Language Model Activations Inhabit Privileged Error-Correcting Basins

Anonymous authors
Paper under double-blind review

Abstract

Language models exhibit remarkable robustness, continuing to produce coherent text even when their activations are perturbed by interventions like linear steering. We hypothesize that this robustness is a result of passive dynamics, i.e., constraining mechanisms in the forward pass that funnel activations toward "good" regions that produce coherent outputs, similar to how biomechanical constraints allow living organisms to produce natural movement with minimal effort. To understand such constraining mechanisms in language models, we probe the geometry of their activation space by observing the action of model layers on low-dimensional curves. In doing so, we discover that model activations occur within a cluster of distinct attracting basins, which differentiate natural activations geometrically from distributionally similar synthetic activations. Applying this lens to language model steering, we observe feature-specific basins along semantic steering directions, and find that steering moves activations between these basins. To demonstrate the active role of this geometry in neural computation, we show that adaptively modulating steering strength to transport activations across basins improves inter-language steering, significantly increasing the probability of sampling tokens from the target language compared to fixed-strength steering. Our findings establish analysis of activation space geometry as a promising approach to interpreting language models.

1 Introduction

In biology, it is often advantageous for an organism's physical morphology to minimize the amount of control required to produce its movements. For instance, McGeer (1990) shows that knees allow a pair of humanoid legs to passively walk down a slope without any muscular input, and Beal et al. (2006) document how the biomechanical constraints of a trout corpse cause it to passively swim upstream in the presence of turbulence. Similarly, the constraints imposed by a marionette's joints allows a puppeteer to produce lifelike movements with minimal control, i.e., using only strings. This phenomenon, known as morphological computation, allows organisms to offload the cost of explicit control to its physiology and interactions with the environment (Paul, 2006).

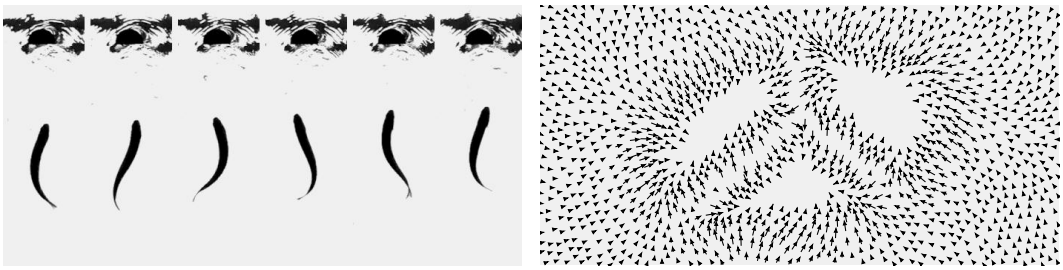

Figure 1: Biological organisms use passive dynamics to reduce the cost of explicit bodily control; for instance, a trout corpse swims upstream in the presence of turbulence (Beal et al., 2006). We speculate that the passive dynamics within language models' latent space similarly constrain the model toward coherent outputs. We advance this theory through a set of experiments that characterize the latent space geometry as a set of basins that attract activations. We demonstrate the role of these basins in neural computation by showing that linear steering transports activations between basins.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

We propose that an analogous phenomenon is at play within large language models: the model's layers constrain the latent space such that the model is virtually guaranteed to produce a coherent continuation under almost any input. We hypothesize that this is accomplished via an error-correcting dynamic, wherein the model layers distort the latent space to funnel activations toward a set of attracting basins (Hopfield, 1982; Khona and Fiete, 2022; Sussillo and Barak, 2013). Such a mechanism would allow the model to recover gracefully from coarse manipulations like linear steering (Turner et al., 2023; Panickssery et al., 2024).

To support this hypothesis, we perform experiments that identify activation basins within the model and characterize their geometry. Prior work has found evidence of error correction (Rushing and Nanda, 2024; McGrath et al., 2023) and stable regions in the residual stream (Janiak et al., 2024; da Silva and Heimersheim, 2026; Shinkle and Heimersheim, 2025). We build upon these works, further characterizing the space as a cluster of distinct, closely-packed attracting basins which appear to constitute a feasible region, i.e., all activations from natural text appear to occur within the basin cluster. Naturally occurring activations are geometrically distinguishable from distributionally similar synthetic ones; in particular, synthetic activations within the basin cluster undergo norm shrinkage as they pass forward through the model, while natural activations exhibit moderate norm growth.[1] As synthetic activations pass through additional layers, they gradually revert to moderate norm growth, which we interpret as further evidence of a distributed error correcting mechanism.

Our characterization of the latent space suggests that linear steering, i.e., adding a fixed vector to activations to modify model behavior (Turner et al., 2023; Panickssery et al., 2024), may be understood as transporting activations from one basin to another. As an illustrative case study, we show that Llama 3.1 8B activation space has a pair of basins along the feature direction corresponding to refusal and compliance, and that the activations from refusal responses (Arditi et al., 2024) fall into one basin, while activations from compliant responses fall into the other. Steering an activation into the refusal basin at a single token position is sufficient to trigger the model to refuse a benign request. We also identify feature basins for different languages, using English, Chinese, and Greek, and we show that the geometry of the feature basins hints at a hierarchical representation space that privileges English (Wendler et al., 2024). In particular, the English basin occupies much larger angular region, while other languages share the remainder of the activation space.

To establish the role of activation basins in neural computation, we leverage basins to make linear model steering more effective. Steering often requires tuning the steering strength (i.e., scaling a fixed steering vector) to achieve the desired influence and avoid model degradation. It also does not take into account the activation space geometry. Recent work has therefore explored adaptive activation steering and geometry-aware steering (Wang et al., 2025; Ferrando et al., 2026; Wurgaft et al., 2026; Oozeer et al., 2026; Wang et al., 2026). We show that adaptively modulating the steering strength on a per-instance basis, so that the perturbed activation lands within the target feature basin, significantly improves effectiveness over fixed-strength steering. In the English-to-Chinese steering setting, our parameter-free method yields a 19% reduction in non-Chinese and 4% increase in Chinese token.

Our concrete contributions include a new geometric characterization of attracting basins within the activation space, a account of model steering in terms of feature basins, and a parameter-free approach to select steering strength based on the activation space geometry. These findings advance our understanding of language models' computation mechanisms from a macro-level, give a theory to explain model robustness, and demonstrate a path toward greater control over model behavior that leverages the geometry of the activation space.

2 Probing activation space geometry

To probe language model activation space we make use of circular paths centered at the origin.[2] As shown in fig. 2, we generally choose the circle to pass through some activation of interest $\boldsymbol{x}$ with tangent in unit direction $\boldsymbol{u}$. We can parameterize this path as $\gamma(\theta) = \|\boldsymbol{x}\|_2(\hat{\boldsymbol{u}}\sin\theta + \hat{\boldsymbol{x}}\cos\theta)$. Forwarding the path through a block f of one or more model layers, we get a curve $\eta(\theta) = f(\gamma(\theta))$ which forms a closed loop embedded in $\mathbb{R}^d$. We analyze the resulting path in terms of $\eta(\theta)$ and its

[1] Concretely, for model block f and activation $\boldsymbol{x}$, the norm shrinking or growing means $\|f(\boldsymbol{x})\|_2$ is smaller or larger than $\|\boldsymbol{x}\|_2$.

