Investigating Sensitive Directions in GPT-2: An Improved Baseline and Comparative Analysis of SAEs

Daniel J. Lee[*]
Department of Biomedical Informatics
Harvard Medical School
Boston, MA

Stefan Heimersheim
Apollo Research
London, UK

Abstract

Sensitive directions experiments attempt to understand the computational features of Language Models (LMs) by measuring how much the next token prediction probabilities change by perturbing activations along specific directions. We extend the sensitive directions work by introducing an improved baseline for perturbation directions. We demonstrate that KL divergence for Sparse Autoencoder (SAE) reconstruction errors are no longer pathologically high compared to the improved baseline. We also show that feature directions uncovered by SAEs have varying impacts on model outputs depending on the SAE's sparsity, with lower $L0$ SAE feature directions exerting a greater influence. Additionally, we find that end-to-end SAE features do not exhibit stronger effects on model outputs compared to traditional SAEs.

1 Introduction

One of the primary goals of mechanistic interpretability is to uncover the variables that neural networks use in their computation. This task is complicated by polysemanticity, a phenomenon where a single neuron activates in response to multiple seemingly unrelated features [1, 2]. Recent studies [3, 4] have employed an unsupervised dictionary learning algorithm called Sparse Autoencoders (SAEs) to disentangle LM activations into sparse, linear combinations of feature directions. While SAEs show significant promise [5], there is limited dataset-independent evidence that the features found by SAEs are indeed true abstractions used by the LMs.

Several works have sought to understand these abstractions by observing how much the next token prediction probabilities change when activations are perturbed along certain directions, a technique hereinafter referred to as sensitive direction analysis. Heimersheim and Mendel [6] demonstrated, for example, that perturbing from one real activation towards another real activation changes the model output earlier (shorter perturbation lengths) than perturbing into random directions. This finding supports the hypothesis that perturbations along true feature directions have a greater impact on model outputs compared to other directions, motivated by toy models of computation in superposition [7].

Sensitive direction analyses have been also used to evaluate Sparse Autoencoders (SAEs). Perturbations along the SAE feature directions appear to alter the model output more than random directions, suggesting that SAEs successfully uncover important "levers" used by the model [8]. However, SAE-reconstructed activation vectors also alter the model output much more than random perturbations of the same L2 distance from the base activation, an observation that puzzled the interpretability community [9]. This phenomenon was characterized as a pathological behavior of SAE reconstruction errors.

[*]daniel_lee@g.harvard.edu

Workshop on Attributing Model Behavior at Scale at NeurIPS 2024.

Our contribution In this paper, we expand on the sensitive directions work. We show that:

- Heimersheim and Mendel [6]'s sensitive direction baselines were flawed in that the perturbation direction involved subtracting the original activation. We propose an improved baseline direction (called *cov-random mixture*) which does not use the original activation.
- Gurnee [9]'s KL-divergence for SAE reconstruction errors no longer seems pathologically high when we use this improved baseline.
- Perturbations into SAE feature directions reveal that (1) SAE directions have smaller or greater impact on the model output than our baseline, depending on the SAE type and $L0$, and (2) lower $L0$ SAE feature directions have a greater impact on the model output than higher $L0$ SAE feature directions.
- Feature directions from end-to-end SAEs do not exhibit a greater influence on the model output than those from traditional SAEs.

2 Experimental Methods

The experiments described in this report focus on perturbing an activation within the residual stream of GPT2-small. Specifically, we perform perturbation as follows:

$$x \leftarrow x^{\text{base}} + \alpha d$$

where x^{base} represents the original activation, α is the perturbation length, and d is a unit direction vector. To assess the impact on the model's output, we use the KL divergence of the next token prediction probabilities (more specifically, KL(original prediction | prediction with substitution)). Unless if otherwise stated, the perturbations are applied in Layer 6 resid_pre. Layer 6 was chosen because Braun et al. [10]'s main results focus on end-to-end SAEs on Layer 6 activations.

