SKY SPHERE REPRESENTATION IN LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

We analyze whether language models have an internal representation of the night sky that they use in answering questions about proximity on the sky. We find such a representation (a "sky sphere") in residual streams of tested LLMs (size 32B-235B parameters) on prompts like "in the night sky [object X] appears near". In 6 out of 7 models the sky sphere is linearly decodable (R^2 score of 73–91% and median angular error down to 11°–18° in leave-one-out tests). The sky sphere is indeed spherical and not a flat atlas chart, and spectral embeddings of graphs derived from constellation grouping and adjacency or from corpus co-occurrence failed to fit the observed geometry as well as the straight sphere. However the neural sky sphere is bent: including deg-2 spherical harmonics in the model was crucial for the following intervention experiments that tested causal use of the sky sphere. Ablating the 8 directions utilized by the bent sky sphere raises the angular distance to a neighbor provided by LLM's output from 17°–21° to 72°–86° in most models (for control, ablating the complementary $\mathbb{R}^8$ to the sky sphere in top 16 PCA raises the separation by less than 3°). In steering, when asked for a neighbor of constellation A, replacing "sky" directions with the model fit for a distant constellation B shifts its prediction (probability 62%–85% near A) to the target (58%–80% near B, 0%–2% still near A), which shows active use of the sky sphere. To our knowledge, this is the first instance of observing a feature manifold where the competing representation models that are reducible to linear combinations of one-dimensional curves are ruled out with high confidence. Code is released.

1 INTRODUCTION

Mechanistic interpretability tools can be roughly divided into two groups: linear methods, such as linear probes (Alain & Bengio, 2016; Belinkov, 2022) and sparse autoencoders (SAEs) (Bricken et al., 2023; Cunningham et al., 2024), and methods that analyze activations with deep learning systems, such as natural language autoencoders (NLAs) (Fraser-Taliente et al., 2026) and activation oracles (Karvonen et al., 2025). The downside of the second group is opaqueness. The downside of the first is that linear methods miss important non-linear structures present in the representation space.

Of particular interest, therefore, are *feature manifolds*: low-dimensional submanifolds of the activation space that correspond to a specific concept. Feature manifolds offer a middle ground between the two groups of tools: they capture non-linear structure while remaining explicit and human-inspectable. For example, when asked "what day is 5 days after Friday?", LLMs represent the days of the week (and, likewise, the months of the year) on a circle and solve this modular arithmetic problem by rotating it (Engels et al., 2025). Similarly, when deciding where to break a line in fixed-width text, a model tracks the running character count on a curved one-dimensional manifold, which attention heads then twist to compare against the line width (Gurnee et al., 2026a). Both are examples of one-dimensional feature manifolds.

In this paper, we demonstrate the existence of a spherical feature manifold in the activation space of LLMs, corresponding to the positions of stars and constellations in the night sky (Figure 1). Our prompts mention the objects only by their proper names and avoid mentioning any coordinates explicitly, but rather ask what is close to the object in the sky.

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.

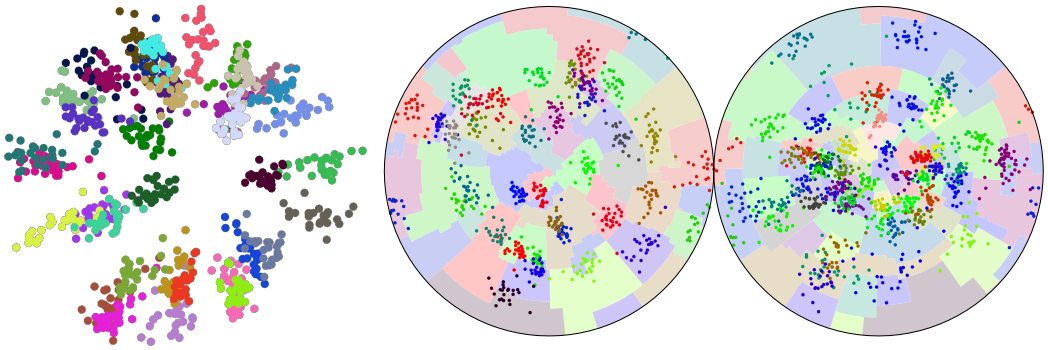

Figure 1: Sky object representation in the residual stream of Mistral Large 2 LLM, layer 66, each object passed in varying prompts produces a uni-color point cloud. *Left*: Selected 38 night sky object OLS-projected to a "sky globe" $\mathbb{R}^3$. *Right*: Constellations OLS-projected onto a stereographic constellation chart.

Thus, a spherical arrangement in activation space is likely not a trivial echo of coordinate strings seen in training, but rather a reflection of the notion of proximity for the night sky objects. Across seven open-source models of 32B–235B parameters, we find that the position of an object on the celestial sphere is linearly decodable from the top principal components of the residual stream on proximity focused prompts. In most models, leave-one-out (LOO) validation (on prompt-averaged vectors) yields R^2 scores of 73–91% and median angular errors down to $11°$–$18°$, and the representation rises into the top $\sim$8 principal components.

Previously discovered geometric representations in language models have mostly been either flat (continents and USA states (Gurnee & Tegmark, 2024), color space (Abdou et al., 2021)) or one-dimensional (months and historical years (Engels et al., 2025), (Karkada et al., 2026), (Modell et al., 2025), sentence length (Gurnee et al., 2026a)). The only other example of a non-flat irreducible feature manifold (meaning, is not a direct sum or mixture of lower-dimensional features, as defined by Engels et al. (2025)) that we found is in Tiblias et al. (2025), which compared different hypothetical geometries for neural representation of cities and found the spherical one to be a better fit than flat or cylindrical.