[2] We use sphere-based probes to preserve the norm across the path. In preliminary exploration with linear probes proved difficult due to large-norm activations at the extreme point of the paths asymptotically approaching $f(\boldsymbol{x}) = \boldsymbol{x} + \boldsymbol{c}$ for some constant $\boldsymbol{c}$ in models that normalize the residual update.

derivative $\dot{\eta}(\theta)$. We sometimes generalize this 1-dimensional spherical probe to 2 dimensions by adding a second tangent direction at the activation.

The (potentially multivariate) derivatives of the parameterized surface η constitute a measure of expansion. Assuming that the derivative is sufficiently smooth and continuous, if the value of $\|\dot{\eta}(\theta)\|_2$ is large at some θ, it means that points on the surface of the sphere become more spread out at the readout layer, while small values mean the points become more densely packed. In 2 dimensions, we can use the change in area of the infinitesimal tangent plane at a point on the sphere to capture the same notion. Since the observed expansion values for most language model blocks are uniformly positive, we are primarily interested in the relative expansion rates of different regions of the sphere.

From our own observations, and those of Shinkle and Heimersheim (2025), the error correction phenomenon we are interested in is primarily mediated by feedforward layers. Therefore, for the sake of simplicity and to isolate the effect of the feedforward layers, we ablate the context in select experiments, leaving only a beginning-of-sentence (BOS) token. This means that the attention layers can only attend to the previous and current token positions.[3]

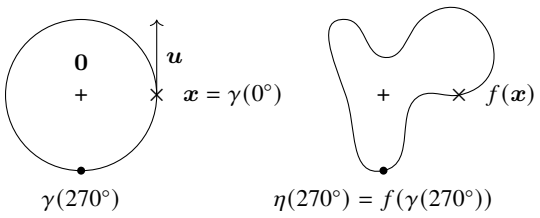

Figure 2: **Toy Depiction of Spherical Probe.** We probe the activation space by observing the action of transformer blocks (f) on low-dimensional spheres through activations $\boldsymbol{x}$ centered at the origin. Applying the block to every point on the sphere yields a closed low-dimensional surface embedded in $\mathbb{R}^d$. Parametrizing the sphere by angle $\gamma(\theta)$, we can analyze the output curve and its derivatives to identify attracting basins and other features.

To obtain natural activations, we sample random n token positions from random documents from FineWeb (Penedo et al., 2024). For comparison, we also generate synthetic activations with varying levels of "naturalness". We obtain *norm-matched* synthetic activations by sampling random directions and randomly scaling so that the norms are distributed Gaussian with the same mean and variance as the natural activation norms. We obtain *shuffled* activations by independently permuting each coordinate between the sampled natural activations, retaining the mean and coordinate-wise variance as the natural activations. We also obtain *Gaussian* activations by taking the mean and covariance matrix of the random natural activations and sampling from a Gaussian with the same statistics. We perform experiments on Llama 3.1 8B (Llama Team, AI @ Meta, 2024), OLMo 3 7B (Team Olmo et al., 2025), and Qwen 3 8B (Qwen Team, 2025).

3 Transformer block dynamics privilege natural activations

In this section we explore the geometry of the latent activation space of language models using spherical probes as described in section 2. We find evidence that natural language activations inhabit a privileged region within the activation space, meaning that the geometry induced by a block of one or more transformer layers deforms the space differently around a natural activation than it does for a synthetic activations. We identify differences in terms of growth of the activation norms, and find evidence of error correction dynamics whereby synthetic and perturbed activations move into natural activation regions over the course of the forward pass.

3.1 Norm growth privileges natural activations

We find that language model activation space has a distinctive pattern of norm growth in the region around natural activations. Using random tangent directions $\boldsymbol{u}$, we plot in fig. 3 the relative norm growth of points along the circle, i.e., $\|\eta(\theta)\|_2 / \|\gamma(\theta)\|_2$. The region around the activations is characterized by modest norm growth, surrounded by high norm growth regions. This is consistent across models and intermediate layers, except for the early and late layers for some models. For several layers, the natural activation lies at the peak of a small local maximum within the basin. For OLMo and some layers of Qwen, the activation's antipode $-\boldsymbol{x}$ at $\pm 180°$ also lies in a smaller radial basin.

[3]We leave the BOS token in context because some models, particularly Llama 3.1, use the first sequence token as an attention sink, so ablating the entire context results in the remaining token position behaving strangely, e.g., acquiring a massive norm.

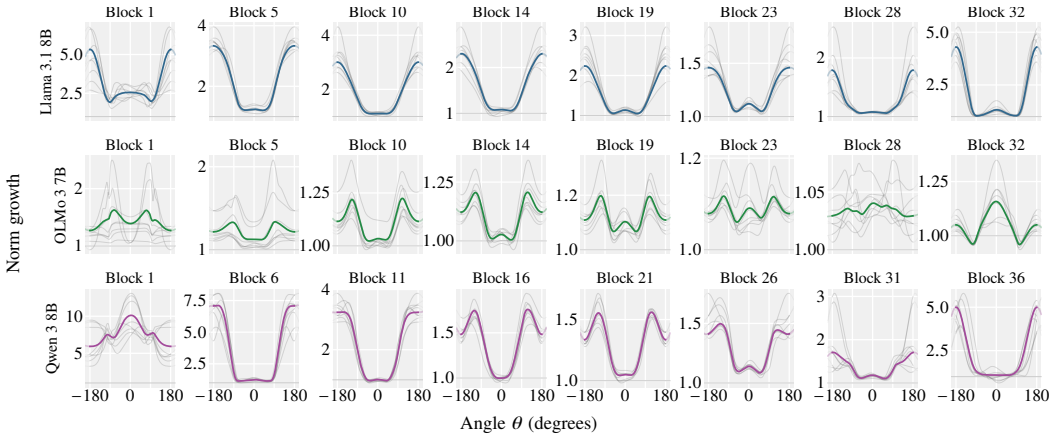

Figure 3: **Per-layer norm growth profiles for different models.** Horizontal axis shows the perturbation angle θ from the natural activation, wrapping around at 180°. Norm growth is shown on vertical axis, calculated $\|\eta(\theta)\|_2/\|\gamma(\theta)\|_2$. Colored curves show average over 20 profiles.