Data The experiments are performed on approximately 2 million tokens (16,000 sequences, each with a length of 128) from Openwebtext. We also run a subset of the experiments (Figure 1) for 15 million tokens from Openwebtext, and confirm that the result stays the same (see Appendix B). We perturb activations for all token positions.

Extrapolation When we extrapolate the perturbation vector, we extend the vector from length 0 to 101 (the mean L2 distance between two actual activations in Layer 6 resid_pre is 81.59). Our results mainly focus on the resulting curves of KL vs perturbation length or L2 distance at Layer 11 vs perturbation length. We use the mean of KL or mean of L2 across the 2 million tokens as our main measure. We use the mean under the assumption that directions with greater functional importance will, on average, induce a more significant change in the model's output.

3 Developing a Better Baseline

Lindsey [8] and Gurnee [9] use random isotropic perturbation as their baseline. Both papers point out that this might be problematic because actual activations are not isotropic, and some sensitivity differences may be explained by that effect. Previous work by Heimersheim and Mendel [6] attempts to address this issue by adjusting the mean and covariance matrix of the randomly generated activations to match real activations. However, the paper's perturbation directions use the direction from the original activation toward another random activation ($x^{\text{target}} - x^{\text{base}}$), which includes the negative of the original activation ($-x^{\text{base}}$) as a component. This makes it an unfair comparison to directions that do not include the original activation. Therefore, we propose two new baselines (*cov-random mixture* and *real mixture*) where the directions do not include the original activation.

Following is the list of perturbation directions discussed in this section:

- **Isotropic random**: Perturb into a random direction (no subtraction)
- **Cov-random difference**: Perturb along $d = x^{\text{cov-random}} - x^{\text{base}}$, i.e. from base towards a cov-random activation. This direction was used in Heimersheim and Mendel [6] ("random direction").

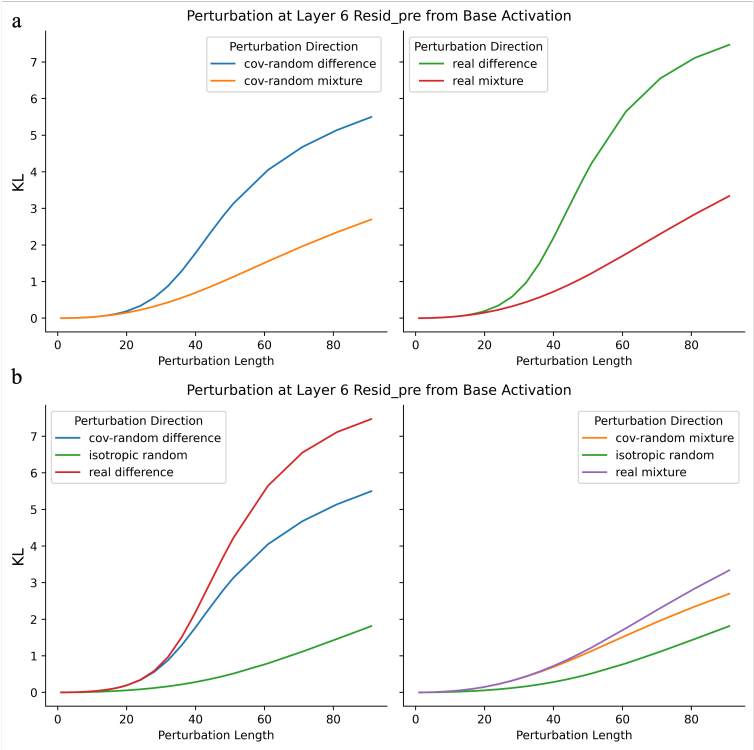

Figure 1: We vary the perturbation length for perturbations in Layer 6 resid_pre. (a) We compare the *difference* versus *mixture* perturbations. For both *cov-random* (left) and *real* (right) cases, the *difference* perturbations have a greater change in model output than *mixture* perturbations. (b) We compare the *cov-random* versus *real* baselines for *difference* (left) and *mixture* (right) types.

