Is This the Subspace You Are Looking for? An Interpretability Illusion for Subspace Activation Patching

Aleksandar Makelov[*]
SERI MATS
aleksandar.makelov@gmail.com

Georg Lange[*]
SERI MATS
georglange4@gmail.com

Atticus Geiger
Linguistics Department
Stanford University
Stanford, CA 94305, USA
atticusg@stanford.edu

Neel Nanda
London, UK
neelnanda27@gmail.com

Abstract

Mechanistic interpretability aims to understand model behaviors in terms of specific, interpretable features, often hypothesized to manifest as low-dimensional subspaces of activations. Specifically, recent studies have explored subspace interventions (such as activation patching) as a way to both manipulate model behavior and attribute the features behind it to given subspaces. In this work, we demonstrate that these two aims diverge, potentially leading to an illusory sense of interpretability. Counterintuitively, even if a subspace intervention modifies end-to-end model behavior in the desired way, this effect may be achieved by activating a *dormant parallel pathway* leveraging a component that is *causally disconnected* from model outputs. We demonstrate this phenomenon in a distilled mathematical example, in two real-world domains (the indirect object identification task and factual recall), and present evidence for its prevalence in practice. In the context of factual recall, we further show a link to rank-1 fact editing, providing a mechanistic explanation for previous work observing an inconsistency between fact editing performance and fact localization. Finally, we remark on what a success case of subspace activation patching looks like.

1 Introduction

The growing capabilities of large language models [Vaswani et al., 2017, Radford et al., 2019, Brown et al., 2020, Wei et al., 2022, OpenAI, 2023] demand a deeper understanding of what they learn and how they make predictions. This is the realm of machine learning interpretability [Lipton, 2016]; within it, mechanistic interpretability (MI) focuses on a rigorous low-level understanding of models through the lens of task-specific algorithms, or 'circuits' [Olah et al., 2020, Wang et al., 2023, Olah, 2022, Vig et al., 2020], operating on concrete building blocks akin to variables in a computer program [Olah, 2022, Geiger et al., 2023a]. A key issue in MI is defining and discovering these building blocks. To this end, *activation patching* [Vig et al., 2020, Geiger et al., 2020] – which forces specific activations to take on the value they would take on a different input and examines how this effects model behavior – has been widely used as an interpretability tool for causally attributing behaviors to specific model components (attention heads, MLP layers Wang et al. [2023], Heimersheim and Janiak,

[0]*Equal contribution.

ATTRIB: Workshop on Attributing Model Behavior at Scale at NeurIPS 2023.

Meng et al. [2022a]) and, increasingly, to low-dimensional *subspaces* of components [Geiger et al., 2023b, Wu et al., 2023, Nanda et al., 2023]. We refer the reader to Appendix C for a more detailed discussion of related work. While such subspace interventions have promise for interpretability, we show that they are prone to a kind of *interpretability illusion*. Specifically, instead of robustly localizing a variable that is used by the model in a wide range of contexts, setting the value of a subspace to that of another example can fabricate such a variable by activating a dormant pathway in the model via exploiting a causally disconnected feature (Figure 1). Our contributions can be summarized as follows.

Mathematical example. In Section 2, we construct a distilled mathematical example of the illusion.

Empirical realizations. In Section 3, we find a realization of the illusion in the context of the indirect object identification task [Wang et al., 2023], where a 1-dimensional subspace of MLP activations found using DAS [Geiger et al., 2023b] can seem to encode position information about names in the sentence, despite this MLP layer having significantly smaller contribution as a whole.

In Section 4 we also exhibit this phenomenon in the setting of *fact editing* [Meng et al., 2022a]. We show that 1-dimensional activation patches imply equivalent rank-1 model edits [Meng et al., 2022a]. In particular, this shows that rank-1 model edits can also be achieved by creating a new pathway in the model, without relying on the presence of a fact in the weight being edited. This suggests a mechanistic explanation for the observation of [Hase et al., 2023] that rank-1 model editing works regardless of whether the fact is present in the weights being edited.

Reasons to expect the illusion in general. In Section 5, we end with arguments and evidence for why this interpretability illusion ought to be prevalent in real-world language models.

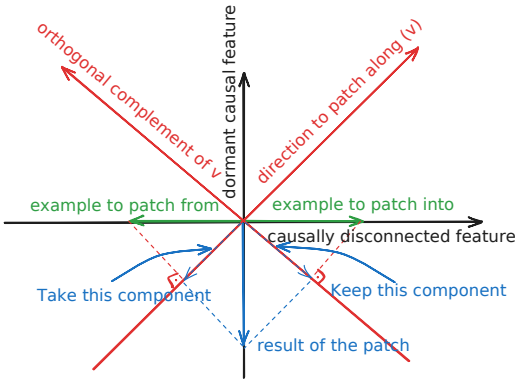


Figure 1: The key mathematical phenomenon behind the activation patching illusion. By setting the projection of an example's activation (green, right) along a vector (red, top-right) to equal another's (green, left) projection, we obtain a vector orthogonal to both activations. This can give counterintuitive results when the original and new directions have fundamentally different roles in a model's computation.

2 A Conceptual View of the Illusion

Activation patching. *Activation patching* [Vig et al., 2020, Geiger et al., 2020, Wang et al., 2023, Chan et al., 2022] is an interpretability technique that intervenes upon model components, forcing them to take on values they would have taken if a different input were provided. For instance, consider a model that has knowledge of the locations of famous landmarks, and completes e.g. the sentence $A =$ 'The Eiffel Tower is in' with 'Paris'. How can we find which component of the model is responsible for knowing that 'Paris' is the right completion?

Activation patching approaches this question by (i) running the model on A, (ii) storing the activation of a chosen component $\mathcal{C}$, and (iii) running the model on e.g. $B =$ 'The Colosseum is in', *but* with the activation of $\mathcal{C}$ taken from A. If we find that the model outputs 'Paris' instead of 'Rome' in step (iii), this suggests that component $\mathcal{C}$ is important for the task of recalling the location of a landmark.

Subspace Activation Patching. The linear representation hypothesis (see Appendix C for background) proposes that *linear subspaces* of vectors will be the most interpretable model components. To search for such subspaces, we can adopt a natural generalization of full component activation patching which only patches the values of a subspace U (while leaving the projection on its orthogonal complement $U^{\perp}$ unchanged). This was proposed in Geiger et al. [2023b], and closely related variants appear in Turner et al. [2023], Nanda et al. [2023], Lieberum et al. [2023].

For the purposes of exposition, we now restrict our discussion to activation patching of a 1-dimensional subspace (i.e. a *direction*) represented by a unit vector $\mathbf{v}$. We remark that the illusion also applies to higher-dimensional subspaces (see Appendix E.1 for details). If $\mathbf{act}_A, \mathbf{act}_B \in \mathbb{R}^d$ are

the activations of a model component $\mathcal{C}$ on examples A, B and $p_A = \mathbf{v}^\top \mathbf{act}_A, p_B = \mathbf{v}^\top \mathbf{act}_B$ are their projections along $\mathbf{v}$, patching from A into B along $\mathbf{v}$ results in the patched activation

$$\mathbf{act}_B^{\text{patched}} = \mathbf{act}_B + (p_A - p_B)\mathbf{v}. \tag{1}$$

Intuition for the illusion. When will the update in Equation 1 change the model's output in the intended way? Intuitively, two properties are necessary: $\mathbf{v}$ must be activated differently by the two prompts (otherwise $p_A \approx p_B$ and the patch has no effect), and $\mathbf{v}$ must be causally connected to the model's outputs (otherwise, if e.g. $\mathbf{v}$ is in the nullspace of downstream model components, changing the activation along $\mathbf{v}$ won't change model predictions). A direction $\mathbf{v}$ faithful to the model's computation will simultaneously have these two properties.

The crux of the illusion is that $\mathbf{v}$ may obtain each of the two properties from two unrelated directions in activation space, as shown in Figure 1. Specifically, we can form $\mathbf{v} = \mathbf{v}_{\text{disconnected}} + \mathbf{v}_{\text{dormant}}$, where $\mathbf{v}_{\text{disconnected}}$ distinguishes between the two prompts, but is in the nullspace of all downstream model components; and $\mathbf{v}_{\text{dormant}}$ can *in principle* steer the model in the way intended by the patch, but is not activated differently by the two prompts. By patching along the sum of these directions, the variation in the disconnected part activates the dormant part, which then achieves the causal effect.

Making the illusion concrete. We refer the reader to Appendix E.3 for a formalization of the concepts of 'causally disconnected' and 'dormant' subspaces, and to Appendix E.4 for a concrete mathematical realization of the illusion in a linear neural network with a single hidden layer.

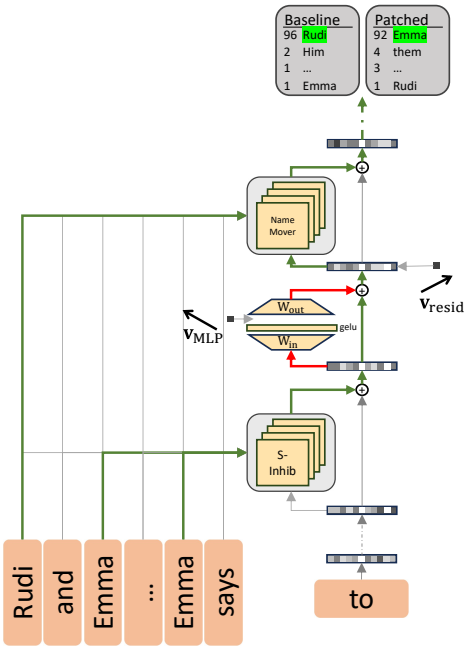


Figure 2: Schematic of the IOI circuit and key interventions. GPT2-Small predicts the correct name by S-inhibition heads writing positional information to the residual stream which is used by the name movers to copy the non-duplicated name (green arrows). Location of subspace interventions $\mathbf{v}_{\text{resid}}$ and $\mathbf{v}_{\text{MLP}}$ are marked. Patching the illusory subspace $\mathbf{v}_{\text{MLP}}$ adds a new path (red) along the established one that is used to flip positional information when patched.

3 The Illusion in the Indirect Object Identification Task

In Wang et al. [2023], the authors analyze how the decoder-only transformer language model GPT-2 Small [Radford et al., 2019] performs the *indirect object identification* task. In this task, the model is required to complete sentences of the form 'When Mary and John went to the store, John gave a bottle of milk to' (with the intended completion in this case being ' Mary'). We refer to the repeated name (John) as **S** (the subject) and the non-repeated name (Mary) as **IO** (the indirect object). Additional details on the data distribution, model and task performance are given in Appendix F.1.

Wang et al. [2023] suggest the model uses the algorithm 'Find the two names in the sentence, detect the repeated name, and predict the non-repeated name' to do this task. In particular, they find a set of four heads in layers 7 and 8 – the **S-Inhibition heads** – that output the signal responsible for *not* predicting the repeated name. The dominant part of this signal is of the form 'Don't attend to the name in first/second position in the first sentence' depending on where the **S** name appears (see Appendix A in Wang et al. [2023] for details). This signal is added to the residual stream [1] at the last token position, and is then picked up by another class of heads in layers 9, 10 and 11 – the **Name**

[1] We follow the conventions of Elhage et al. [2021] when describing internals of transformer models. The residual stream at layer k is the sum of the output of all layers up to $k-1$, and is the input into layer k.