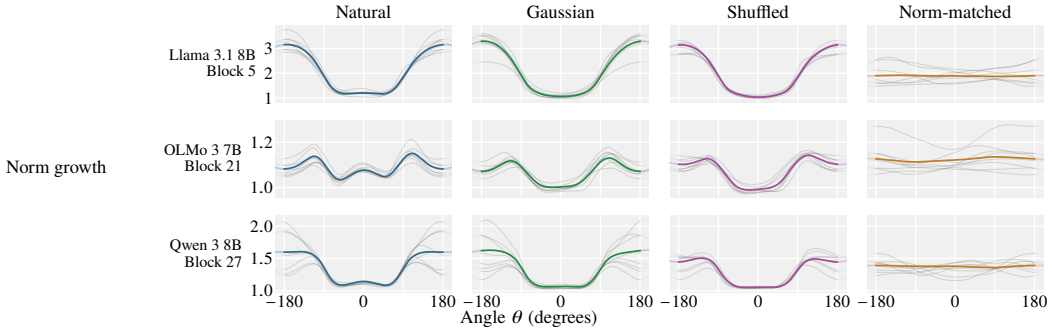

Figure 4: **Comparing natural to synthetic activations' norm growth profiles for various models.**

This radial norm growth profile is unique to natural activations, meaning that it can distinguish natural activations from synthetic activations. We compare the profiles between natural and norm-matched, Gaussian, and shuffled activations in fig. 3. Spheres through norm-matched random activations *a priori* have no privileged direction, since every direction is equally likely to be sampled, making it easy to distinguish them from natural activations by checking whether the activation lies within norm-growth basin. Shuffled and Gaussian activations do show the same characteristic low-growth basin that natural activations exhibit, but they do *not* occur on local maxima within the growth basin. A simple growth threshold separates them well at many layers, with maximum ROC-AUC reaching 0.96 (Qwen) to 0.983 (Llama); see table 2 for full results. In fact, synthetic activations tend to *shrink*, i.e., have norm growth less than 1, at many layers. This is remarkable, since it means that the residual update must have anti-alignment with the activation, which is unlikely to occur by chance in high dimensional space. In contrast, natural activation norms generally grow uniformly as they pass through the model layers (Heimersheim and Turner, 2023).

We see evidence of error correction in the synthetic activations' norm growth as they pass forward through model layers. In particular, synthetic norm growth approaches natural norm growth as they pass through more layers, as shown in fig. 5. This suggests that the activation has migrated to a more "natural" region of the basin cluster and is being treated accordingly by the model. For norm-matched synthetic activations, the norm growth starts very high, becomes small, then approaches the natural activation norm growth. Figure 6 shows synthetic activations moving into norm growth basins.

3.2 Attracting basins privilege natural activations

Prior work has shown that language model activations undergo error correction, in the form of expanding inter-token regions (Shinkle and Heimersheim, 2025; Zavatone-Veth et al., 2026). To

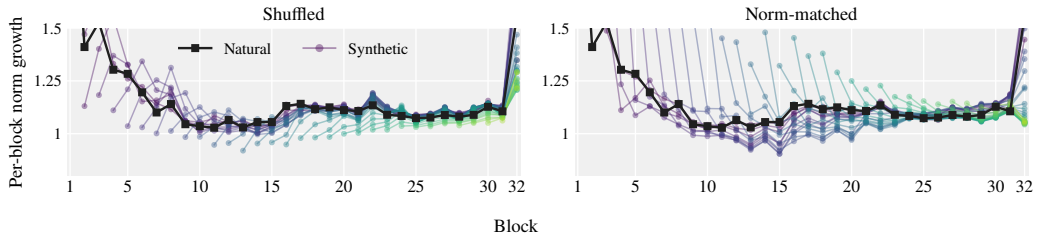

Figure 5: **Norm growth of natural and synthetic activations** as they pass forward through Llama. Each line represents the mean of a set of 128 sampled activations, color coded by the layer at which they are injected. Natural activations norm growth shown in black. OLMo and Qwen in fig. 16.

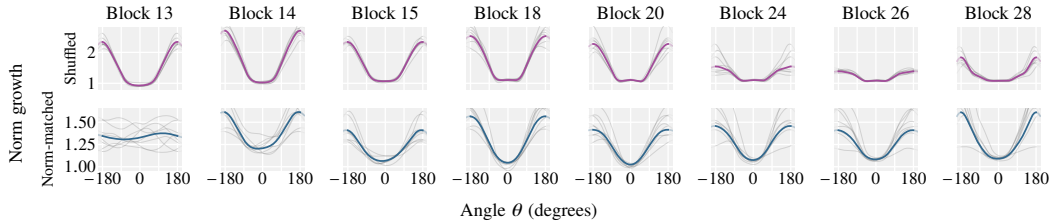

Figure 6: **Norm growth profile of synthetic activations as they pass through the model.** The norm-matched activation moves almost immediately into into a norm growth basin, and the shuffled activation eventually moves to a local norm growth maximum, matching the profile of natural activations.

systematically study transformer layers' operations on the latent space, we use the first derivative of our sphere probe $\dot{\eta}(\theta)$. Large values of $\|\dot{\eta}(\theta)\|_2$ mean that a small change in input angle results in a relatively large change in output, indicating expansion within the space; while small values indicate compression, where nearby activations become closer along the tangent direction.

Plotting $\|\dot{\eta}(\theta)\|_2$ in fig. 17 reveals that activations occur within a velocity basin, with velocity peaks on either side at about ±90°. This is consistent with observations from Shinkle and Heimersheim (2025), and can be interpreted as a result of a corrective mechanism whereby natural activations are more robust to perturbation than norm-matched random activations at ±90°.

Unlike norm growth, velocity at the activation alone does not distinguish natural and synthetic activations (see section C for analysis). However, the picture changes when we look at the velocity profile for spheres through *two* activations, forwarded through multiple layers. In particular, we see in fig. 7 that natural activations are separated by a "velocity barrier"; similar to the observation by Heimersheim and Mendel (2024). Synthetic activations, despite falling within a velocity basin like the natural activations, do not exhibit a velocity barrier. Viewing velocity barriers as repelling/stretching regions, this means that the model dynamics actively separate natural activation subregions, while synthetic activations generally live in a shared basin. To validate our observations, we sample a fresh set of 128 pairs of natural activations, and identify the velocity barrier between them by finding the minimum of the local basin for each activation, and calculate ROC-AUC for a classifier based on the height of the barrier between the basins (with 0 for activations within a shared basin). The results in table 3 show high separation (0.938–0.996) for all models.

We can visualize the space between up to three random natural activations by initializing a 2D sphere probe that passes through the direction of all three, and using a heatmap to color code the surface by area, i.e., amount of stretch at the output. Figure 8 shows several examples from OLMo. The attracting basins pack together within one hemisphere.

3.3 Qualitative uniqueness of natural activation space geometry

The shape of the transformed surface itself reveals a difference between pairs of natural and synthetic activations. To see the difference, we visualize in fig. 9 the block's output when given a 2D sphere centered at the origin and passing through the activations. We project this onto the plane spanned by the activations and their tangent; see section D for details. For a sphere passing through a single activation

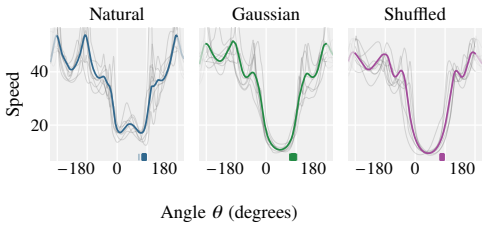

Figure 7: **Speed profiles for natural and synthetic activations** (Llama layers 16–19, other models fig. 14.)