- **Cov-random mixture**: Perturb along $d = x_1^{\text{cov-random}} - x_2^{\text{cov-random}}$, i.e. into the difference of two randomly generated covariance matrix adjusted activations.
- **Real difference**: Perturb along $d = x^{\text{real}} - x^{\text{base}}$, i.e. from base towards another real activation. A real activation is sampled from the activations from 2 million tokens. This direction was used in Heimersheim and Mendel [6] ("random other"). Like the *cov-random difference*, this direction contains the original activation.
- **Real mixture**: Perturb along $d = x_1^{\text{real}} - x_2^{\text{real}}$, i.e. into the difference of two real activations (not the original activation). The real activations are sampled from the activations from 2 million tokens. The real mixture no longer contains the original activation.

Mixture directions are useful baselines because, under the Linear Representation Hypothesis, they are a linear combination of true feature directions (see Appendix C) [3].

3.1 Comparing different baselines

On average, perturbation directions that include the negative original activation ($-x^{\text{base}}$) cause a greater change in the model output compared to those that do not include the original activation. In Figure 1a, KL for *cov-random difference* is greater than KL for *cov-random mixture* and the KL for *real difference* is greater than KL for *real mixture*. This finding suggests that the *difference* directions may primarily reflect the subtraction of the original activation, which seems related to the observation in Lindsey [8] that "feature ablation" has a much greater effect than other perturbations including "feature doubling." The result supports the use of *mixture* baselines to ensure a fair comparison with directions like SAE features or SAE errors, which do not necessarily involve the original activation.

Cov-random mixture directions influence the model's output more significantly than isotropic random directions (see the right plot of Figure 1b). This aligns with the hypothesis that isotropy reduces the influence of perturbations on the model's logits. Since *cov-random* directions are derived

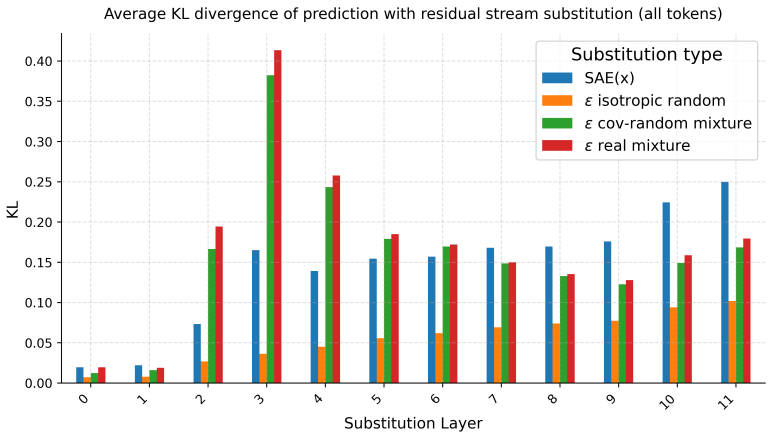

Figure 2: Comparison of the average KL divergence of four different substitution types. On the x-axis we have different GPT2-small layers. SAE from Bloom [11] was used.

from a multivariate normal distribution and real activations are likely more clustered than normally distributed, *cov-random* directions may not be the ideal baseline. Heimersheim and Mendel [6] observed that *real difference* directions had a stronger effect on the model's output than *cov-random difference* directions (as shown in the left plot of Figure 1b), which initially suggested that sampling from real activations might provide a better baseline. However, the minimal difference between *real mixture* and *cov-random mixture* directions (shown in right plot of Figure 1b) suggests that Heimersheim and Mendel [6]'s finding was likely influenced by the negative activation component.