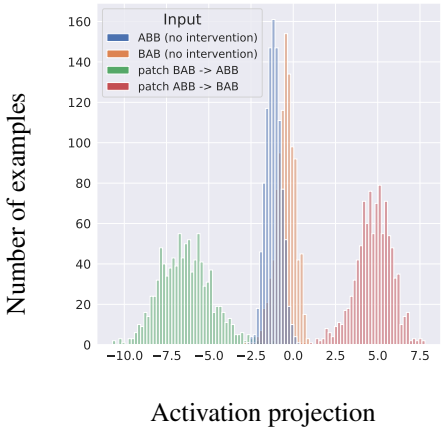


Figure 3: Projections of the output of the MLP layer on the gradient direction $\mathbf{v}_{\text{grad}}$ before (blue/orange) and after (green/red) the activation patch along $\mathbf{v}_{\text{MLP}}$. Here, 'ABB'/'BAB' denotes prompts where the **IO** name comes first/second.

Patching subspace	Frac. logit diff lost	Interchange accuracy
full MLP	-8%	0.4%
$\mathbf{v}_{\text{MLP}}$	46.7%	4.7%
$\mathbf{v}_{\text{MLP}}$ nullspace$^{\perp}$	13.5%	0.7%
$\mathbf{v}_{\text{MLP}}$ nullspace	0%	0.5%
full residual stream	123.6%	55.3%
$\mathbf{v}_{\text{resid}}$	140.7%	75.3%
$\mathbf{v}_{\text{resid}}$ nullspace$^{\perp}$	127.5%	63.6%
$\mathbf{v}_{\text{resid}}$ nullspace	13.9%	0.9%
$\mathbf{v}_{\text{grad}}$	111.5%	45.6%
$\mathbf{v}_{\text{grad}}$ nullspace$^{\perp}$	106.47%	41.1%
$\mathbf{v}_{\text{grad}}$ nullspace	2.2%	0.5%

Table 1: Effects of activation patching of full components and 1-dimensional subspaces on the IOI task: fractional logit difference lost (higher means more successful patch; 0% means no change) and interchange accuracy (fraction of predictions flipped; higher means more successful patch).

Mover heads – which use it to shift attention to the **IO** name and copy it to the last token position, so that it can be predicted (Figure 2).

3.1 Finding Subspaces Mediating Name Position Information

How, precisely, is the positional signal communicated? In particular, 'don't attend to the first/second name' is plausibly a binary feature represented by a 1-dimensional subspace. In this subsection, we present methods to look for such a subspace.

Gradient of name mover attention scores. As shown in Wang et al. [2023], the three name mover heads identified therein will attend to one of the names, and the model will predict whichever name is attended to. The position feature matters mechanistically by determining whether they attend to **IO** over **S**. This motivates us to consider the gradient $\mathbf{v}_{\text{grad}}$ of the difference of attention scores of these heads on the **S** and **IO** names with respect to the residual stream after layer 8. This gradient is the direction that maximally shifts attention between the two names (per unit ℓ_2 norm), so we expect it to be a strong mediator of the position signal. Implementation details are given in Appendix F.2.

Distributed alignment search. We can also directly optimize for a direction that mediates the position signal. This is the approach taken by DAS [Geiger et al., 2023b]. In our context, DAS optimizes for an activation subspace which, when activation patched from prompt B into prompt A, makes the model behave as if the relative position of the **IO** and **S** names in the sentence is as in prompt B. This approach is based purely on the model's predictions, and does not make any assumptions about its internal computations. We let $\mathbf{v}_{\text{MLP}}$ and $\mathbf{v}_{\text{resid}}$ be 1-dimensional subspaces found by DAS in the layer 8 MLP activations and layer 8 residual stream output at the last token, respectively (see Figure 2). These locations are chosen to be between the S-Inhibition and Name Mover heads; however, Wang et al. [2023] did not find any significant contribution from the MLP layer, making it a potential location for our illusion. Implementation details are given in Apendix F.

3.2 Demonstrating the Illusion for the $\mathbf{v}_{\text{MLP}}$ Direction

We now show that patching the $\mathbf{v}_{\text{MLP}}$ direction exhibits the illusion from Section 2. By contrast, we revisit $\mathbf{v}_{\text{grad}}$ and $\mathbf{v}_{\text{resid}}$ in Appendix D, where we show that both are representations of the name position information that are highly faithful to the model's computation.

Methodology. In this section, we perform all patches between examples that only differ in the variable we want to localize in the model, i.e. the position of the **S** and **IO** names in the first sentence. That is, we patch from e.g. 'Then, Mary and John went to the store. John gave a book to' into 'Then,

John and Mary went to the store. John gave a book to', and vice-versa. Let x_{source} be the prompt we take activations from and x_{base} the prompt we patch into. For each subspace/component activation patch, we measure (1) the **interchange accuracy** (higher = more successful patch), which is the fraction of patches for which the model predicts the **S** (i.e., wrong) name for the patched run; (2) the **fractional logit difference lost** (higher = more successful patch), which is the average **decrease** in the model's confidence on the task (as measured by the difference between the logits of the **IO** and **S** name) as a result of the patch, measured as a fraction of the logit difference on the clean run on x_{base}. This is a soft counterpart of the 0-1 based accuracy metric.

Results. Metrics are shown in Table 1. In particular, we exhaustively confirm the mechanics of the illusion are at play.

THE CAUSALLY DISCONNECTED COMPONENT OF $\mathbf{v}_{\text{MLP}}$ DRIVES THE EFFECT: while patching the $\mathbf{v}_{\text{MLP}}$ direction has a significant effect on the logit difference lost (46.7%), this effect is greatly diminished when we remove the component of $\mathbf{v}_{\text{MLP}}$ in $\ker W_{out}$ whose activations are (provably) causally disconnected from model predictions (13.5%), or when we patch the entire MLP activation (-8%, actually increasing confidence). By contrast, performing analogous ablations on $\mathbf{v}_{\text{resid}}$ leads to very similar numbers ($140.7\%/127.5\%/123.6\%$; we refer the reader to Appendix D for details on the $\mathbf{v}_{\text{resid}}$ experiments). We note that the contribution of $\mathbf{v}_{\text{MLP}}$ to model outputs is only through the W_{out} matrix; in particular, a related way of looking at ablating the $\ker W_{out}$ component of $\mathbf{v}_{\text{MLP}}$ is to instead activation-patch the subspace $W_{out}\mathbf{v}_{\text{MLP}}$ in the output of the MLP layer (which obtains similar results).

PATCHING $\mathbf{v}_{\text{MLP}}$ ACTIVATES A DORMANT PATHWAY THROUGH THE MLP: in Figure 3, we plot the projection of the MLP layer's contribution to the residual stream on the gradient direction $\mathbf{v}_{\text{grad}}$ before and after patching, in order to see how it contributes to the attention of name mover heads. We observe that in the absence of intervention, the MLP output is weakly sensitive to the name position information, whereas after the patch this changes significantly. Further validations of the illusion are provided in Appendix F.5, where we show that the nullspace component of $\mathbf{v}_{\text{MLP}}$ is substantial and much more correlated with position information, and in Appendix F.6, where we show that we can find a direction with these properties even if we replace the MLP weights with random matrices. While the contribution of the $\mathbf{v}_{\text{MLP}}$ patch to logit difference may appear relatively small, in Appendix F.4 we argue that this is significant for a single component.

4 Factual Recall

In this section, we show that the interpretability illusion can also be exhibited for the factual recall capability of language models, a much broader setting than the IOI task. We further show that the illusory subspace implies an equivalent rank-one edit (in the sense of Meng et al. [2022a]) to the weights that changes the recalled fact. This provides a simple mechanistic explanation for the observation (see e.g. [Hase et al., 2023]) that fact editing seems to work even in layers where the fact is supposedly not stored. Specifically, as we discuss in Section 5, we expect that in practice there will be many MLP layers where the conditions of our illusion are met – and rank-one fact edits will exist in all these MLP layers, regardless of whether they are responsible for recalling the fact being edited.

4.1 Finding Illusory 1-Dimensional Patches for Factual Recall

Given a fact expressed as a subject-relation-object triple (s, r, o) (e.g., $s =$ 'Eiffel Tower', $r =$ 'is in', $o =$ 'Paris'), we say that a model M *recalls* the fact (s, r, o) if M completes a prompt expressing just the (s, r) pair (e.g., 'The Eiffel Tower is in') with o. Let us be given two facts (s, r, o) and (s', r, o') for the same relation that a model recalls correctly, with corresponding factual prompts A expressing (s, r) and B expressing (s', r) (e.g., $r =$ 'is in', $A =$ 'The Eiffel Tower is in', $B =$ 'The Colosseum is in'). In this subsection, we patch from B into A, with the goal of changing the model's output from o to o'. Implementation details are given in Appendix H.1.

Results are shown in figure 4. We find a stronger version of the same qualitative phenomena as in the IOI illusory direction: (i) the directions we find have a strong causal effect (successfully changing o to o'), but (ii) this effect disappears when we ablate the component in the nullspace of W_{out}, and (iii) patching the entire MLP activation instead has a negligible effect on the difference in logits between the correct and incorrect objects. Further experiments confirming the illusion are in Appendix H.2.

4.2 1-Dimensional Fact Patches Imply Equivalent Rank-1 Fact Edits

Next, we show that the existence of an activation patch as in Subsection 4.1 implies the existence of a different kind of intervention with a similar effect: a *rank-one model edit* to the weights of the MLP layer. Proposed in Meng et al. [2022a], a rank-one model edit updates the W_{out} weight of a single MLP layer to $W'_{out} = W_{out} + \mathbf{a}\mathbf{b}^\top$ for some $\mathbf{a} \in \mathbb{R}^{d_{\text{resid}}}, \mathbf{b} \in \mathbb{R}^{d_{\text{MLP}}}$.

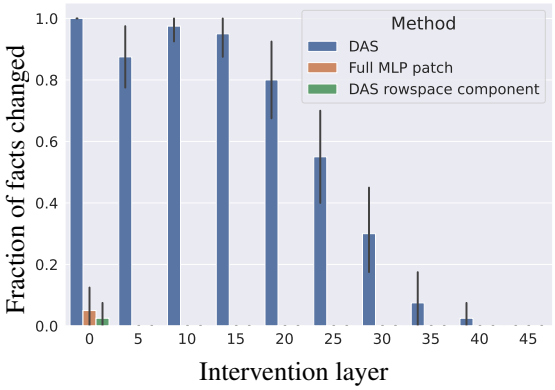


Figure 4: Fraction of successful fact patches under three interventions: patching along the direction found by DAS (blue), patching the component of the DAS direction in the rowspace of W_{out} (green), and patching the entire hidden MLP activation (orange).

In Meng et al. [2022a], the authors also propose a specific kind of rank-one model edit, abbreviated ROME, whose goal is to make a model that recalls the fact (s, r, o) recall (s, r, o') instead, while minimally modifying the model otherwise. The edit takes a vector $\mathbf{k} \in \mathbb{R}^{d_{\text{MLP}}}$ representing the subject (e.g., an average of its last-token MLP post-gelu activations) and a vector $\mathbf{v} \in \mathbb{R}^{d_{\text{resid}}}$ which, when output by the MLP layer, will cause the model to predict the new object o' for the factual prompt (together with some other conditions); see Appendix H.3 for details.

Intuitively, a 'fact patch' as in Subsection 4.1 should have a corresponding rank-1 edit with the same effect: the last subject token MLP activation $\mathbf{u}_A$ for prompt A takes the role of $\mathbf{k}$, and the patch modifies the MLP's output (making it $\mathbf{v}$) to change the model's output to o'. We make this intuition formal in Appendix H.5, where we show that for each 1-dimensional activation patch in the post-gelu activations of an MLP layer, there is a rank-1 model edit to W_{out} that results in the same change to the MLP layer's output at the token where we do the patching.