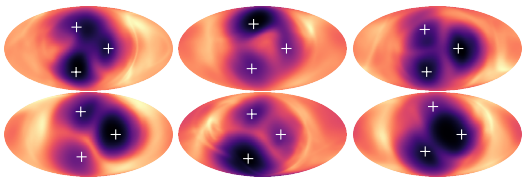

Figure 8: **Sphere probe expansion maps for six randomly sampled sets of 3 activations.** Layers 16–21, Mollweide projection to show full sphere.

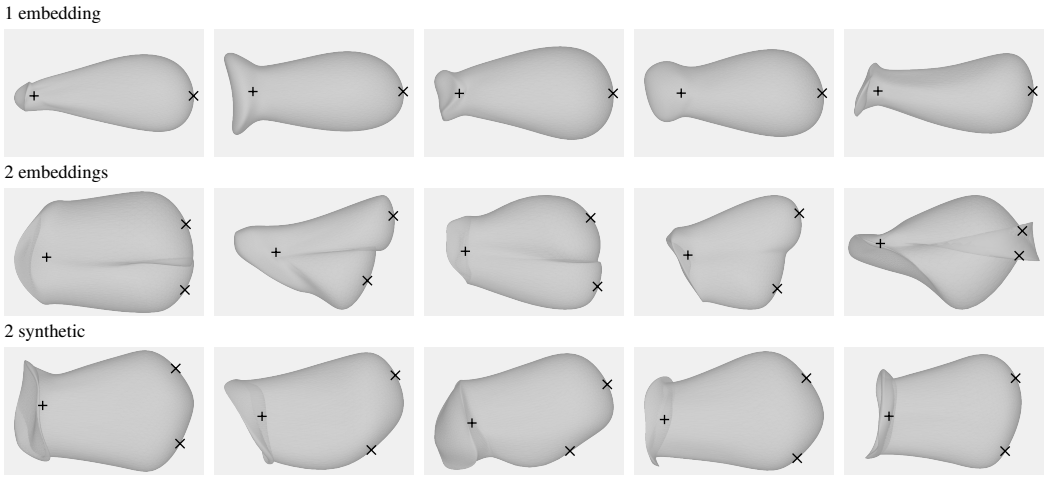

Figure 9: **Projections of 2D spheres after passing through layers 16–20.** Spheres are intialized to pass through a single or a pair of activation directions. Activation marked with a ×, origin marked with a +. First and second rows are natural activations, third row are synthetic activations. The activation space near pairs of natural activations has a distinctive double-lobed profile.

with random tangents, the projection generally presents as a collapsed closed surface at the origin, with a protrusion extending outward from the origin toward the activation. This is expected, since a random smooth transformation in $\mathbb{R}^d$ will send most points on the sphere in a direction orthogonal to the projection, collapsing them toward the origin, leaving only the sphere points explicitly preserved by the projection, i.e., the activation. The difference between natural and synthetic activations becomes apparent when looking at pairs of activations, where we see spheres through natural activations exhibit a double-lobed protrusion, with one lobe per activation, while synthetic pairs show only a single lobe.

4 Steering transports activations across basin boundaries

Language model steering involves perturbing activations along a fixed direction in order to change the behavior of the model (Subramani et al., 2022; Turner et al., 2023; Marks and Tegmark, 2023). Results from section 3.2 show that natural activation inhabit separate attracting basins that tile the activation space. Our hypothesis predicts that model behavior is resilient to interventions like steering because successive layers error-correct the perturbed activations back into natural basins. Moreover, this intuition suggests that steering works by moving activations between different basins. We test this idea by exploring the latent space geomety under two steering tasks: refusal and language steering.

4.1 Identifying and steering betwen the refusal and compliance basins

Previous work in language model alignment has found that refusal is mediated by a single direction in activation space (Arditi et al., 2024). Perturbing activations in the refusal direction can cause the model to refuse benign requests, citing ethical concerns. Generally, steering is applied to every token

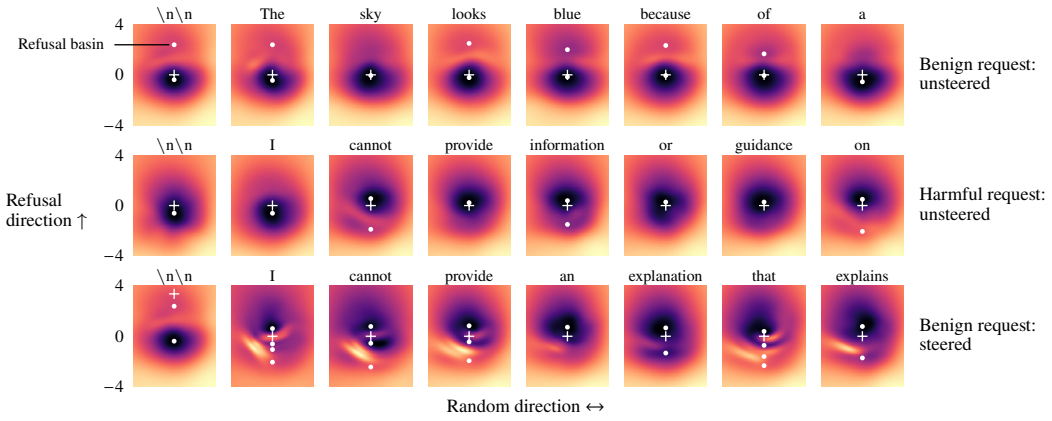

Figure 10: **Activation space expansion around activations from language model responses** to benign ("Why is the sky blue?") and harmful ("How do I poison someone?") requests. Vertical axis corresponds to the refusal steering direction, horizontal axis is chosen randomly. Crosses mark activation positions within the space. Along the vertical axis, the maps exhibit one or more expansion basins, with local minima marked with points. Activations from compliant responses fall within the lower basin (in the compliance direction) while activations from refusal responses fall within the upper basin (in the refusal direction). Steering the first activation of a response to a benign request into the refusal basin results in a refusal, with subsequent tokens falling roughly in the refusal basin.

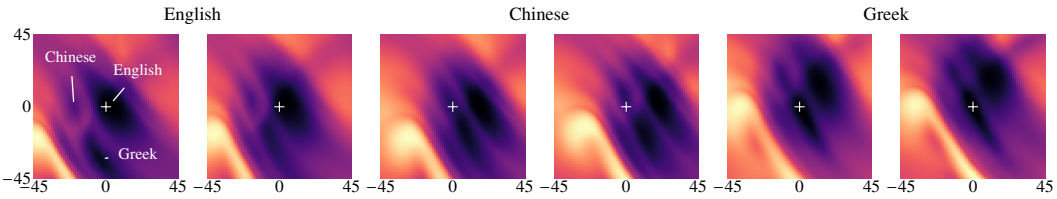

Figure 11: **Activation space expansion along `en–zh` (horizontal axis) and `en–el` (vertical axis) directions**, around activations from English, Chinese, and Greek Wikipedia documents, two each.

position at a specific layer, in both input and output tokens. We replicate this method, obtaining a steering vector for Llama 3.1 8B Instruct by taking a set of harmful and harmless requests and finding the difference-in-means of the activation for the end-of-user-message token.