3.2 Revisiting pathological errors under new baselines

We reran the analysis from Gurnee [9], this time incorporating the two new baselines. We also compared multiple SAEs with different $L0$ values. Our results confirmed the original finding that substituting the base activation with the SAE reconstruction, SAE(x), changes the next token prediction probabilities significantly more than substituting an isotropically random point at the same distance ϵ (Figure 2). When perturbing along the *cov-random mixture* or *real mixture* directions, the average KL divergence is generally closer to that of SAE(x). However, there is considerable variance depending on the layer. For Layer 6, the SAE models across $L0$ generally seem to have nearly the same KL as that of cov-random mixture (Figure 5). While this suggests that addressing isotropy mitigates the previously observed pathologically high-KL behavior in SAE errors, questions remain about the dependence observed across different layers.

4 Comparative Analysis of SAEs

Recently, a new type of SAEs called end-to-end SAEs has been introduced [10]. End-to-end SAEs aim to identify functionally important features by minimizing the KL divergence between the output logits of the original activations and those of the SAE-reconstructed activations. There are two variants of end-to-end SAEs: e2e SAE and e2e+ds SAE (where ds is short for downstream). Braun et al. [10] proposed e2e+ds SAEs as a superior approach because it also minimizes reconstruction errors in subsequent layers (whereas e2e SAEs might follow a different computational path through the network). In this section, we will compare traditional (local) SAEs, e2e SAE, and e2e+ds SAE across various $L0$s.

The following are the perturbation directions discussed in this paper:

- **SAE Reconstruction Error Direction**: Perturb along $d = \text{SAE}(x^{\text{base}}) - x^{base}$, i.e. from base activation towards the reconstructed activation.
- **SAE Feature Direction**: Perturb along $d = W_i^{\text{dec}}$, which corresponds to one of the directions i from the SAE decoder matrix W^{dec}. We choose SAE features that are alive, but not active in the given sequence.

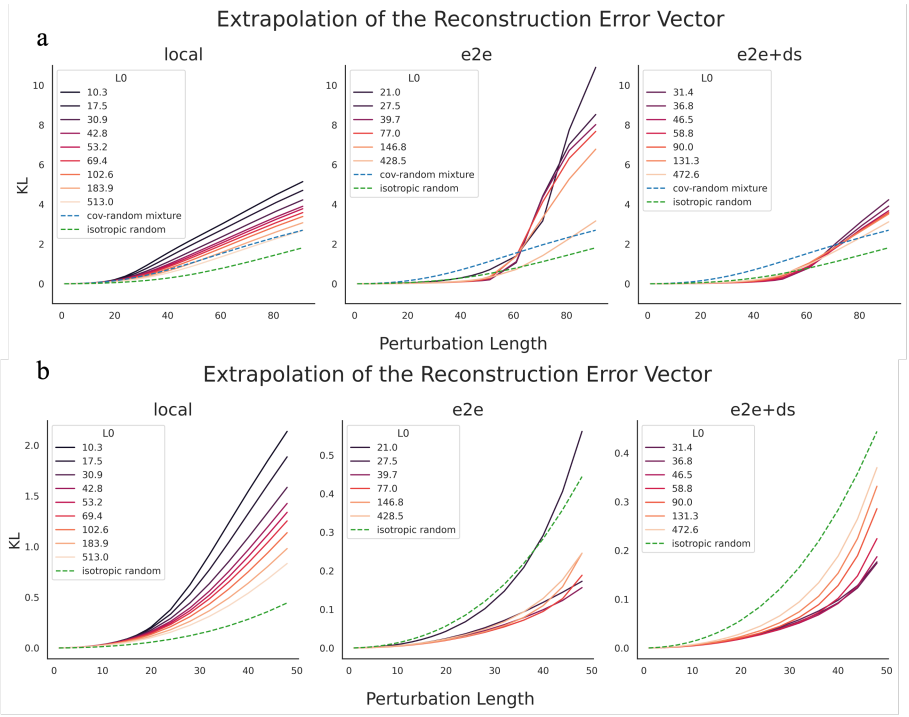

Figure 3: We vary the perturbation length for perturbations in Layer 6 resid_pre. For each columns we show different SAE model types. We compare the SAE reconstruction error directions with *cov-random mixture* and *isotropic random* directions. We color the lines by different $L0$ values of the SAEs. (b) is the same as (a), but with a reduced x-axis limit.