While this shows that the patch implies a rank-1 edit with the same behavior *at the token being patched*, the rank-1 edit is applied *permanently* to the model, which means that it (unlike the activation patch) applies to *every* token. Thus, it is not a priori obvious whether the rank-1 edit will still succeed in making the model predict o' instead of o. To this end, in Appendix H.6, we evaluate empirically how using the rank-1 edit derived in Appendix H.5 instead of the activation patch changes model predictions, and we find negligible differences.

5 Discussion and Conclusion

Do we expect this illusion to be prevalent? We only exhibit our illusion empirically in two settings, IOI and factual recall, but we believe it is likely prevalent in practice. Specifically, we expect the illusion to occur whenever we have an MLP M which is not used in the model's computation on a given task, but is between two components A and B which *are* used, and communicate through the direction $\mathbf{v}$ via the skip connections of intervening layers. This structure has been frequently observed in the mechanistic interpretability literature [Lieberum et al., 2023, Wang et al., 2023, Olsson et al., 2022, Geva et al., 2021]: circuits contain components composing with each other separated by multiple layers, and circuits have often been observed to be sparse, with most components (including most MLP layers) not playing a significant role.

Under a linear view of features, such a setup likely gives rise to a dormant direction $\mathbf{v}_{\text{dormant}}$ with causal effect. Picking an MLP hidden activation $\mathbf{u} \in \mathbb{R}^{d_{\text{MLP}}}$ such that $W_{out}\mathbf{u} = \mathbf{v}$ gives the causal part, and is always possible as W_{out} is empirically full rank (see Appendix I.1). Furthermore, we can choose $\mathbf{v}_{\text{dormant}} = W^+_{out}\mathbf{v}$; under the assumption that the MLP layer does not participate in the task, the activation along this direction should not change as the activation along $\mathbf{v}$ changes in the residual stream. On the other hand, $\mathbf{u} \in \mathbb{R}^{d_{\text{MLP}}}$ will be correlated with $\mathbf{v}$ if projections of the hidden activations $\mathrm{gelu}(W_{in}\mathbf{x}_{\text{resid}})$ on $\mathbf{u}$ 'track well' the projection of $\mathbf{x}_{\text{resid}}$ on $\mathbf{v}$. We find empirical evidence for this in Appendix I.2, suggesting that $\mathbf{v}_{\text{disconnected}}$ will also exist.

Takeaways and recommendations. Optimization-based methods using subspace activation patching can find both faithful (see Appendix D) and illusory features with respect to the model's computation. We recommend running such methods in activation bottlenecks such as the residual stream, as well as using validations beyond end-to-end evaluation to ascertain the precise role of such features.

References

Julius Adebayo, Justin Gilmer, Michael Muelly, Ian J. Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 9525–9536, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/294a8ed24b1ad22ec2e7efea049b8737-Abstract.html.

Sanjeev Arora, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. Linear algebraic structure of word senses, with applications to polysemy. *Transactions of the Association for Computational Linguistics*, 6:483–495, 2018.

Tolga Bolukbasi, Adam Pearce, Ann Yuan, Andy Coenen, Emily Reif, Fernanda Viégas, and Martin Wattenberg. An interpretability illusion for BERT. *arXiv preprint arXiv:2104.07143*, 2021.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. Discovering latent knowledge in language models without supervision. *arXiv preprint arXiv:2212.03827*, 2022.

Lawrence Chan, Adria Garriga-Alonso, Nicholas Goldowsky-Dill, Ryan Greenblatt, Jenny Nitishinskaya, Ansh Radhakrishnan, Buck Shlegeris, and Nate Thomas. Causal scrubbing: a method for rigorously testing interpretability hypotheses, 2022. URL https://www.alignmentforum.org/posts/JvZhhzycHu2Yd57RN/causal-scrubbing-a-method-for-rigorously-testing.

Roi Cohen, Eden Biran, Ori Yoran, Amir Globerson, and Mor Geva. Evaluating the ripple effects of knowledge editing in language models. *arXiv preprint arXiv:2307.12976*, 2023.

Arthur Conmy, Augustine N Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards automated circuit discovery for mechanistic interpretability. *arXiv preprint arXiv:2304.14997*, 2023.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2126–2136, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1198. URL https://aclanthology.org/P18-1198.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8493–8502, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.581. URL https://aclanthology.org/2022.acl-long.581.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. URL https://transformer-circuits.pub/2021/framework/index.html.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. Toy models of superposition. *Transformer Circuits Thread*, 2022. URL https://transformer-circuits.pub/2022/toy_model/index.html.

Atticus Geiger, Kyle Richardson, and Christopher Potts. Neural natural language inference models partially embed theories of lexical entailment and negation. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 163–173, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.blackboxnlp-1.16. URL https://www.aclweb.org/anthology/2020.blackboxnlp-1.16.

Atticus Geiger, Hanson Lu, Thomas Icard, and Christopher Potts. Causal abstractions of neural networks. *Advances in Neural Information Processing Systems*, 34:9574–9586, 2021.

Atticus Geiger, Christopher Potts, and Thomas Icard. Causal abstraction for faithful model interpretation. Ms., Stanford University, 2023a. URL https://arxiv.org/abs/2301.04709.

Atticus Geiger, Zhengxuan Wu, Christopher Potts, Thomas Icard, and Noah D Goodman. Finding alignments between interpretable causal variables and distributed neural representations. *arXiv preprint arXiv:2303.02536*, 2023b.

Robert Geirhos, Roland S Zimmermann, Blair Bilodeau, Wieland Brendel, and Been Kim. Don't trust your eyes: on the (un) reliability of feature visualizations. *arXiv preprint arXiv:2306.04719*, 2023.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495, Online and Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.446. URL https://aclanthology.org/2021.emnlp-main.446.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. Dissecting recall of factual associations in auto-regressive language models. *arXiv preprint arXiv:2304.14767*, 2023.

Nicholas Goldowsky-Dill, Chris MacLeod, Lucas Sato, and Aryaman Arora. Localizing model behavior with path patching. *arXiv preprint arXiv:2304.05969*, 2023.

Peter Hase, Mohit Bansal, Been Kim, and Asma Ghandeharioun. Does localization inform editing? surprising differences in causality-based localization vs. knowledge editing in language models. *arXiv preprint arXiv:2301.04213*, 2023.

Stefan Heimersheim and Jett Janiak. A circuit for Python docstrings in a 4-layer attention-only transformer. URL https://www.alignmentforum.org/posts/u6KXXmKFbXfWzoAXn/a-circuit-for-python-docstrings-in-a-4-layer-attention-only.

Evan Hernandez, Belinda Z. Li, and Jacob Andreas. Inspecting and editing knowledge representations in language models, 2023.

John Hewitt and Christopher D Manning. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4129–4138, 2019.

Tom Lieberum, Matthew Rahtz, János Kramár, Geoffrey Irving, Rohin Shah, and Vladimir Mikulik. Does circuit analysis interpretability scale? evidence from multiple choice capabilities in chinchilla. *arXiv preprint arXiv:2307.09458*, 2023.

Zachary C. Lipton. The mythos of model interpretability, 2016. URL https://arxiv.org/abs/1606.03490.

Kevin Meng, David Bau, Alex J Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, 2022a.

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

Tomás Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 3111–3119, 2013. URL https://proceedings.neurips.cc/paper/2013/hash/9aa42b31882ec039965f3c4923ce901b-Abstract.html.

Neel Nanda. Attribution patching: Activation patching at industrial scale, 2023. URL https://www.neelnanda.io/mechanistic-interpretability/attribution-patching.

Neel Nanda and Joseph Bloom. Transformerlens. https://github.com/neelnanda-io/TransformerLens, 2022.

Neel Nanda, Andrew Lee, and Martin Wattenberg. Emergent linear representations in world models of self-supervised sequence models. *arXiv preprint arXiv:2309.00941*, 2023.

Chris Olah. Mechanistic interpretability, variables, and the importance of interpretable bases. *https://www.transformer-circuits.pub*, 2022. URL https://www.transformer-circuits.pub/2022/mech-interp-essay.

Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. Zoom in: An introduction to circuits. *Distill*, 2020. doi: 10.23915/distill.00024.001.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. In-context learning and induction heads. *Transformer Circuits Thread*, 2022. URL https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/index.html.

OpenAI. Gpt-4 technical report, 2023.

Judea Pearl. Direct and indirect effects. In *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence*, UAI'01, page 411–420, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc. ISBN 1558608001.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

P. Smolensky. Neural and conceptual interpretation of PDP models. In *Parallel Distributed Processing: Explorations in the Microstructure, Vol. 2: Psychological and Biological Models*, page 390–431. MIT Press, Cambridge, MA, USA, 1986. ISBN 0262631105.

Alessandro Stolfo, Yonatan Belinkov, and Mrinmaya Sachan. Understanding arithmetic reasoning in language models using causal mediation analysis. *arXiv preprint arXiv:2305.15054*, 2023.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1452. URL https://aclanthology.org/P19-1452.

Alex Turner, Lisa Thiergart, David Udell, Gavin Leech, Ulisse Mini, and Monte MacDiarmid. Activation addition: Steering language models without optimization. *arXiv preprint arXiv:2308.10248*, 2023.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Simas Sakenis, Jason Huang, Yaron Singer, and Stuart Shieber. Causal mediation analysis for interpreting neural nlp: The case of gender bias. URL https://arxiv.org/abs/2004.12265.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Simas Sakenis, Jason Huang, Yaron Singer, and Stuart Shieber. Causal mediation analysis for interpreting neural nlp: The case of gender bias. *arXiv preprint arXiv:2004.12265*, 2020.

Jonas Wallat, Jaspreet Singh, and Avishek Anand. BERTnesia: Investigating the capture and forgetting of knowledge in BERT. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 174–183, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.blackboxnlp-1.17. URL https://aclanthology.org/2020.blackboxnlp-1.17.

Kevin Ro Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in GPT-2 small. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=NpsVSN6o4ul.

Ziheng Wang, Jeremy Wohlwend, and Tao Lei. Structured pruning of large language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6151–6162, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.496. URL https://aclanthology.org/2020.emnlp-main.496.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022.

Zhengxuan Wu, Atticus Geiger, Christopher Potts, and Noah D Goodman. Interpretability at scale: Identifying causal mechanisms in alpaca. *arXiv preprint arXiv:2305.08809*, 2023.

Zexuan Zhong, Zhengxuan Wu, Christopher D. Manning, Christopher Potts, and Danqi Chen. Mquake: Assessing knowledge editing in language models via multi-hop questions, 2023.

A Acknowledgments

AM and GL did this work as part of the SERI MATS independent research program, with support from AI Safety Support. The authors would like to thank Tom Lieberum, Senthooran Rajamanoharan, Christopher Potts, Curt Tigges, Oskar Hollingsworth and Peli Grietzer for valuable feedback and discussions. The authors extensively used the open-source library `transformerlens` Nanda and Bloom [2022] to carry out the experiments related to the IOI task.

B Author Contributions

AM developed the factual recall results, ran the experiments for Sections 4, 5 and part of D (with the exception of experiments from Appendix I.2 ran by GL), and wrote the majority of the paper. GL also ran the experiments for Sections 3 and D with guidance from AM and contributed to Section D and various other sections. AG provided high-level feedback on experiments and writing for Sections 3 and D, wrote part of Section 2, and provided feedback on most of the paper. NN was the main supervisor for this project, and provided high level feedback on experiments, prioritisation, and writing throughout. NN came up with the original idea of the illusion and the conceptual example.