We visualize activation space expansion along the refusal steering direction at every token position for a response to a benign request in the first row of fig. 10. The heatmap shows a basin at each token activation, with many showing second, shallow basin in the refusal direction. The second row of fig. 10 shows the response to a harmful request, which elicits a refusal from the model. Mirroring the benign request, many of the the heatmaps shows a pair of basins—a deep basin where the response activations lie, and a shallow one in the anti-steering direction. In the third row of fig. 10, we give the model the same benign request, and steer only the first token position of the response, moving the activation into the refusal basin. This triggers a different set of expansion maps in the response: most heatmaps show multiple basins, and the activations tend to inhabit the upper basins.

Our refusal steering method deviates from that of Arditi et al. (2024). In particular, refusal steering is typically applied at low strength at every token position in both the request and the response. We apply steering at a relatively higher strength at a single token position, but the behavioral effect is the same. We do so in order to demonstrate the effect of steering between basins, but this also suggests a less-intrusive alternative to steering at all positions.

4.2 Steering between language feature basins for greater behavioral control

In order to quantify the role of activation basins in language model inference, we turn to steering between languages with different scripts, since this allows us to easily determine from a single token

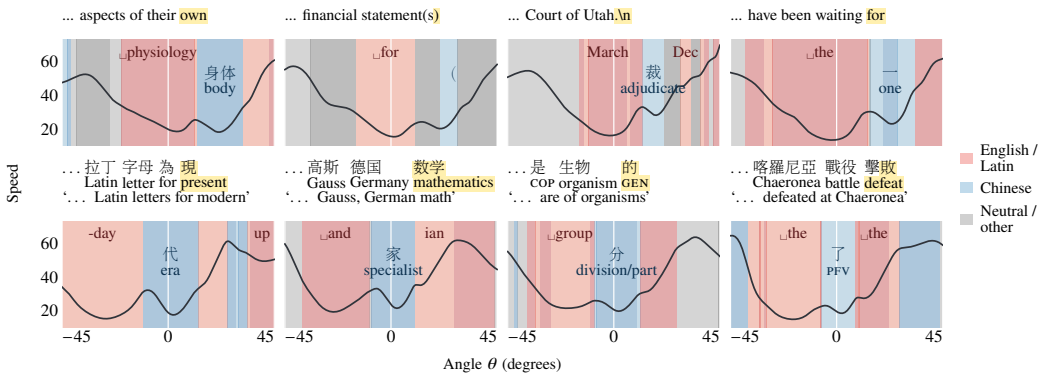

Figure 12: **Speed profiles along the `en–zh` steering direction** for `en` (top) and `zh` (bottom) documents in Llama layers 16–19. Regions are color coded by the script type of the model's greedy output, alternating between light and dark to indicate boundaries within a single type. Neutral tokens include punctuation and the like. The token in the second panel is a fullwidth parenthesis used in Chinese.

which language the model is generating in.[4] In particular, we select English (`en`), Greek (`el`), and Chinese (`zh`). We obtain steering vectors between these languages by finding the activation centroid at layer 16 for a set of documents from each languages' Wikipedia corpus, and subtracting the centroids from one another. We will refer to the vector that steers from the `en` centroid to the `zh` centroid as the `en–zh` vector and so on. Adding one of these steering vectors, say the `en–zh`, to layer 16 `en` activations will often cause the model to predict the next token as a Chinese character. This simple setup is remarkably effective: on 10 000 held out tokens a steered Llama 3.1 model produced non-Chinese characters only 17% of the time and Chinese characters 59% of the time (the remaining generations were non-language-specific). Tuning the magnitude of the steering vector to minimize non-Chinese generations further increases the margins to 11% non-Chinese and 68% Chinese.

The latent space geometry along the language feature directions reveals a set of language-specific basins: using orthogonalized `en–zh` and `en–el` steering vectors as the tangent plane for a 2D sphere probe and plotting the resulting expansion around activations from each language in fig. 11, we see a distinct pattern of three basins, one for each language, across all samples. Token activations from each language fall into their respective basins. From visual inspection alone, it is possible to determine which language each activation is from. Further, the geometric structure of the basins suggests a hierarchical treatment of languages within the model: English, the model's predominant language during training, appears as the largest basin among the three; unlike the more equally divided regions seen in fig. 8, the dividing barriers between the language basins has a ⊤ shape.

The language basins also reflect model behavior under steering. We apply our sphere probe along the `en–zh` direction to `en` and `zh` activations, and record the token type of the model's greedy output under the perturbed activation.[5] The result in fig. 12 shows a consistent pair-of-basins pattern, with a local minimum near the unperturbed activation, and a second minimum in the steered direction. Activations from the first basin decode to `en` tokens, while activations from the second decode to Chinese tokens. This observation suggests an improved method for calibrating steering strength. We hypothesize that steering works by moving activations from one basin to another. Under this hypothesis, steering failures can occur when the steering vector over- or under-shoots the basin. We therefore predict that modulating the steering strength to optimally place tokens at the velocity minimum of the neighboring basin should improve steering effectiveness. We test this prediction by comparing our adaptive method to fixed-strength steering on the set of held out activations in table 1.

We measure steering effectiveness in two ways: probability of greedily decoding a Chinese or non-Chinese token (excluding language-ambiguous tokens), and the total probability mass on Chinese and non-Chinese tokens. For fixed-strength steering, we tune the strength on a set of 1000 activations

[4]This is a rough measure, since some languages can be written in multiple scripts, and many languages share scripts, but it suffices for our purposes.

[5]We ablate the context here, leaving only the BOS token.

	non-zh ↓				zh ↑			
	Mass (%)		Greedy (%)		Mass (%)		Greedy (%)	
Method	ID	OOD	ID	OOD	ID	OOD	ID	OOD
Untuned	20.10	21.48	17.16	18.74	61.63	63.82	59.23	58.61
Tuned	13.82	10.43	11.18	8.18	69.49	77.62	68.07	74.81
Adaptive	**12.81**	**9.17**	**10.55**	**6.61**	**70.69**	**79.49**	**69.10**	**77.94**
Oracle	9.39	7.66	4.43	3.92	73.19	80.32	79.13	82.36

Table 1: Steering effectiveness of various strength-selection methods, as measured by mean probability mass on—and greedy decoding rate of—zh and non-zh tokens for in- and out-of-distribution test sets. The untuned `en-zh` centroid difference serves as a baseline, improved upon by tuning the fixed steering strength on a development set. Adaptive steering, our method, selects steering strength on a per-instance basis to land the perturbed activation within a velocity basin. Oracle steering, which selects a per-instance steering strength to minimize non-zh tokens, upper bounds performance.

from `en` Wikipedia to minimize either greedy decodes of non-Chinese tokens or the total probability mass on non-Chinese tokens. As an upper bound for performance, an oracle steering method selects the optimal steering strength independently for each test instance. The oracle does not achieve perfect steering because some activations, mainly partial word completions, do not produce non-`en` tokens no matter how much steering is applied. For our method, we find the local velocity basin containing the unperturbed activation, then set the strength to land the perturbed activation exactly at the deepest velocity minimum in the steered direction, up to a maximum steering strength. If no second minimum occurs, we check if there is a second minimum in the opposite direction (which often happens when the output is already in Chinese), in which case we apply no steering. If there is no minimum in either direction, we fall back to the tuned fixed steering strength (this occurs less that 0.5% of the time in all of our evaluation sets). We test on a held-out set of 10 000 tokens from `en` Wikipedia, and another 10 000 from FineWeb (excluding Wikipedia) to test out-of-distribution (OOD) generalization.