4.1 SAE reconstruction error extrapolation

To gain insight into the model's sensitivity to SAE reconstruction errors, we extend the error directions across various perturbation lengths. The KL curves typically exhibit a consistent pattern: an initial plateau followed by a more linear increase in KL after the plateau. Th plateau aligns with the observation made in Heimersheim and Mendel [6]. For local SAEs, the behavior is straightforward: lower $L0$ corresponds to a stronger perturbation effect (left plot in Figure 3a). For e2e (and e2e+ds) SAEs, the behavior is more complex: the effect of $L0$ at small perturbation scales is the opposite of its effect at larger scales. For perturbation lengths below 50, lower $L0$ results in greater KL divergence for e2e and e2e+ds SAEs, except for $L0$ = 21.0 or 27.5 e2e SAEs (middle and right plots in Figure 3b). For perturbations above 70, lower $L0$ corresponds to a stronger perturbation effect (Figure 3a).

While the curves for the local SAEs are close to the curves for the *cov-random mixture* baseline, the curves deviate a lot for e2e and e2e+ds SAEs. Notably, the curves for e2e and e2e+ds SAEs remain low and then spike up from perturbation length of around 50 (Figure 3a). The former is expected as e2e SAEs generally have a high L2 reconstruction error while having a low KL-divergence).

4.2 SAE feature extrapolation

To explore the functional relevance of SAE features, we extrapolate the SAE feature directions across various perturbation lengths. We select a random SAE feature that is alive, but not active in the given context the token is located in. All SAE features have a greater impact on the model output than *isotropic random* directions (Figure 4). When compared to *cov-random mixture*, the effect varies based on the type of SAE and its $L0$ value. For all three types of SAEs, lower $L0$ corresponds to greater change in model output (Figure 4).

We select a specific $L0$ value to conduct a more detailed comparison of the SAE models ($L0 = 30.9$ for local SAE, $L0 = 27.5$ for e2e SAE, and $L0 = 31.4$ for ds+e2e SAE). Among these, e2e SAE features have the least impact on the model output (Figure 3). At shorter perturbation lengths, local

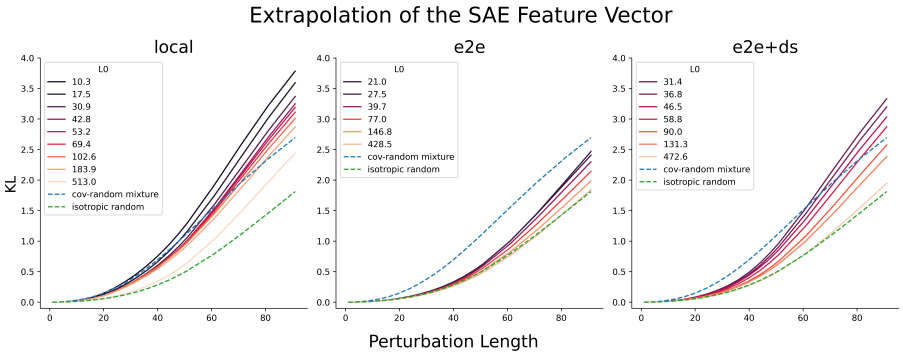

Figure 4: This plot varies the perturbation length for SAE feature directions in Layer 6 resid_pre. For the three columns, we compare the three different SAE model types . We color the lines by different L0 values of the SAEs.