C Related Work

Discovering and causally intervening on representations with activation patching. Researchers have been exploring increasingly fine-grained ways of reverse-engineering and steering their behavior. In this context, *activation patching* [Vig et al., 2020, Geiger et al., 2020] is a widely used causal intervention, whereby the model is run on an input A, but chosen activations are 'patched in' from input B. Motivated by causal mediation analysis [Pearl, 2001] and causal abstraction Geiger et al.

[2023a], activation patching has been used to localize model components causally involved in various behaviors, such as gender bias [Vig et al.], factual recall [Meng et al., 2022a], multiple choice questions [Lieberum et al., 2023], arithmetic [Stolfo et al., 2023] and natural language reasoning [Geiger et al., 2021, Wang et al., 2023, Geiger et al., 2023b, Wu et al., 2023], code [Heimersheim and Janiak], and (in certain regimes) topic/sentiment/style of free-form natural language [Turner et al., 2023].

Activation patching is an area of active research, and many recent works have extended the method, with patching paths between components [Goldowsky-Dill et al., 2023], automating the finding of sparse subgraphs [Conmy et al., 2023], fast approximations [Nanda, 2023], and automating the verification of hypotheses [Chan et al., 2022]. In particular, pinpointing features to entire model components is not the end of the story. A wide range of interpretability work [Mikolov et al., 2013, Conneau et al., 2018, Tenney et al., 2019, Hewitt and Manning, 2019, Burns et al., 2022, Nanda et al., 2023] suggests the *linear representation hypothesis*: models encode features as linear subspaces of component activations that can be arbitrarily rotated with respect to the standard basis (due to phenomena like superposition, polysemanticity [Arora et al., 2018, Elhage et al., 2022] and lack of privileged bases [Smolensky, 1986, Elhage et al., 2021]).

Motivated by this, recent work such as Geiger et al. [2023b], Wu et al. [2023], Lieberum et al. [2023] has generalized activation patching to operate only on linear subspaces of features rather than patching entire components (heads, layers and neurons). Our work contributes to this research direction by demonstrating both (i) a common illusion to avoid when looking for such subspaces and (ii) a detailed case study of successfully localizing a binary feature to a 1-dimensional subspace.

Factual recall. A well-studied domain for discovering and intervening on learned representations is the localization and editing of factual knowledge in language models [Geva et al., 2023, Meng et al., 2022b, Wallat et al., 2020, Dai et al., 2022, Hernandez et al., 2023]. A work of particular note is Meng et al. [2022a], which localizes and edits factual information with a rank-1 intervention on model weights. However, recent work has shown that rank-1 editing can work even on weights where the fact supposedly is not encoded [Hase et al., 2023], and that editing a single fact often fails to have its expected common-sense effect on logically related downstream facts [Cohen et al., 2023, Zhong et al., 2023].

We contribute to this line of work by showing a formal and empirical connection between activation patching along 1-dimensional subspaces and rank-1 model editing. In particular, rank-1 model edits can work by creating a dormant pathway of an MLP layer, regardless of whether the fact is stored there. This provides a mechanistic explanation for the discrepancy observed in Hase et al. [2023].

Interpretability illusions. Despite the promise of interpretability, it is difficult to be rigorous and easy to mislead yourself. A common theme in the field is identifying ways that techniques and prior work may lead to misleading conclusions about of model behavior [Lipton, 2016]. In computer vision, Adebayo et al. [2018] show that a popular at the time class of pixel attribution methods is not sensitive to whether or not the model used to produce is has actually been trained or not. In Geirhos et al. [2023], the authors show how a circuit can be hardcoded into a learned model so that it fools interpretability methods; this bears some similarity to our illusion, especially its fact editing counterpart. In natural language processing, Bolukbasi et al. [2021] show that interpreting single neurons with maximum activating dataset examples may lead to conflicting results across datasets due to subtle polysemanticity [Elhage et al., 2022].

D Finding and Validating a Faithful Direction Mediating Name Position in the IOI Task

As a counterpoint to the illusion, in this section we demonstrate a success case for subspace activation patching and DAS by revisiting the directions $\mathbf{v}_{\text{grad}}$ and $\mathbf{v}_{\text{resid}}$ defined in Subsection 3.1, and arguing they are faithful to the model's computation to a high degree. Specifically, we subject these directions to the same tests we used for the illusory direction $\mathbf{v}_{\text{MLP}}$ and arrive at significantly different results. Through this and additional validations, we demonstrate that these directions possess the necessary and sufficient properties of a successful activation patch – being both correlated with input variation and causal for the targeted behavior – in an irreducible way.

Ruling out the illusion. Let $W_Q^{\text{name movers}} \in \mathbb{R}^{768\times 192}$ be the stacked query matrices of the name mover heads. In Table 1, we show the fractional logit difference and interchange accuracy when patching $\mathbf{v}_{\text{resid}}$ and $\mathbf{v}_{\text{grad}}$, as well as their components along $\ker W_Q^{\text{name movers}}$ (denoted 'nullspace') and its orthogonal complement (denoted 'causal', a proxy for the causally relevant subspace of the residual stream). We observe that the non-nullspace metrics are broadly similar; in particular, removing the causally disconnected component of $\mathbf{v}_{\text{resid}}$ does not greatly diminish the effect of the patch in terms of the logit difference metrics (as it does for $\mathbf{v}_{\text{MLP}}$). We also find that $\mathbf{v}_{\text{resid}}$ is predominantly in $(\ker W_Q^{\text{name movers}})^\perp$ (and so is $\mathbf{v}_{\text{grad}}$, but this is to be expected).

Importantly, since the residual stream activation where $\mathbf{v}_{\text{resid}}$ and $\mathbf{v}_{\text{grad}}$ are patched is a full bottleneck for the model's computation, it is not possible for these directions to be causal but dormant (in the sense of Section 2): there can be no earlier model component that activates this direction in a way that avoids the patch via a skip connection (unlike for the $\mathbf{v}_{\text{MLP}}$ direction). Indeed, in Figure 7 in Appendix G we show that the $\mathbf{v}_{\text{resid}}$ direction gets written to by the S-Inhibition heads, and in Figure 19 in Appendix J.2, we show they strongly discriminate between the ABB and BAB prompts.

Additional validations. In Appendix G, we further validate these directions' faithfulness to the computation of the IOI circuit from Wang et al. [2020] by finding the model components that write to them and studying how they generalize on the pre-training distribution (OpenWebText); representative samples annotated with attention scores are shown in Figures 10, 8, 9 in Appendix G.

E Additional Details for Section 2

E.1 The Illusion for Higher-Dimensional Subspaces

In the main text, we mostly discuss the illusion for activation patching of 1-dimensional subspaces for ease of exposition. Here, we develop a more complete picture of the mechanics of the illusion for higher-dimensional subspaces.

Let $\mathcal{C}$ be a model component taking values in $\mathbb{R}^d$, and let $U \subset \mathbb{R}^d$ be a linear subspace. Let V be a matrix whose columns form an orthonormal basis for U. If the $\mathcal{C}$ activations for examples A and B are $\mathbf{act}_A, \mathbf{act}_B \in \mathbb{R}^d$ respectively, patching U from A into B gives the patched activation

$$\mathbf{act}_B^{patched} = \mathbf{act}_B + VV^\top(\mathbf{act}_A - \mathbf{act}_B) = (I - VV^\top)\mathbf{act}_B + VV^\top\mathbf{act}_A$$

For intuition, note that $VV^\top$ is the orthogonal projection on U, so this formula says to replace the orthogonal projection of $\mathbf{act}_B$ on U with that of $\mathbf{act}_A$, and keep the rest of $\mathbf{act}_B$ the same.

Generalizing the discussion from Section 2, for the illusion to occur for subspace S, we need S to be sufficiently aligned with a causally disconnected subspace $V_{\text{disconnected}}$ that is correlated with the feature being patched, and a dormant but causal subspace V_{dormant} which, when set to out of distribution values, can achieve the wanted causal effect. For example, a particularly simple way in which this could happen is if we let $V_{\text{disconnected}}, V_{\text{dormant}}$ be 1-dimensional subspaces (like in the setup for the 1-dimensional illusion), and we form S by combining $V_{\text{disconnected}} + V_{\text{dormant}}$ with a number of orthogonal directions that are approximately constant on the data with respect to the feature we are patching. These extra directions effectively don't matter for the patch (because they are cancelled by the $\mathbf{act}_A - \mathbf{act}_B$ term). Given a specific feature, it is likely that such weakly-activating directions will exist in a high-dimensional activation space. Thus, if the 1-dimensional illusion exist, so will higher-dimensional ones.

E.2 Illusory 1-Dimensional Patches are Approximately Equal Parts Causally Disconnected and Dormant

In this subsection, we prove a quantitative corollary of the model of our illusion that suggests that we should expect illusory patching directions to be of the form $v = \frac{1}{\sqrt{2}}(v_{disconnected} + v_{dormant})$ for unit vectors $\|v_{disconnected}\|_2 = \|v_{dormant}\|_2 = 1$. In other words, we expect the best illusory patches to be formed by combining a disconnected and illusory direction with equal coefficients, like depicted in Figure 1:

Lemma E.1. *Suppose we have two distributions of input prompts $\mathcal{D}_A, \mathcal{D}_B$. In the terminology of Section 2, let $v_{disconnected} \perp v_{dormant}$ be unit vectors such that the subspace spanned by*

$v_{disconnected}$ is a causally disconnected subspace, and the subspace spanned by $v_{dormant}$ is strongly dormant, in the sense that the projections of the activations of all examples $\mathcal{D}_{source} \cup \mathcal{D}_{base}$ onto $v_{dormant}$ are equal to some constant c.

Suppose we form $v = v_{disconnected} \cos\alpha + v_{dormant} \sin\alpha$ as a unit-norm linear combination of the two directions. Then the magnitude of the expected change in projection along $v_{dormant}$ when patching from $x_A \sim \mathcal{D}_A$ into $x_B \sim \mathcal{D}_B$ is maximized when $\alpha = \frac{\pi}{4}$, i.e. $\cos\alpha = \sin\alpha = \frac{1}{\sqrt{2}}$.

Proof. Recall that the patched activation from x_A into x_B along v is

$$\text{act}_B^{\text{patched}} = \text{act}_B + (p_A - p_B)v \tag{2}$$

where $p_A = v^\top \text{act}_A, p_B = v^\top \text{act}_B$ are the projections of the two examples' activations on v. The change along $v_{dormant}$ is thus

$$\begin{aligned} &v_{dormant}^\top \left(\text{act}_B^{\text{patched}} - \text{act}_B\right) = (p_A - p_B)\sin\alpha = (v^\top \text{act}_A - v^\top \text{act}_B)\sin\alpha \\ &= v_{disconnected}^\top (\text{act}_A - \text{act}_B)\cos\alpha\sin\alpha \end{aligned}$$

where we used the assumption that $v_{dormant}^\top \text{act}_A = v_{disconnected}^\top \text{act}_B$. Hence, the expected change is

$$\cos\alpha \sin\alpha \, v_{disconnected}^\top \mathbb{E}_{A\sim\mathcal{D}_A, B\sim\mathcal{D}_B}\left[\text{act}_A - \text{act}_B\right].$$

The function $f(\alpha) = \cos\alpha\sin\alpha$ for $\alpha \in [0, \pi/2]$ is maximized for $\alpha = \pi/4$, concluding the proof. □

E.3 Formal definitions of disconnected and dormant subspaces

Let $\mathcal{M} : \mathcal{X} \to \mathcal{O}$ be a machine learning model that on input $x \in \mathcal{X}$ outputs a vector $y \in \mathcal{O}$ of probabilities over a set of output classes. Let $\mathcal{D}$ be a distribution over $\mathcal{X}$, and $\mathcal{C}$ be a component of $\mathcal{M}$, such that for $x \sim \mathcal{D}$ the hidden activation of $\mathcal{C}$ is a vector $\mathbf{c}_x \in \mathbb{R}^d$. For a subspace $U_\mathcal{C} \subset \mathbb{R}^d$, we let $\mathbf{u}_x$ be the orthogonal projection of $\mathbf{c}_x$ onto $U_\mathcal{C}$. Finally, let $\mathcal{M}_{U_\mathcal{C} \leftarrow \mathbf{u}_y}(x)$ be the result of running $\mathcal{M}$ with the input x and setting the orthogonal projection on the subspace $U_\mathcal{C}$ to $\mathbf{u}_y$.

