhang and Neel Nanda. 2023. Towards best practices of activation patching in language models: Metrics and methods. *arXiv preprint arXiv:2309.16042*.

Jason Zhang and Scott W Viteri. Uncovering latent chain of thought vectors in large language models. In *Workshop on Neural Network Weights as a New Data Modality*.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, and 1 others. 2023. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*.

A Datasets

We extract the reasoning knowledge vectors, validate our linear assumption and refine these vectors based on complementary hypothesis using three logical reasoning datasets: JustLogic (Chen et al., 2025), DEER (Yang et al., 2024), and ART (Bhagavatula et al., 2019):

JustLogic (Chen et al., 2025) provides logically structured arguments constructed without relying on external world knowledge, ensuring that models must perform inference rather than recall memorized facts. Each instance consists of a set of formal premises paired with a conclusion that must logically follow. In our experiments, we adopt the official dataset split, which contains 4900 training and 1050 test data points for vector extraction, complementary refinement and evaluation.

DEER (Yang et al., 2024) is a natural-language inductive reasoning dataset in which models are required to infer general rules from concrete textual examples. Each data point contains several observed facts along with a corresponding rule template and the ground truth rule that must be induced. The dataset captures a wide range of natural textual patterns and serves as a realistic benchmark for assessing rule induction. Following the official setup, we use the provided training and test splits (438/762 for train/test) in our experiments.

ART (Bhagavatula et al., 2019) is a large-scale abductive reasoning benchmark focused on commonsense explanation generation. Each example presents an observation pair describing a narrative situation, and models must choose the most plausible intermediate hypothesis that links them. The dataset reflects human-like abductive reasoning by emphasizing plausible explanation inference rather than surface-level pattern matching. The dataset consists of 170K training examples and 1.5K test examples. We sampled 4K instances from the training set and used the entire test split from the original benchmark for all experiments.

B Implementation Details

Since we need the SAEs for our complementary refinement and further analysis, we choose Llama3.1-

8B-it (Grattafiori et al., 2024) and Gemma-2-9B-it (Team et al., 2024) as our evaluation LLMs. And since SAEs trained on the middle-layer residual stream of base models generally transfer well to the corresponding instruction fine-tuning models (Kissane et al., 2024), we choose llama_scope_lxr_8x from Llama Scope (Lieberum et al., 2024) and gemma-scope-9b-pt-res-canonical from Gemma Scope (He et al., 2024) to extract features from Llama3.1-8B-it and Gemma-2-9B-it respectively. For GPT-OSS-20B, we use saes-gpt-oss-20b (Marks et al., 2024).

Based on experience of previous work (Rimsky et al., 2024; Zhang and Viteri), we extract the activations and steering the model on layer 13 for both Llama-3.1-8B-it and Gemma-2-9B-it. Both the activation extraction and knowledge steering is at the generation stage. For each data point, we collected the activations from every token of generated text. And also we perform knowledge steering by adding it at every token position of the generated text after the end of the initial prompt. The max text generation length is set to be 512 tokens.

All experiments are conducted on NVIDA B200 GPUs. The batch size is 16. In subspace extraction, we set $\varepsilon = 1e-6$, $\tau = 0.9$ and $K = 3000$. In complementary vectors extraction, using Adam as the optimizer (Adam et al., 2014), the learning rate is set to be $1e-3$, λ_{com} is $1e-1$ and λ_{sub} is $1e-2$. TransformerLens (Nanda and Bloom, 2022) is used for activation patching.

C Activation Patching

Since we applied the extracted reasoning knowledge vectors at layer 13, we patch the attention heads from layer 14 to the final layer to observe the effects of our knowledge vectors. First, we run the model without any intervention to cache the orignal attention heads' activations. At the intervening run, we follow the steering style and patch with corresponding cached activations at each generated step. For deductive and abductive reasoning, we choose the answer token logit difference as the metric to evaluate the the effect of knowledge vector on specific attention head. Since there's no ground truth answer token for inductive reasoning, we instead use the last token hidden states semantic difference to evaluate it. The results are shown in Figure 8, which illustrating the influence of different reasoning vectors on models' internal circuit.