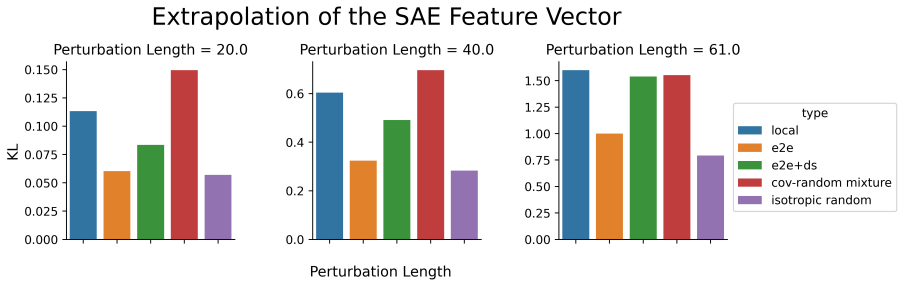

Figure 5: Comparison of the change in model output for various perturbation lengths for different SAE feature directions and baselines in Layer 6 resid_pre.

SAE features influence the model more than e2e+ds SAE features, but this difference shrinks as the perturbation lengths increase. We note that using the same $L0$ may not be a fair way to compare the three SAE models. This is because end-to-end SAEs are known to explain more network performance given the same $L0$ [10].

Discussion The result was initially surprising because we would have expected that end-to-end SAEs would more directly capture the features most crucial for token predictions. Our hypothesis for the explanation for this observation is that e2e SAE features perform worse because they are more isotropic (see Figure 3(a) from [10]). This could be an unintended and undesirable consequence of end-to-end SAEs. While e2e SAE might exhibit this behavior, it is unclear to what extent e2e+ds SAE also does this.

5 Conclusion

Summary In this work, we run sensitive direction experiments for various perturbations on GPT2-small activations. We make several findings. First, SAE errors are no longer pathologically large when compared to more realistic baselines. Second, GPT2-small is more sensitive to lower $L0$ SAE features. Third, End-to-end SAE features do not exhibit stronger effect on the model output than traditional SAE features.

Limitation In this post, we primarily use the mean (of KL) as our main measure. However, relying solely on the mean as a summary statistic might oversimplify the complexity of sensitive directions. For instance, the overall shape of the curve for each perturbation could be another important feature that we may be overlooking. While we did examine some individual curves and observed that real mixture and cov-random mixture generally exhibited greater model output change compared to isotropic random, the pattern was not as clear-cut.

References

[1] Sanjeev Arora et al. “Linear algebraic structure of word senses, with applications to polysemy”. In: *arXiv [cs.CL]* (Jan. 2016).

[2] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. “Feature Visualization”. en. In: *Distill* 2.11 (Nov. 2017), e7.

[3] Trenton Bricken et al. *Towards Monosemanticity: Decomposing Language Models With Dictionary Learning*. https://transformer-circuits.pub/2023/monosemantic-features. Accessed: 2024-9-17. 2023.

[4] Hoagy Cunningham et al. “Sparse autoencoders find highly interpretable features in language models”. In: *arXiv [cs.LG]* (Sept. 2023).

[5] Adly Templeton et al. *Scaling Monosemanticity: Extracting Interpretable Features from Claude 3 Sonnet*. en. https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html. Accessed: 2024-9-17. 2024.

[6] Stefan Heimersheim and Jake Mendel. “[Interim research report] Activation plateaus & sensitive directions in GPT2”. In: *LessWrong* (2024).

[7] Kaarel Hänni et al. “Mathematical Models of Computation in Superposition”. In: *ICML 2024 Workshop on Mechanistic Interpretability*. June 2024.

[8] Jack Lindsey. *How Strongly do Dictionary Learning Features Influence Model Behavior?* en. https://transformer-circuits.pub/2024/april-update/index.html#ablation-exps. Accessed: 2024-9-17. 2024.

[9] Wes Gurnee. *SAE reconstruction errors are (empirically) pathological*. https://www.lesswrong.com/posts/rZPiuFxESMxCDHe4B/sae-reconstruction-errors-are-empirically-pathological. Accessed: 2024-9-15. Mar. 2024.

[10] Dan Braun et al. “Identifying functionally important features with end-to-end sparse dictionary learning”. In: *arXiv [cs.LG]* (May 2024).

[11] Joseph Bloom. “Open Source Sparse Autoencoders for all Residual Stream Layers of GPT2-Small”. In: *LessWrong* (2024).