We say U is *causally disconnected* if $\mathcal{M}_{U_\mathcal{C} \leftarrow \mathbf{u}'}(x) = \mathcal{M}(x)$ for all $\mathbf{u}' \in U$. In other words, setting the value of a causally disconnected subspace to any vector has no effect on model outputs. We say U is *dormant* if $\mathcal{M}_{U_\mathcal{C} \leftarrow \mathbf{u}_y}(x) = \mathcal{M}(x)$ with high probability over $x, y \sim \mathcal{D}$, but is *not* causally disconnected. In other words, a dormant subspace is approximately causally disconnected on the data distribution, but can have substantial causal effect if set to out of distribution values.

E.4 Concrete mathematical example of the illusion

For a distilled example of the illusion, consider a network $\mathcal{A}$ that takes in a real valued input $x \in \mathbb{R}$, computes a three dimensional hidden representation $\mathbf{h} = W_1^T x$, and then a real valued output $y = W_2^T \mathbf{h}$. Define the weights to be $W_1 = [1 \quad 0 \quad 1]$ and $W_2 = [0 \quad -2 \quad 1]$ and observe that the network computes the identity function (see Figure 18 in Appendix J.1 for an illustration). While the 3rd hidden neuron clearly mediates this effect, surprisingly, patching the direction along the sum of the first two neurons does as well, despite the fact that the 1st neuron is causally disconnected, and the 2nd is dormant.

Specifically, consider Figure 18. It should be obvious that the hidden unit H_3 fully mediates the information flow from input to output, and that H_2 is dormant while H_3 is disconnected. However, it may be surprising that the linear subspace of H_2 and H_3 defined by the unit vector $[\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}]$ also fully mediates the information flow, despite it consisting of dormant and disconnected directions. Activation patching on this subspace leverages the information stored in the disconnected subspace in order to activate the dormant subspace by fixing it to an out of distribution value. In this way, activation patching on a subspace can activate a 'dormant parallel circuit'.

F Additional Details for Section 3

F.1 Dataset, Model and Evaluation Details for the IOI Task

We use GPT2-Small for the IOI task, with a dataset that spans 216 single-token names, 144 single-token objects and 75 single-token places, which are split $1:1$ across a training and test set. Every example in the data distribution includes (i) an initial clause introducing the indirect object (**IO**, here 'Mary') and the subject (**S**, here 'John'), and (ii) a main clause that refers to the subject a second time. Beyond that, the dataset varies in the two names, the initial clause content, and the main clause content. Specifically, use three templates as shown below:

Then, [ ] and [ ] had a long and really crazy argument. Afterwards, [ ] said to
Then, [ ] and [ ] had lots of fun at the [place]. Afterwards, [ ] gave a [object] to
Then, [ ] and [ ] were working at the [place]. [ ] decided to give a [object] to

and we use the first two in training and the last in the test set. Thus, the test set relies on unseen templates, names, objects and places. We used fewer templates than the IOI paper Wang et al. [2020] in order to simplify tokenization (so that the token positions of our names always align), but our results also hold with shifted templates like in the IOI paper.

On the test partition of this dataset, GPT2-Small achieves an accuracy of $\approx 91\%$. The average difference of logits between the correct and incorrect name is ≈ 3.3, and the logit of the correct name is greater than that of the incorrect name in $\approx 99\%$ of examples. Note that, while the logit difference is closely related to the model's correctness, it being > 0 does not imply that the model makes the correct prediction, because there could be a third token with a greater logit than both names.

F.2 Details for Computing the Gradient Direction $\mathbf{v_{grad}}$

For a given example from the test distribution and a given name mover head, we compute the gradient of the difference of attention scores from the final token position to the 3rd and 5th token in the sentence (where the two name tokens always are in our data). We then average these gradients over a large sample of the full test distribution and over the three name mover heads, and finally normalize the resulting vector to have unit ℓ_2 norm.

We note that there is a 'closed form' way to compute approximately the same quantity that requires no optimization. Namely, for a single example we can collect the keys k_S, k_{IO} to the name mover heads at the first two names in the sentence (the **S** and **IO** name). Then, for a single name mover head with query matrix W_Q, a maximally causal direction v in the residual stream at the last token position after layer 8 will be one such that $W_Q v$ is in the direction of $k_S - k_{IO}$, because the attention score is simply the dot product between the keys and queries. We can use this to 'backpropagate' to v by multiplying with the pseudoinverse W_Q^+. This is slightly complicated by the fact that we have been ignoring layer normalization, which can be approximately accounted for by estimating the scaling parameters (which tend to concentrate well) from the IOI data distribution. We note that this approach leads to broadly similar results.

F.3 Training Details for DAS

To train DAS, we always sample examples from the training IOI distribution as described in Appendix F. We sample equal amounts of pairs of base (which will be patched into) and source (where we take the activation to patch in from) prompts where the two names are the same between the prompts, and pairs of prompts where all four names are distinct. We optimize DAS to maximize the logit difference between the name that should be predicted if the position information from the source example is correct and the other name.

For training, we use a learned rotation matrix as in the original DAS paper [Geiger et al., 2023b], parametrized with `torch.nn.utils.parametrizations.orthogonal`. We use the Adam optimizer and minibatch training over a training set of several hundred patching pairs. We note that results remain essentially the same when using a higher number of training examples.

F.4 Discussion of the Magnitude of the Illusion

While the contribution of the $\mathbf{v}_{\text{MLP}}$ patch to logit difference may appear relatively small, we note that this is the result of patching a direction in a single model component at a single token position. Typical circuits found in real models (including the IOI circuit from Wang et al. [2023]) are often composed of multiple model components, each of which contribute. In particular, the position signal itself is written to by 4 heads, and chiefly read by 3 other heads. As computation tends to be distributed, when patching an individual component accuracy may be a misleading metric (eg patching 1 out of 3 heads is likely insufficient to change the output), and a fractional logit diff indicates a significant contribution. By contrast, patching in the residual stream is a more potent intervention, because it can affect *all* information accumulated in the model that is communicated to downstream components.

F.5 Analyzing the nullspace and rowspace components of $\mathbf{v}_{\text{MLP}}$

We note that $\mathbf{v}_{MLP}$ *decomposes as a causally disconnected and dormant component*: we observe that $\mathbf{v}_{\text{MLP}}$ is roughly equal parts in $\ker W_{out}$ and $(\ker W_{out})^{\perp}$, as predicted by the model of the illusion (see Appendix E.2). We also look at how sensitive the (normalized) components of $\mathbf{v}_{\text{MLP}}$ in $\ker W_{out}$ and $(\ker W_{out})^{\perp}$ are to the information of whether the first/second name is repeated in Figure 20 in Appendix J.2, and find that, as expected, the causally disconnected component is significantly more sensitive, further confirming the mechanics of the illusion are at play.

F.6 Random ablation of MLP weights

How certain are we that MLP8 doesn't actually matter for the IOI task? While we find the IOI paper analysis convincing, to make our results more robust to the possibility that it does matter, we also design a further experiment.

Given our conceptual picture of the illusion, the computation performed by the MLP layer where we find the illusory subspace does not matter as long as it propagates the correlational information about the position feature from the residual stream to the hidden activations, and as long as the output matrix W_{out} is full rank (also, see the discussion in 5). Thus, we expect that if we replace the MLP weights by randomly chosen ones with the same statistics, we should still be able to exhibit the illusion.

Specifically, we randomly sampled MLP weights and biases such that the norm of the output activations matches those of MLP8. As random MLPs might lead to nonsensical text generation, we don't replace the layer with the random weights, but rather train a subspace using DAS on the MLP activations, and add the difference between the patched and unpatched output of the random MLP to the real output of MLP8. This setup finds a subspace that reduces logit difference even more than the $\mathbf{v}_{\text{MLP}}$ direction.

This suggests that the existence of the $\mathbf{v}_{\text{MLP}}$ subspace is less about *what* information MLP8 contains, and more about *where* MLP8 is in the network.

G Additional Details for Section D

Which model components write to the $\mathbf{v}_{\text{resid}}$ direction? To test how every attention head and MLP contributes to the value of projections on $\mathbf{v}_{\text{MLP}}$, we sampled activations from head and MLP outputs at the last token position of IOI prompts, and calculated their dot product with $\mathbf{v}_{resid}$ (Figure 7). We found that the dot products of most heads and MLPs was low, and that the S-inhibition heads were the only heads whose dot product differed between different patterns ABB and BAB. This shows that only the S-inhibition heads write to the $\mathbf{v}_{\text{resid}}$ direction (as one would hope). Importantly, this test separates $\mathbf{v}_{resid}$ from the interpretability illusion $\mathbf{v}_{MLP}$. While patching $\mathbf{v}_{\text{MLP8}}$ also writes to $\mathbf{v}_{\text{resid8}}$ (i.e. $\mathbf{v}_{\text{MLP8}} W_{out} \approx \mathbf{v}_{\text{resid8}}$), the MLP layer does not write this subspace on the IOI task (see Figure 3). This further supports the observation that the $\mathbf{v}_{\text{MLP}}$ patch activates a dormant pathway in the model.

Generalization beyond the IOI distribution. We also investigate how the subspace generalizes. We sample prompts from OpenWebText-10k and look at those with particularly high and low activations in $\mathbf{v}_{\text{sinhib}}$. Representative examples are shown in Figure 8 together with the name movers attention at the position of interest, how the probability changes after subspace ablation, and how the name movers attention changes.

Stability of found solution. Finally, we note that solutions found by DAS in the residual stream are stable, including when trained on a subset of S-inhibition heads (see Figure 5).