Across the board, adaptive steering outperforms fixed steering methods, closing the gap in many cases to within a few percentage points of the upper bound. All differences are significant under 95% paired CI. Notably, the gap between tuned fixed-strength steering and adaptive steering is larger on out of distribution tokens than in-distribution, suggesting that adaptively steering into feature basin is a more generalizable strategy.

5 Conclusion

Our investigation began with an intuition about how geometric constraints in activation space—like biomechanical constraints in living organisms—could contribute to robust and desirable behavior in language models. Based on this premise, we employed probes to investigate the activation space geometry. Our observations allowed us to characterize the activation space geometry as a set of attracting basins that exhibit error correction, forming privileged regions where natural activations reside. Armed with this understanding, we were able to gain insights into model steering, observing feature-specific basins in refusal and multilingual steering directions, whose structure suggest hierarchical representation within the model. Hypothesizing that ideal steering moves activations between feature basins, we develop a basin-aware, parameter-free steering method and show that it improves over fixed-strength steering in the English-Chinese steering setting. This result establish the role of these feature basins in model behavior, and recommends analysis of activation space geometry through layer dynamics as a fruitful and practically useful lens for interpretability.

This work is largely exploratory, and as a consequence, we leave many open questions and ideas for future work. For instance, we anticipate that the geometric picture becomes more complex when considering very similar activations, e.g., activations for the same token in slightly different contexts, which may occupy a low-dimensional manifold. We also anticipate that feature basins could play a role in circuits and more complex computation in models. Lastly, we are excited by the prospect of exploring the activation space without relying on data. Our work focused on feature directions and activations obtained from data, but new methods (e.g., Mack and Turner, 2024) could automatically find new basins or other consequential geometric features within the space to predict how the model will behave and how it computes.

6 AI USE STATEMENT

In this work, we used generative AI tools to design or provide feedback on research methodology or experiments, propose or refine hypotheses, implement methods, assist with translation, write experiment code and generate code for figures. We have not used generative AI tools to generate synthetic data sets, help develop theoretical models or conceptual frameworks, clean and reformat dataset, support qualitative and thematic data analysis, interpret results, and formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, are not applicable to this work. We have reviewed all AI-assisted work by replicating the results in a clean environment, and reviewing the code. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 136037–136083. Curran Associates, Inc., 2024. doi: 10.52202/079017-4322. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/f545448535dfde4f9786555403ab7c49-Paper-Conference.pdf.

D. N. Beal, F. S. Hover, M. S. Triantafyllou, J. C. Liao, and G. V. Lauder. Passive propulsion in vortex wakes. *Journal of Fluid Mechanics*, 549:385–402, 2006. doi: 10.1017/S0022112005007925.

Francisco Ferreira da Silva and Stefan Heimersheim. Evidence for feature-specific error correction in llms. *arXiv preprint arXiv:2606.24964*, 2026.

Alex Ferrando, Xavier Suau, Jordi Gonzàlez, and Pau Rodriguez. Dynamically scaled activation steering. *arXiv preprint arXiv:2512.03661*, 2026.

Stefan Heimersheim and Jake Mendel. [interim research report] activation plateaus & sensitive directions in GPT2. AI Alignment Forum, July 2024. URL https://www.alignmentforum.org/posts/LajDyGyiyX8DNNsuF/interim-research-report-activation-plateaus-and-sensitive-1.

Stefan Heimersheim and Alex Turner. Residual stream norms grow exponentially over the forward pass. AI Alignment Forum, May 2023. URL https://www.alignmentforum.org/posts/8mizBCm3dyc432nK8/residual-stream-norms-grow-exponentially-over-the-forward.

John J Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558, 1982.

Jett Janiak, Jacek Karwowski, Chatrik Singh Mangat, Giorgi Giglemiani, Nora Petrova, and Stefan Heimersheim. Characterizing stable regions in the residual stream of LLMs. In *NeurIPS 2024 Workshop on Scientific Methods for Understanding Deep Learning*, 2024. URL https://openreview.net/forum?id=IWiXfSHdxm.

Mikail Khona and Ila R Fiete. Attractor and integrator networks in the brain. *Nature Reviews Neuroscience*, 23(12):744–766, 2022.

Llama Team, AI @ Meta. The Llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Andrew Mack and Alex Turner. Mechanistically eliciting latent behaviors in language models. *AI Alignment Forum*, 2024. https://www.alignmentforum.org/posts/ioPnHKFyy4Cw2Gr2x/mechanistically-eliciting-latent-behaviors-in-language-1.

Samuel Marks and Max Tegmark. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. *arXiv preprint arXiv:2310.06824*, 2023.

T. McGeer. Passive walking with knees. In *Proceedings., IEEE International Conference on Robotics and Automation*, pages 1640–1645 vol.3, 1990. doi: 10.1109/ROBOT.1990.126245.

Thomas McGrath, Matthew Rahtz, Janos Kramar, Vladimir Mikulik, and Shane Legg. The hydra effect: Emergent self-repair in language model computations. *arXiv preprint arXiv:2307.15771*, 2023.

Narmeen Oozeer, Shivam Raval, Philip Quirke, Manikandan Ravikiran, Jeff Phillips, Shriyash Upadhyay, and Amirali Abdullah. Riemannian-manifold steering: Geometry-aware generative autoencoders for label-free steering, 2026. URL https://arxiv.org/abs/2605.24942.

Nina Panickssery, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Matt Turner. Steering llama 2 via contrastive activation addition, 2024. URL https://arxiv.org/abs/2312.06681.

Chandana Paul. Morphological computation: A basis for the analysis of morphology and control requirements. *Robotics and Autonomous Systems*, 54(8):619–630, 2006. ISSN 0921-8890. doi: https://doi.org/10.1016/j.robot.2006.03.003. URL https://www.sciencedirect.com/science/article/pii/S0921889006000613. Morphology, Control and Passive Dynamics.

Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. The fineweb datasets: Decanting the web for the finest text data at scale. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024. URL https://openreview.net/forum?id=n6SCkn2QaG.

Qwen Team. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Cody Rushing and Neel Nanda. Explorations of self-repair in language models. *arXiv preprint arXiv:2402.15390*, 2024.

Matthew Shinkle and Stefan Heimersheim. Activation plateaus: Where and how they emerge. LessWrong, October 2025. URL https://www.lesswrong.com/posts/WMfSbt7AAcJdHzysB/activation-plateaus-where-and-how-they-emerge.

Nishant Subramani, Nivedita Suresh, and Matthew Peters. Extracting latent steering vectors from pretrained language models. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Findings of the Association for Computational Linguistics: ACL 2022*, pages 566–581, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-acl.48. URL https://aclanthology.org/2022.findings-acl.48/.