A Additional Figures

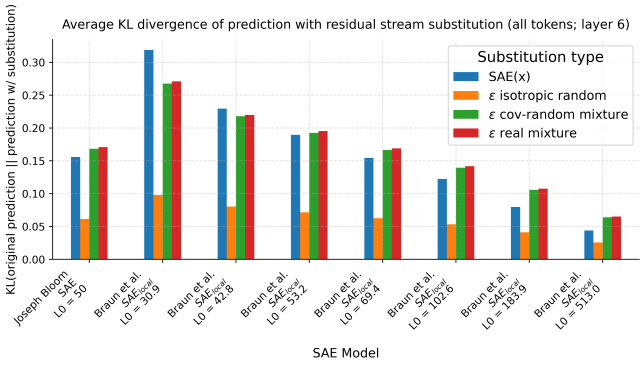

Figure 6: This plot compares the average KL divergence of four different substitution types. On the x-axis we have different SAE models. Joseph Bloom SAE was the SAE used in the original Gurnee 2024 paper. The local SAE from Braun 2024 refers to traditional SAEs. The isotropic random substitutions have a much smaller average KL divergence than other substitution types. Across the various SAE models, the three other substitution types (SAE(x), cov-random mixture, and real mixture) have generally similar average KL divergence.

B Experiments with 15 million tokens

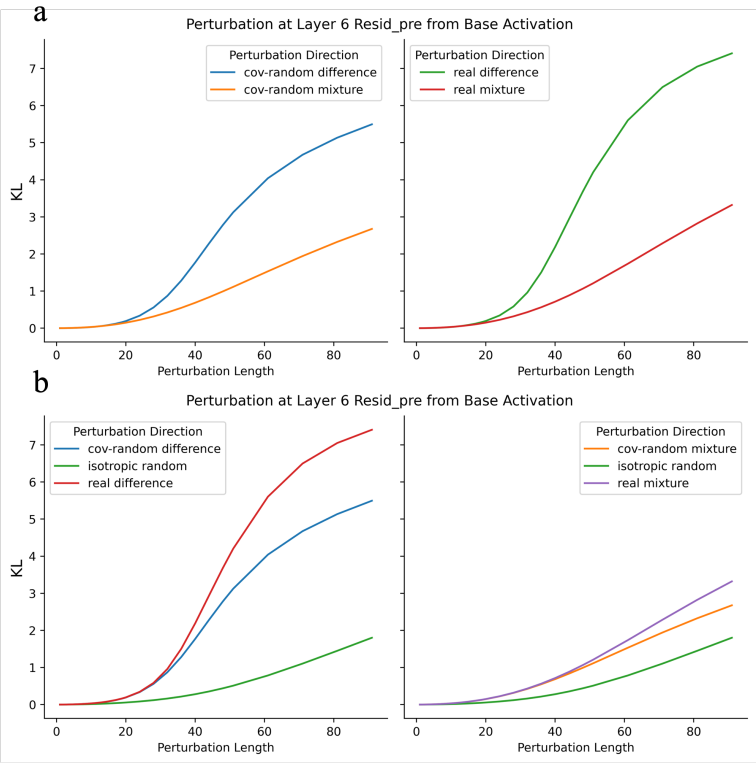

Figure 7: We rerun figure 1 experiment with 15 million tokens.

C Why the Difference Between Two Activations?

Under the Linear Representation Hypothesis (LRH), we can represent an activation x as

$$x \approx b + \sum_i f_i(x) d_i,$$

where $f_i(x)$ is the activation of (hypothetical) feature i, d_i is the unit "direction" vector of feature i, and b is the bias.

If we take the difference between two activations x_1 and x_2, we get:

$$x_1 - x_2 \approx \sum_i [f_i(x_1) - f_i(x_2)] d_i$$

Therefore, assuming LRH, subtracting any two real activations is a linear combination of (hypothetical) true features without the bias term. We note that this will also include "negative features," which is not expected to be as meaningful in the models.