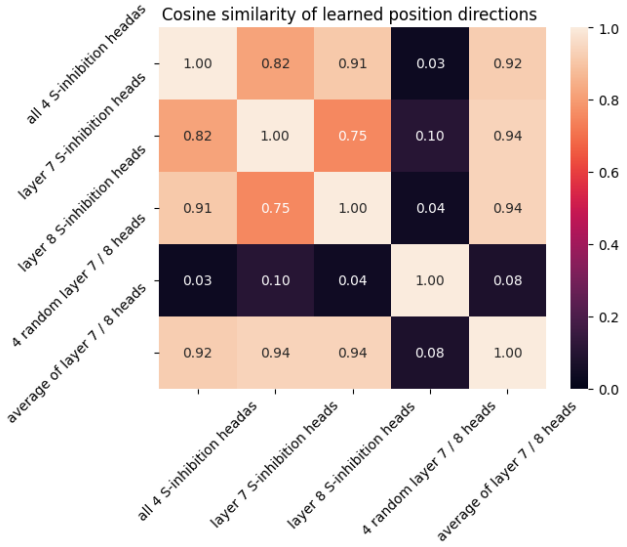


Figure 5: Cosine Similarity between learned position subspaces in the S-inhibition heads is high even when using only a subset of S-inhibition heads for training

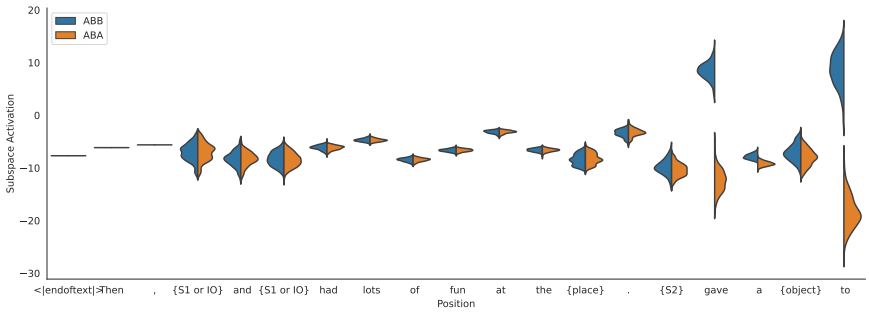


Figure 6: The IOI position subspace activates at words that predict a repeated name. S-inhibition subspace activations for different IOI prompts per position

H Additional details for Section 4

H.1 Training Details for Fact Patching (Section 4.1)

We patch from the last token of s' in B to the last token of s in A (prior work has shown that the fact is retrieved on s [Geva et al., 2023]), and we again use DAS Geiger et al. [2023b] to optimize for a direction that maximizes the logit difference between o' and o. We use a selection of examples from the COUNTERFACT dataset [Meng et al., 2022a] and use GPT-2 XL (1.5B parameters) for experiments.

We use the first 1000 examples from the COUNTERFACT dataset [Meng et al., 2022a]. We filter the facts which GPT2-XL correctly recalls. Out of the remaining facts, for each relation we form all pairs of distinct facts, and we sample 5 such pairs from each relation with at least 5 facts. This results in a collection of 40 facts spanning 8 different relations.

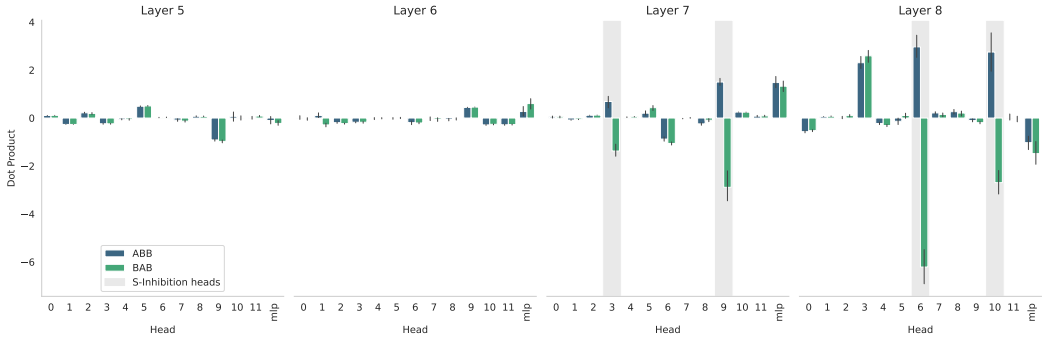


Figure 7: S-Inhibition heads but not MLP8 write to the position subspace in the residual stream that is causally connected to the name movers on the IOI task

H.2 Additional fact patching experiments

In figure 11, we show the distribution of the fractional logit difference metric (see Subsection 3.1 for a definition) when patching between facts as described in Subsection 4.1. Like in the related Figure 4, we observe that, while patching along the directions found by DAS achieves strongly negative values (indicating that the facts are very often successfully changed by the patch), the interventions that replace the entire MLP layer or only the causally relevant component of the DAS directions have no such effect.

Next, we observe that the nullspace component of the patching direction is the one similar to the variation in the inputs (difference of last-token activations at the two subjects). Specifically, in Figure 12, we plot the (absolute value of the) cosine similarity between the difference in activations for the two last subject tokens, and the nullspace component of the DAS direction. We note that this similarity is consistently significantly high (note that it can be at most 1, which would indicate perfect alignment).

Finally, we observe that the nullspace component of the patching direction is a non-trivial part of the direction in Figure 13, where we plot the distribution of the ℓ_2 norm of this component.

H.3 ROME implementation details

ROME takes as input a vector $\mathbf{k} \in \mathbb{R}^{d_{\text{MLP}}}$ representing the subject (e.g. an average of last-token representations of the subject) and a vector $\mathbf{v} \in \mathbb{R}^{d_{\text{resid}}}$ which, when output by the MLP layer, will cause the model to predict a new object for the factual prompt, but at the same time won't change other facts about the subject. ROME modifies the MLP weight by setting $W'_{out} = W_{out} + \mathbf{a}\mathbf{b}^\top$, where $\mathbf{a} \in \mathbb{R}^{d_{\text{resid}}}, \mathbf{b} \in \mathbb{R}^{d_{\text{MLP}}}$ are chosen so that $W'_{out}\mathbf{k} = \mathbf{v}$, and the MLP's output is otherwise minimally changed. Without loss of generality, the first condition implies that $\mathbf{a} = \mathbf{v} - W_{out}\mathbf{k}$ and $\mathbf{b}^\top\mathbf{k} = 1$; the second condition is then modeled by minimizing the variance of $\mathbf{b}^\top\mathbf{x}$ when $\mathbf{x} \sim \mathcal{N}(0, \Sigma)$ for an empirical estimate $\Sigma \in \mathbb{R}^{d_{\text{MLP}} \times d_{\text{MLP}}}$ of the covariance of MLP activations (see Lemma H.1 in Appendix H for details and a proof).

H.4 ROME as an Optimization Problem

We now review the ROME method from Meng et al. [2022a] and show how it can be characterized as the solution of a simple optimization problem. Following the terminology of 4.2, let us have an MLP layer with an output projection W_{out}, a key vector $\mathbf{k} \in \mathbb{R}^{d_{\text{MLP}}}$ and a value vector $\mathbf{v} \in \mathbb{R}^{d_{\text{resid}}}$.

In Meng et al. [2022a], equation 2, the formula for the rank-1 update to W_{out} is given by

$$W'_{out} = W_{out} + (\mathbf{v} - W_{out}\mathbf{k})\frac{\mathbf{k}^\top\Sigma^{-1}}{\mathbf{k}^\top\Sigma^{-1}\mathbf{k}} \tag{3}$$

where Σ is an empirical estimate of the uncentered covariance of the pre-W_{out} activations. We derive the following equivalent characterization of this solution (which may be of independent interest):

Lemma H.1. *Given a matrix $W_{out} \in \mathbb{R}^{d_{resid} \times d_{MLP}}$, a key vector $\mathbf{k} \in \mathbb{R}^{d_{MLP}}$ and a value vector $\mathbf{v} \in \mathbb{R}^{d_{resid}}$, let $\Sigma \succ 0, \Sigma \in \mathbb{R}^{d_{MLP} \times d_{MLP}}$ be a positive definite matrix (specifically, the uncentered empirical covariance), and let $\mathbf{x} \sim \mathcal{N}(0, \Sigma)$ be a normally distributed random vector with zero mean and covariance Σ. Then, the ROME weight update is $W'_{out} = W_{out} + \mathbf{a}\mathbf{b}^\top$ where $\mathbf{a} \in \mathbb{R}^{d_{resid}}, \mathbf{b} \in \mathbb{R}^{d_{MLP}}$ solve the optimization problem*

$$\min_{\mathbf{a},\mathbf{b}} \operatorname{trace}(\operatorname{Cov}_{\mathbf{x}} [W'_{out}\mathbf{x} - W_{out}\mathbf{x}]) \quad \textit{subject to} \quad W'_{out}\mathbf{k} = \mathbf{v}.$$

In other words, the ROME update is the update that causes W_{out} to output $\mathbf{v}$ on input $\mathbf{k}$, and minimizes the total variance of the extra contribution of the update in the output of the MLP layer under the assumption that the pre-W_{out} activations are normally distributed with covariance Σ[2].

Proof. Using $\mathbb{E}_{\mathbf{x}}[\mathbf{x}\mathbf{x}^\top] = \Sigma$ and the cyclic property of the trace, we see that

$$\operatorname{trace}(\operatorname{Cov}_{\mathbf{x}} [W'_{out}\mathbf{x} - W_{out}\mathbf{x}]) = \|\mathbf{a}\|_2^2 \mathbf{b}^\top \Sigma \mathbf{b}$$

We must have $\mathbf{a}\mathbf{b}^\top\mathbf{k} = \mathbf{v} - W\mathbf{k}$, so without loss of generality we can rescale $\mathbf{a}, \mathbf{b}$ so that $\mathbf{a} = \mathbf{v} - W\mathbf{k}$. Then, we want to solve the problem

$$\min_{\mathbf{b}} \mathbf{b}^\top \Sigma \mathbf{b} \quad \text{subject to} \quad \mathbf{b}^\top \mathbf{k} = 1$$

which we can solve using Lagrange multipliers. The Lagrangian is

$$\mathcal{L}(\mathbf{b}, \lambda) = \frac{1}{2}\mathbf{b}^\top \Sigma \mathbf{b} - \lambda \mathbf{b}^\top \mathbf{k}$$

and the derivative w.r.t. $\mathbf{b}$ is $\Sigma\mathbf{b} - \lambda\mathbf{k} = 0$, which tells us that $\mathbf{b}$ is in the direction of $\Sigma^{-1}\mathbf{k}$. Then the constraint $\mathbf{b}^\top\mathbf{k} = 1$ forces the constant of proportionality, and we arrive at $\mathbf{b} = \frac{\mathbf{k}^\top \Sigma^{-1}}{\mathbf{k}^\top \Sigma^{-1} \mathbf{k}}$ □

H.5 Connection between 1-dimensional activation patching and model editing

Lemma H.2. *Given prompts A and B, two token positions t_A, t_B, and an MLP layer with output projection weight $W_{out} \in \mathbb{R}^{d_{resid} \times d_{MLP}}$, let $u_A, u_B \in \mathbb{R}^{d_{MLP}}$ be the respective (post-nonlinearity) activations at these token positions in this layer. If v is a direction in the activation space of the MLP layer, then there exists a ROME edit $W'_{out} = W_{out} + ab^\top$ such that the activation patch from u_B into u_A along v and the edit result in equal outputs of the MLP layer at token t_A when run on prompt A. Moreover, the ROME edit is given by*

$$a = \left((u_B - u_A)^\top v\right) W_{out} v \quad \textit{and any } b \textit{ that satisfies} \quad b^\top u_A = 1.$$

Choosing $b = \frac{\Sigma^{-1}u_A}{u_A^T \Sigma^{-1} u_A}$ minimizes the change to the model (in the sense of Meng et al. [2022a]) over all such rank-1 edits.

Proof. The activation after patching from B into A along v is $u'_A = u_A + ((u_B - u_A)^\top v)v$, which means that the change in the output of the MLP layer at this token will be

$$W_{out}u'_A - W_{out}u_A = ((u_B - u_A)^\top v) W_{out} v$$

The change introduced by a fact edit at this token is

$$W'_{out}u_A - W_{out}u_A = ab^\top u_A = \left(b^\top u_A\right)\left((u_B - u_A)^\top v\right) W_{out} v$$

and the two are equal because $b^\top u_A = 1$.