D Prompt Templates

Following the prompt formats provided in the original papers of each dataset, we construct positive and negative prompt templates for deductive, inductive, and abductive reasoning. These templates are designed to elicit positive and negative activations corresponding to each reasoning type, enabling us to extract reasoning-aligned activation patterns for subsequent vector construction and analysis.

D.1 Deductive Reasoning

Positive. "You are given a paragraph of premises, followed by a statement. Perform deductive logical reasoning with propositional logic on the paragraph to determine the truth value of the statement.
Here is the list of argument forms:
- Modus Ponens
- Modus Tollens
- Hypothetical Syllogism
- Disjunctive Syllogism
- Reductio ad absurdum
- Constructive Dilemma
- Disjunction Elimination
You must answer with either one of the 3 options:
- TRUE: When the premises in the paragraph lead to the statement
- FALSE: When the premises in the paragraph directly contradict the statement
- UNCERTAIN: When the premises in the paragraph neither support nor contradict the statement
Your answer should be short and clear, containing:
1. Do not repeat paragraph content. Instead, give a brief logical reasoning process directly.
2. Final answer: one of "TRUE", "FALSE", or "UNCERTAIN" (on a new line, quoted).
Use only the information in the paragraph. Assume all premises are true.
—
Paragraph: {PARAGRAPH}
Statement: {STATEMENT}
Answer:"
Negative. "You are given a paragraph of premises, followed by a statement. Determine the truth value of the statement based on the paragraph.
You must answer with either one of the 3 options:
- TRUE: When the premises in the paragraph lead to the statement
- FALSE: When the premises in the paragraph directly contradict the statement
- UNCERTAIN: When the premises in the paragraph neither support nor contradict the statement

Your answer should be short and clear, containing:
1. Do not repeat paragraph content.
2. Final answer: one of "TRUE", "FALSE", or "UNCERTAIN" (on a new line, quoted).
Use only the information in the paragraph. Assume all premises are true.
—
Paragraph: {PARAGRAPH}
Statement: {STATEMENT}
Answer:"

D.2 Inductive Reasoning

Positive. "You are a reasoning assistant capable of inductive generalization. Your task is to observe several specific facts and infer a general rule that could have produced them. Focus on finding a pattern that explains all examples and write the rule that satisfies the rule template and the given facts.
Do not include '_' in generation.
Fact: {FACT}
Rule Template: {TEMPLATE}"
Negative. "Generate a rule in the LAST line that satisfies the rule template and the given facts.
Do not include '_' in generation.
Fact: {FACT}
Rule Template: {TEMPLATE}"

D.3 Abductive Reasoning

Positive. "You are an expert in abductive logical reasoning. Given the following two observations, your task is to carefully evaluate two hypotheses and determine which one provides a better causal explanation.
Observations:
1. {OBS1}
2. {OBS2}
Hypotheses:
A. {HYP1}
B. {HYP2}
For each hypothesis, consider whether it explains both observations plausibly and logically. Think carefully and explain your reasoning briefly.
Then, answer the following question:
Which hypothesis (A or B) more plausibly explains the observations?
Format your answer as:
Reasoning: <your reasoning here>
Answer: <A or B>"
Negative. "You are given two observations that describe a situation, and two hypotheses that attempt to explain what happened.

Inductive	Deductive	Abductive
27.55±0.41	56.46±0.85	40.95±0.62

Table 5: Summary of sensitivity analysis results on Llama-3.1-8B-it over different choices of λ_{com} and λ_{sub}. We report the mean and standard deviation of complementary-steering performance across all tested settings.