David Sussillo and Omri Barak. Opening the black box: low-dimensional dynamics in high-dimensional recurrent neural networks. *Neural computation*, 25(3):626–649, 2013.

Team Olmo, Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, Hamish Ivison, Jacob Morrison, Jake Poznanski, Kyle Lo, Luca Soldaini, Matt Jordan, Mayee Chen, Michael Noukhovitch, Nathan Lambert, Pete Walsh, Pradeep Dasigi, Robert Berry, Saumya Malik, Saurabh Shah, Scott Geng, Shane Arora, Shashank Gupta, Taira Anderson, Teng Xiao, Tyler Murray, Tyler Romero, Victoria Graf, Akari Asai, Akshita Bhagia, Alexander Wettig, Alisa Liu, Aman Rangapur, Chloe Anastasiades, Costa Huang, Dustin Schwenk, Harsh Trivedi, Ian Magnusson, Jaron Lochner, Jiacheng Liu, Lester James V. Miranda, Maarten Sap, Malia Morgan, Michael Schmitz, Michal Guerquin, Michael Wilson, Regan Huff, Ronan Le Bras, Rui Xin, Rulin Shao, Sam Skjonsberg, Shannon Zejiang Shen, Shuyue Stella Li, Tucker Wilde, Valentina Pyatkin, Will Merrill, Yapei Chang, Yuling Gu, Zhiyuan Zeng, Ashish Sabharwal, Luke Zettlemoyer, Pang Wei Koh, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. Olmo 3, 2025. URL https://arxiv.org/abs/2512.13961.

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

Sihan Wang, Jiayi Zhao, Qingyan Cao, Hongbo Yao, and Lin Shu. Fishback: Pullback fisher geometry for optimal activation steering in transformers, 2026. URL https://arxiv.org/abs/2605.17231.

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

Chris Wendler, Veniamin Veselovsky, Giovanni Monea, and Robert West. Do llamas work in english? on the latent language of multilingual transformers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15366–15394, 2024.

Daniel Wurgaft, Can Rager, Matthew Kowal, Vasudev Shyam, Sheridan Feucht, Usha Bhalla, Tal Haklay, Eric Bigelow, Raphael Sarfati, Thomas McGrath, et al. Manifold steering reveals the shared geometry of neural network representation and behavior. *arXiv preprint arXiv:2605.05115*, 2026.

Jacob A Zavatone-Veth, Sheng Yang, Julian Alex Rubinfien, and Cengiz Pehlevan. How does training shape the riemannian geometry of neural network representations? In Francisco Acosta, Simone Azeglio, Bahareh Tolooshams, Chase van de Geijn, Christian Shewmake, Sophia Sanborn, and Nina Miolane, editors, *Proceedings of the 4th (2025) and 3rd (2024) NeurIPS Workshops on Symmetry and Geometry in Neural Representations*, volume 282 of *Proceedings of Machine Learning Research*, pages 792–883. PMLR, 14 Dec 2024–07 Dec 2025 2026. URL `https://proceedings.mlr.press/v282/zavatone-veth26a.html`.

A Norm-growth profiles with original context

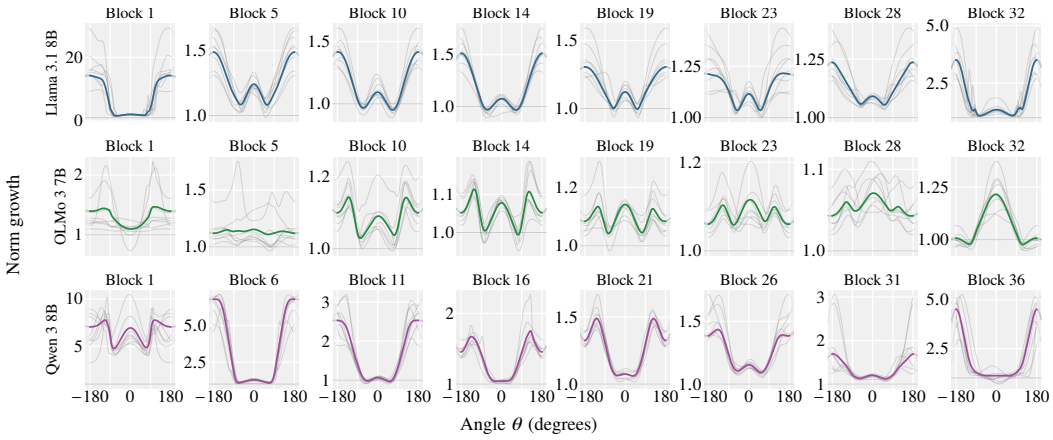

Figure 13: Norm growth profiles with non-ablated context.

B Velocity barriers in additional models

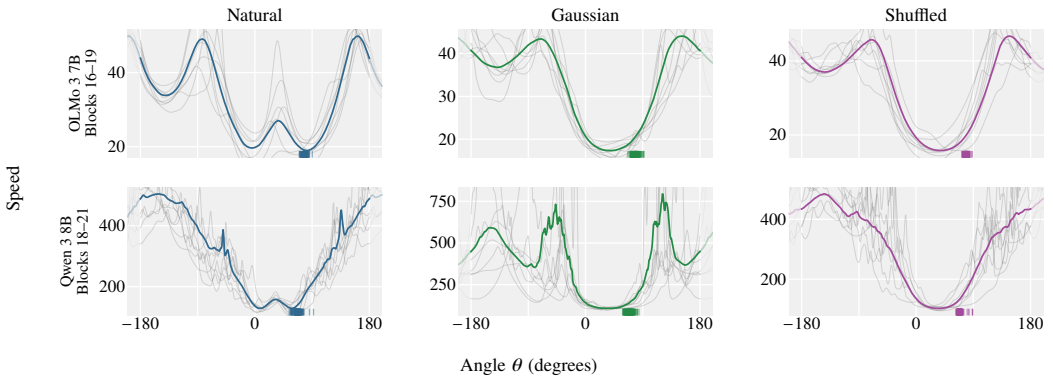

Figure 14: Barriers between natural activations in OLMo 3 7B and Qwen 3 8B, complementing the Llama results in fig. 7. Columns compare natural, Gaussian, and coordinate-shuffled pairs. Colored curves show means over 128 pairs; gray curves show individual examples.

C Speed at activation does not distinguish natural and synthetic activations

Figure 12 shows the velocity profiles for single layer transformer blocks. There is no significant difference between natural and synthetic activations in terms of velocity profile for just a single layer.

D Details for 2D sphere images

For single activations $\boldsymbol{x}$, we choose two random tangents at $\boldsymbol{x}$ to define the sphere. We project the surface onto the plane spanned by $f(\boldsymbol{x})$ and $\dot{\eta}(0)$, where $\eta(\theta)$ parameterizes the 1D sphere along one of the tangents. For pairs of activations $\boldsymbol{x}, \boldsymbol{x}'$, we scale $\boldsymbol{x}'$ to have the same norm as $\boldsymbol{x}$, then define the sphere with one tangent $\boldsymbol{u}$ at $\boldsymbol{x}$ toward the second activation $\boldsymbol{x}'$, i.e., $\boldsymbol{u} = (\boldsymbol{x} - \boldsymbol{x}') \perp \boldsymbol{x}$, and choose a random second tangent. For the projection parameterize a 1D sphere probe with $\eta(0) = f(\boldsymbol{x})$ and $\eta(\phi) = f(\boldsymbol{x}')$, and project the transformed 2D sphere onto the plane spanned by $f(\boldsymbol{x}) + f(\boldsymbol{x}')$ and $\dot{\eta}(0) + \dot{\eta}(\phi)$.