To find the b that minimizes the change to the model, we minimize the variance of $b^\top x$ when $x \sim \mathcal{N}(0, \Sigma)$ subject to $b^\top u_A = 1$. The variance is equal to $b^\top \Sigma b$, so we have a constrained (convex) minimization problem

$$\min \frac{1}{2} b^\top \Sigma b \quad \text{subject to} \quad b^\top u_A = 1$$

[2]Note that in practice W_{out} may be singular or poorly conditioned, because the layer normalization encourages features to sum to zero, which could to some extent also persist after a non-linearity. If this is the case, all our results apply with Σ^+ instead of Σ^{-1}

The rest of the proof is the same as in Lemma H.1. Namely, we can solve this optimization problem using Lagrange multiplies. The Lagrangian is

$$\mathcal{L}(b, \lambda) = \frac{1}{2} b^\top \Sigma b - \lambda b^\top u_A$$

and the derivative w.r.t. b is $\Sigma b - \lambda u_A = 0$, which tells us that b is in the direction of $\Sigma^{-1} u_A$. Then the constraint $b^\top u_A = 1$ forces the constant of proportionality. □

H.6 Additional experiments comparing fact patching and rank-1 editing

In Figure 14, we plot the distributions of the logit difference between the correct object for a fact and the object we are trying to substitute when patching the 1-dimensional subspaces found by DAS, and performing the equivalent rank-1 weight edit according to Lemma H.2. We observe that the two metrics quite closely track each other, indicating that the additional effects of using a weight edit (as opposed to only intervening at a single token) are negligible.

Similarly, in Figure 15, we show the success rate of the the two methods in terms of making the model output the object of the fact we are patching from. Again, we observe that they quite closely track each other.

I Why Do We Expect the Illusion to be Prevalent in Practice?

I.1 MLP weights are full-rank matrices

In figure 16, we plot the 100 smallest singular values of the MLP weights in GPT2-Small for all 12 layers. We observe that they the vast majority are bounded well away from 0. This confirms that both MLP weights are full-rank transformations.

I.2 Features in the residual stream propagate to hidden MLP activations

Intuition. Suppose we have two classes of examples that are linearly separable in the residual stream. The transformation from the residual stream to the hidden MLP activations is a linear map followed by a nonlinearity, specifically $x \mapsto \text{gelu}(W_{in}x)$. As we observed in I.1, the W_{in} matrix is full-rank, meaning that all the information linearly present in x will also be so in $W_{in}x$. Even better, since W_{in} maps x from a d_{resid}-dimensional space to a $d_{\text{MLP}} = 4d_{\text{resid}}$-dimensional space, this should intuitively make it much easier to linearly separate the points, because in a higher-dimensional space there are many more linear separators. On the other hand, the non-linearity has an opposite effect: by compressing the space of activations, it makes it harder for points to be separable. So it is a priori unclear which intuition is decisive.

Empirical validation. However, it turns out that empirically this is not such a problem. To test this, we run the model GPT2-Small on random samples from its data distribution (we used OpenWebText-10k), and extract 2000 activations of an MLP-layer after the non-linearity. We train a linear regression with ℓ_2-regularization to recover the dot product of the residual stream immediately before the MLP-layer of interest and a randomly chosen direction. We repeat this experiment with different random vectors and for each layer. We observe that all regressions are better than chance and explain a significant amount of variance on the held-out test set ($R^2 = 0.71 \pm 0.17, \text{MSE} = 0.31 \pm 0.18, p < 0.005$). Results are shown in Figure 17 (right) (every marker corresponds to one regression model using a different random direction).

The position information in the IOI task is really a binary feature, so we are also interested in whether *binary* information in general is linearly recoverable from the MLP activations. To test this, we sample activations from the model run on randomly-sampled prompts. This time however, we add or subtract a multiple of a random direction v to the residual stream activation u, and calculate the MLP activations using this new residual stream vector u':

$$u' = u + y \times z \times \|u\|_2 \times v$$

where $y \in \{-1, 1\}$ is uniformly random, z is a scaling factor we manipulate, and v is a randomly chosen direction of unit norm. For each classifier, we randomly sample a direction v that we either

add or subtract (using y) from the residual stream. The classifier is trained to predict y. We rescale v to match the average norm of a residual vector and then scale it with a small scalar z.

Then, a logistic classifier is trained on 1600 samples. Again, we repeat this experiment for different v and z, and for each layer. We observe that the classifier works quite well across layers even with very small values of z (still, accuracy drops for $z = 0.0001$). Results are shown in Figure 17 (right), and Table 2.

Table 2: Mean Accuracy for Different Values of z

z	Mean Accuracy
0.0001	0.69
0.001	0.83
0.01	0.87
0.1	0.996

J Supplementary Figures

J.1 Additional Figures for Section 2

J.2 Additional Figures for Section 3

Figure 8: The IOI position subspace generalizes to arbitrary OpenWebText prompts

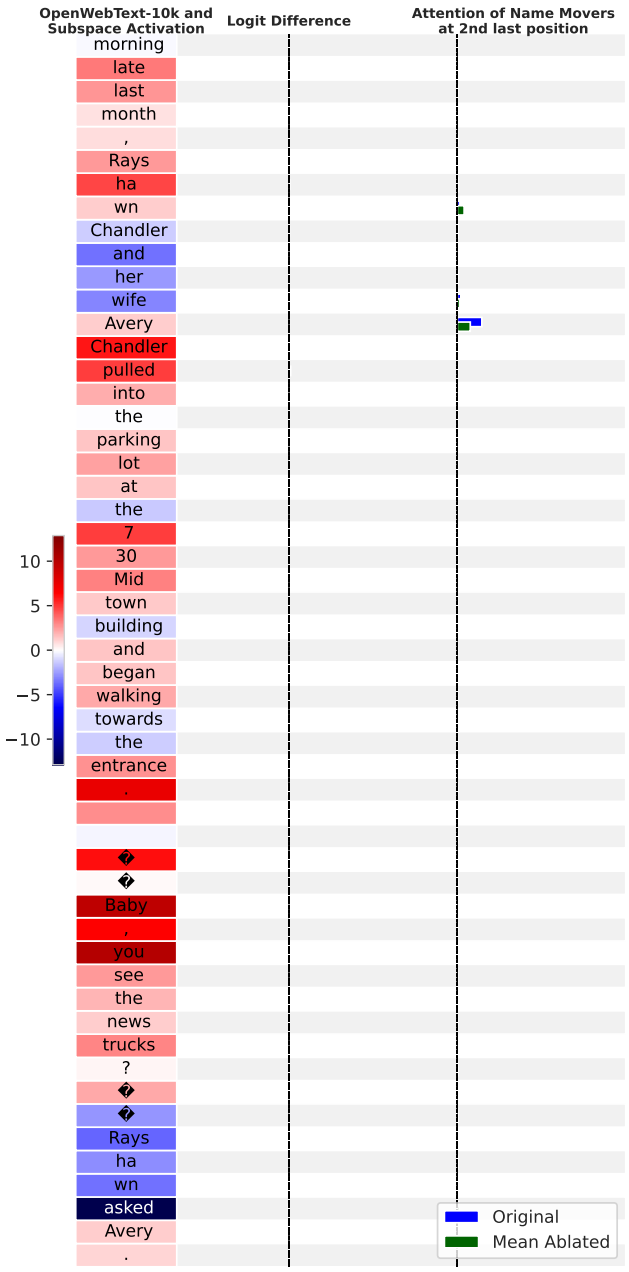

OpenWebText-10k and Subspace Activation
Logit Difference
Attention of Name Movers at 2nd last position
morning
late
last
month
,
Rays
ha
wn
Chandler
and
her
wife
Avery
Chandler
pulled
into
the
parking
lot
at
the
7
30
Mid
town
building
and
began
walking
towards
the
entrance
.
10
5
0
−5
−10
Baby
,
you
see
the
news
trucks
?
Rays
ha
wn
asked
Avery
.
Original
Mean Ablated

Figure 9

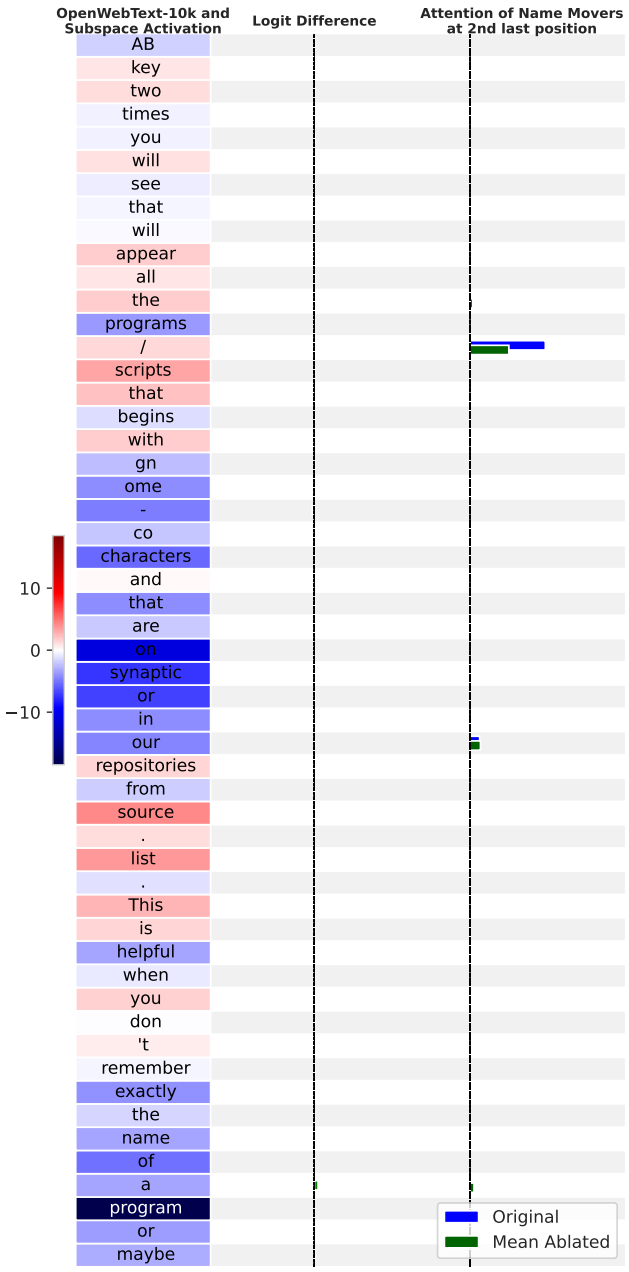

OpenWebText-10k and Subspace Activation
Logit Difference
Attention of Name Movers at 2nd last position
AB
key
two
times
you
will
see
that
will
appear
all
the
programs
/
scripts
that
begins
with
gn
ome
-
co
characters
and
that
are
on
synaptic
or
in
our
repositories
from
source
.
list
.
This
is
helpful
when
you
don
't
remember
exactly
the
name
of
a
program
or
maybe
10
0
−10
Original
Mean Ablated

Figure 10

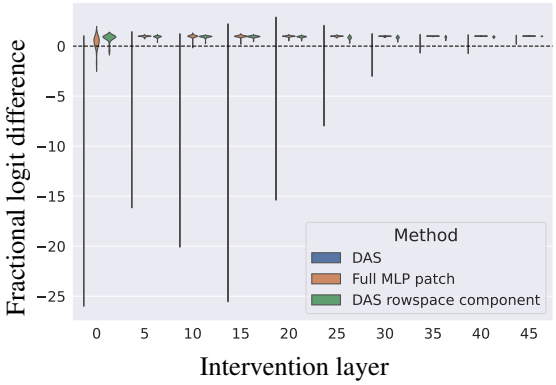


Figure 11: Fractional logit difference distributions under three interventions: patching along the direction found by DAS (blue), patching the component of the DAS direction in the rowspace of W_{out} (green), and patching the entire hidden MLP activation (orange).