Your task is to determine which hypothesis is more plausible given the two observations.
Observations:
1. {OBS1}
2. {OBS2}
Hypotheses:
A. {HYP1}
B. {HYP2}
Explain your reasoning briefly.
Then, answer the following question:
Which hypothesis (A or B) more plausibly explains the observations?
Format your answer as:
Reasoning: <your reasoning here>
Answer: <A or B>"

E Sensitivity Analysis

To examine the robustness of the refinement objective to the choice of hyperparameters, we conduct a sensitivity analysis on Llama-3.1-8B-it by varying the two loss weights in Equation 10. Specifically, we consider $\lambda_{\text{com}} \in \{10^{-2}, 10^{-1}, 1.0\}$ and $\lambda_{\text{sub}} \in \{10^{-1}, 10^{-2}, 10^{-3}\}$, and summarize the resulting complementary-steering performance in Table 5. Across these settings, performance varies only marginally for deductive, inductive, and abductive reasoning, indicating that the method operates in a relatively stable regime rather than depending on a sharp hyperparameter optimum.

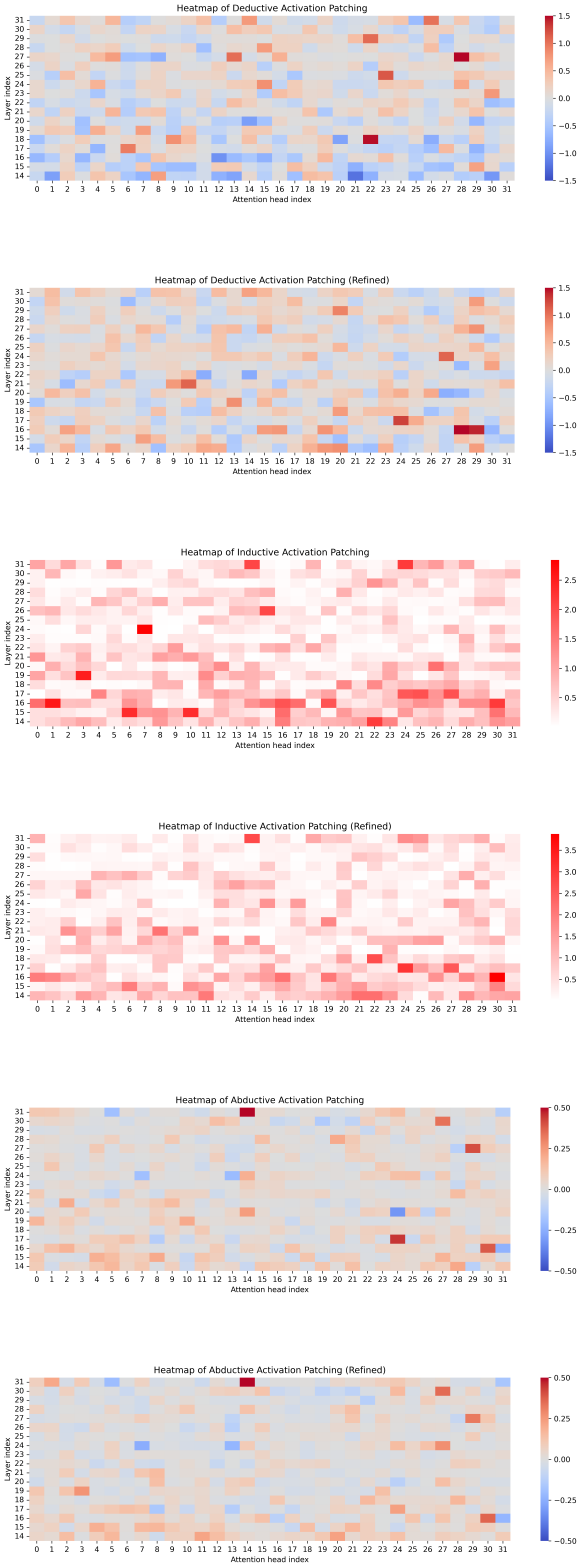


Figure 8: Activation patching heatmaps. The top-to-bottom panels show naive steering followed by refined steering for deductive, inductive, and abductive reasoning. Each unit denotes the logit difference of the answer token or the semantic difference in hidden states between patched and unpatched executions.