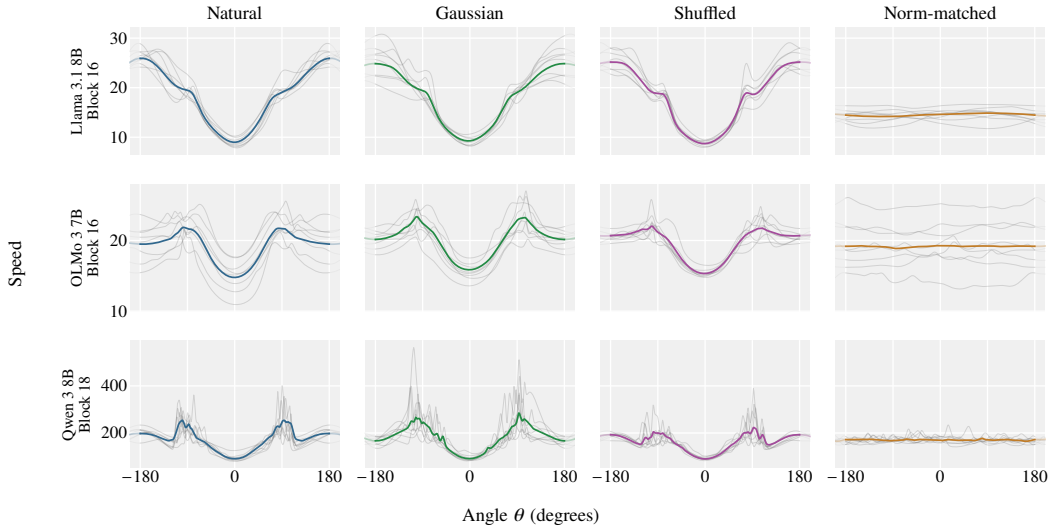

Figure 15: Comparing natural to synthetic activations' speed profiles at for various models.

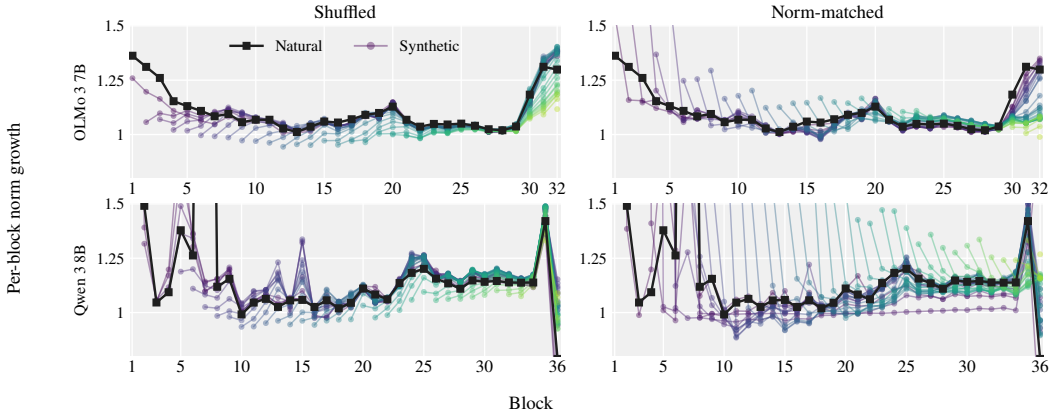

Figure 16: Norm growth trajectories for synthetic activations in OLMo and Qwen.

Class	Layer / Model	1	4	6	8	11	14	16	18	21	24	26	28	31	34	36
Gaussian	Llama 3.1 8B	0.851	0.963	0.915	0.897	0.789	0.818	0.963	0.983	0.981	0.899	0.869	0.759	0.555	—	—
	OLMo 3 7B	0.618	0.808	0.887	0.890	0.893	0.933	0.947	0.954	0.997	0.973	0.935	0.928	0.742	—	—
	Qwen 3 8B	0.765	0.621	0.824	0.865	0.923	0.833	0.690	0.740	0.830	0.923	0.965	0.955	0.836	0.669	0.478
Shuffled	Llama 3.1 8B	0.927	0.999	0.991	0.956	0.980	0.999	1.000	0.998	1.000	0.980	0.950	0.928	0.835	—	—
	OLMo 3 7B	0.525	0.887	0.954	0.913	0.927	0.956	0.974	0.986	0.990	0.975	0.903	0.902	0.866	—	—
	Qwen 3 8B	0.997	0.906	0.969	0.908	0.943	0.953	0.871	0.957	0.995	0.998	0.995	0.981	0.944	0.786	0.503

Table 2: For selected layers, we measure ROC-AUC of a simple, norm growth threshold classifier for whether an activation is natural or synthetic. Most layers of most models score high, meaning that the 128 natural and synthetic activations in our test set are almost perfectly separable.

Synthetic class	Llama 3.1 8B	OLMo 3 7B	Qwen 3 8B
Gaussian	0.963	0.984	0.938
Shuffled	0.980	0.996	0.986

Table 3: Velocity barrier–based classification of synthetic and natural activations. All models and synthetic activation types show high separation.

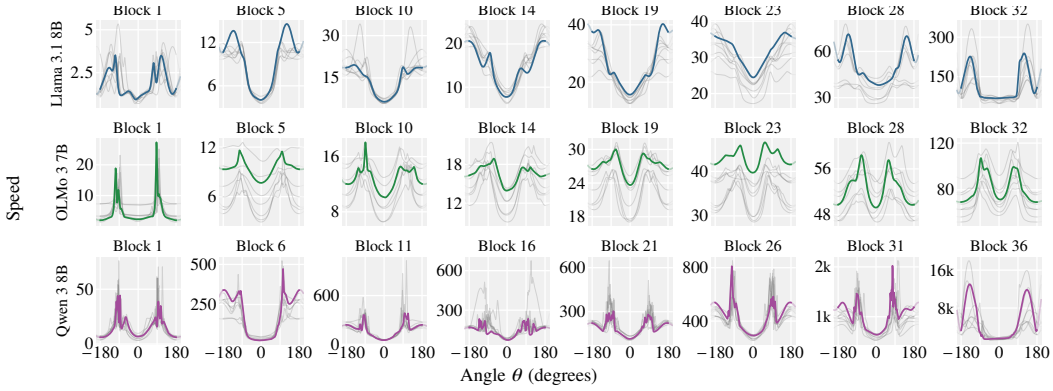

Figure 17: **Per-layer speed around natural activations.** Vertical axis gives speed, i.e., $\|\dot{\eta}(\theta)\|_2$.