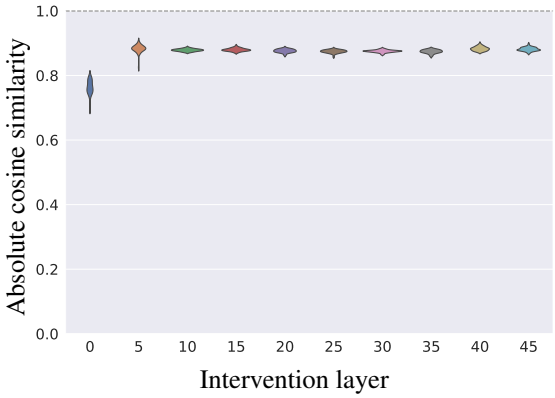


Figure 12: Distribution of the absolute value of the cosine similarity between the nullspace component of the DAS fact patching directions and the difference in activations of the last tokens of the two subjects.

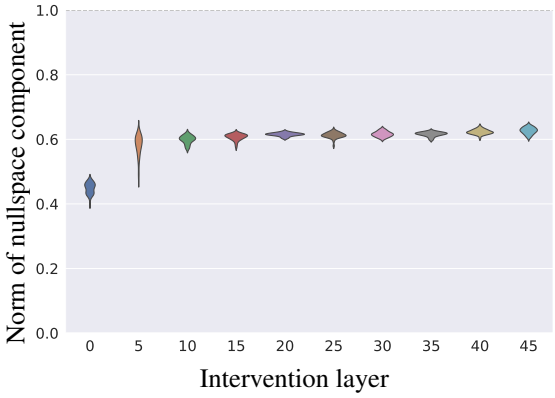


Figure 13: Distribution of the norm of the nullspace component of the DAS direction across intervention layers.

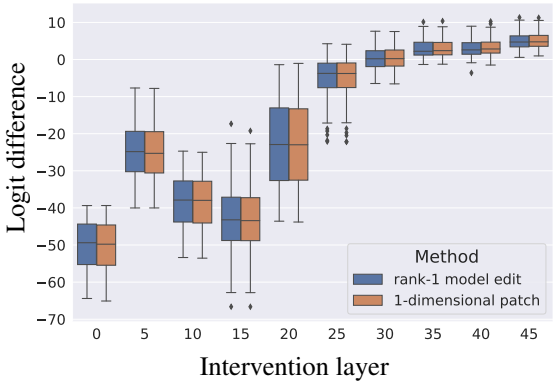


Figure 14: Comparison of logit difference between 1-dimensional fact patches and their derived rank-1 model edits

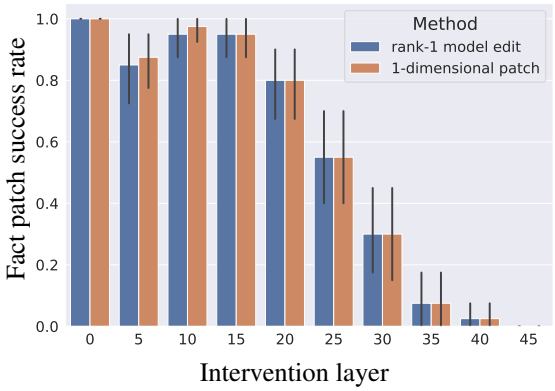


Figure 15: Comparison of fact editing success rate between 1-dimensional fact patches and their derived rank-1 model edits

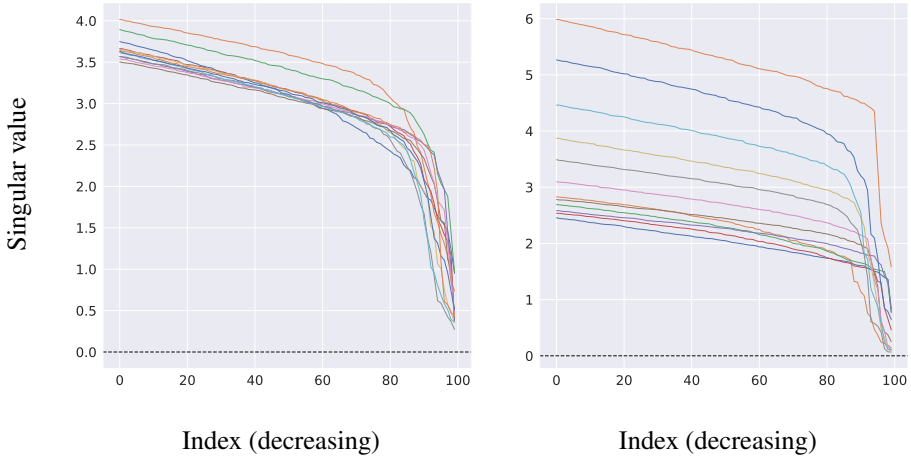


Figure 16: Smallest 100 singular values of the W_{in} (left) and W_{out} (right) MLP weights by layer in in GPT2-Small

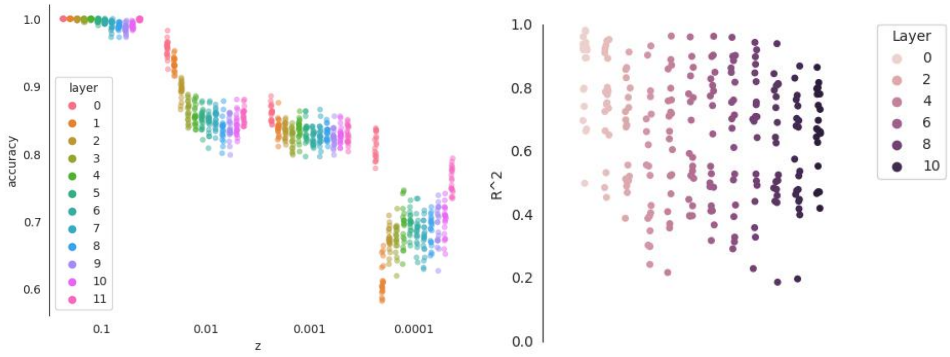
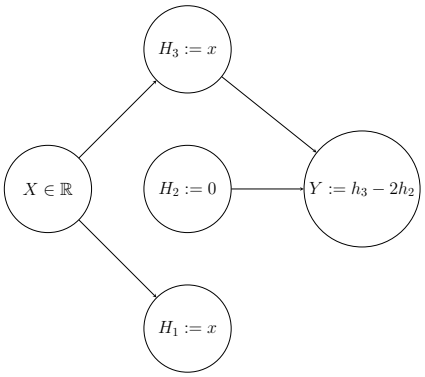
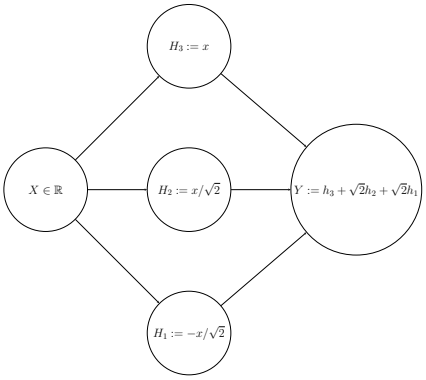


Figure 17: Recovering residual stream features linearly from hidden MLP activations: classification (left) and regression (right).



(a) The network $\mathcal{A}$ that computes the identity function. The hidden unit H_3 stores the value of the input and passes this to the output, while the unit H_2 is dormant and H_1 is disconnected. However, the linear subspace of H_1 and H_3 defined by the unit vector $[\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}]$ fully mediates the information flow from input to output just like the unit H_3.



(b) The network $\mathcal{A}$ except the weights have been transformed such that the hidden units H_2 and H_3 are viewed under the coordinate vectors $[\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}]$ and $[\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}]$. When we generalize activation patching to arbitary subspaces, we are forced to consider this transformed network to be analytically identical to $\mathcal{A}$.

Figure 18: Diagrams of small linear networks illustrating a concrete instantiation of the interpretability illusion, alongside a subspace faithful to a model's computation.

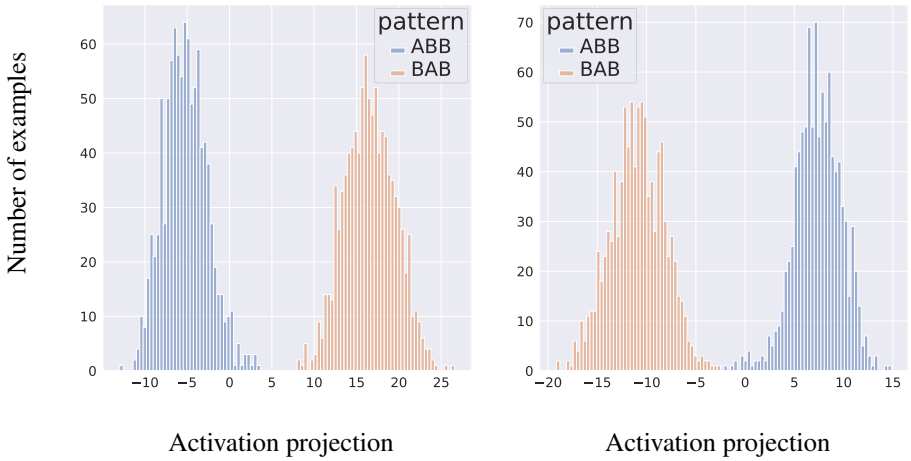


Figure 19: Projections of dataset examples' activations in the residual stream after layer 8 onto the $\mathbf{v}_{\text{resid}}$ direction found by DAS and the $\mathbf{v}_{\text{grad}}$ direction which is the gradient for difference in attention of the name mover heads to the two names.

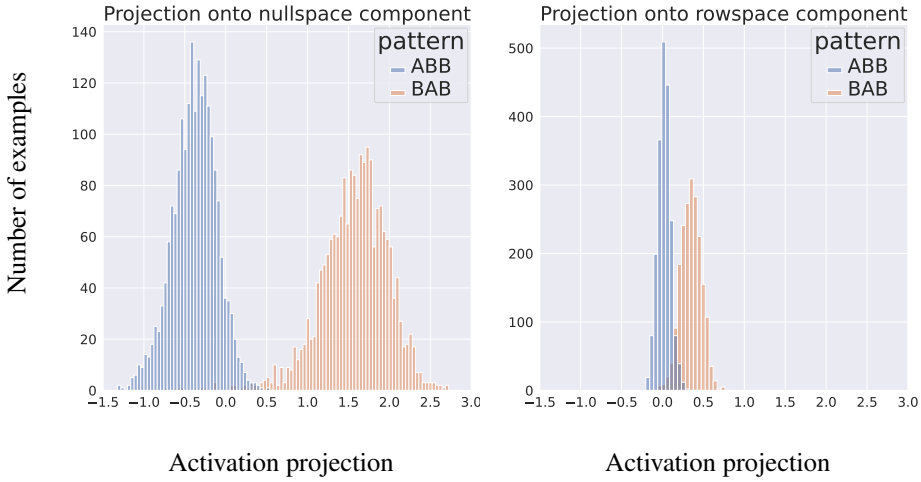


Figure 20: Projections of dataset examples onto the two components of the illusory patching direction found in MLP8: the nullspace (irrelevant) component (left), and the rowspace (dormant) component (right).