2 Methodology and Data

2.1 General pipeline

We feed various prompts (from a fixed list) about night sky objects into an LLM in auto-complete mode (no thinking), and extract the top 128 principal components of the distribution of the residual stream vectors in every layer. We find a linear regression fit (using the top $n = 4, 8$ or 16 PC-components) to the true coordinates of the objects on the unit sphere of the night sky. For validation on object X it is left out of the data for the regression fit, and its residual stream is averaged over prompts with object X. We then record the accuracy of how well it predicts the unseen object X across the whole dataset: the R^2 score (variance explained by the regression model) and the median angular error. We also record the PCA components that carry the best fit. We also use a 80/20% train/test split rather than leaving one object out. In all models, median R^2 scores on 11 random splits confirmed the sphere model signal with empirical p below 10^{-5}. See Appendix D for details.

2.2 Input Data

We use 188 astronomical objects of 3 types: 85 brightest stars, all 88 official constellations and 15 arbitrary galaxies, nebulae and clusters (see Appendix B). The coordinates of the constellations are defined as their area centroids (projected back on the unit sphere). We feed these objects into LLMs wrapped in 25 prompts, like *the closest constellation to [X] is*, or *stargazers spot [X] right beside*, where [X] is replaced by a given object, so the whole dataset is of size $25 \times 188 = 4700$. The

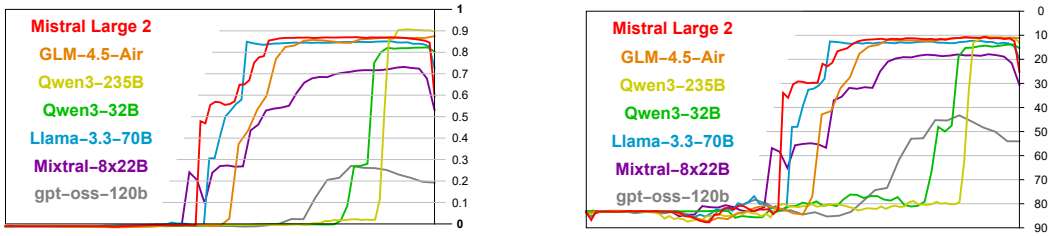

Figure 2: Decodability of sky representation across normalized layer depth. Leave-one-out analysis: predicting the coordinates on the night sky of an object from residual stream (top-8 PCA) using linear regression on stream data from other objects. *Left*: R^2-score. *Right*: Median angular error

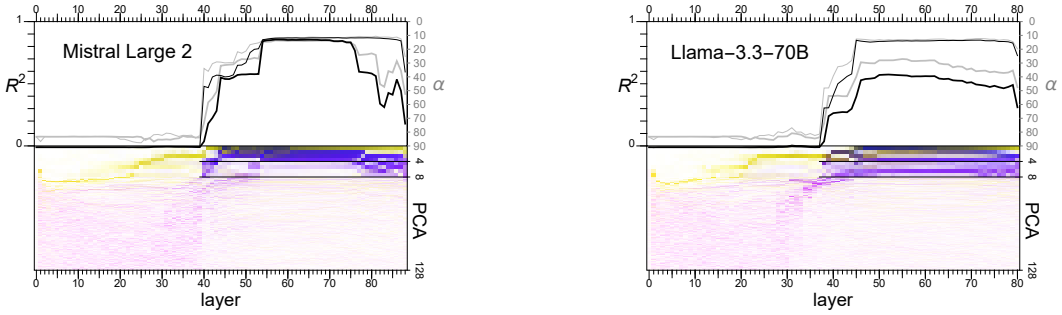

Figure 3: R^2-score (black) and median angular error (gray) in LOO validation for sky fitting when using top-8 (thin) and top-4 (thick) PCA components. Colored charts show which PC's correlate with the optimal projection (blue&purple) and with the object type (yellow)

prompts focus on the apparent proximity on the night sky but avoid explicit mentions of coordinates in any way. For the whole prompt set and more details see Appendix C.

2.3 Models

We used the following 7 open-source models in our investigation (their details are listed in Table 1): Qwen3-32B, Qwen3-235B, gpt-oss-120b, Llama-3.3, Mixtral-8x22B, Mistral Large 2, GLM-4.5-Air.

3 Results

3.1 Evidence of the representation

Figure 2 shows significant decodability of the position on the celestial sphere from residual stream in most tested models. Note that the poor performance of gpt-oss in this test does not reflect poorly on its actual abilities: when asked directly for stellar coordinates, it locates over 70% of objects to within 1°, similar to other models with thinking in our set, see Appendix D.4.

3.2 Representation prominence in PCA

Figure 3 shows that the OLS embedding of the celestial sphere is weak and spread along many PCA components in early layers (light purple), but later strengthens and rises to top∼8 (and even top∼4) PCA components (dark blue). In many models it is only surpassed by the direction that separates stars from constellations (yellow) and leaves the sky dome mostly in PCs 2-4 (see Appendix H). We note that the prominence and clarity of this signal relies on our prompts being focused on night sky proximity. On more generic prompts, the signal weakens and sinks to lower PCA ranks, see Appendix D.

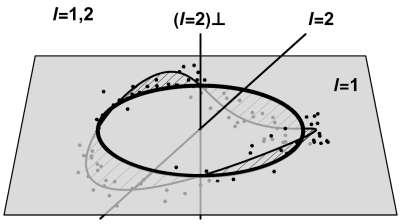

Figure 4: Modes directions for ablation and steering. Linear modes (first harmonics $l = 1$) fit data with a straight sphere, adding quadrupole moments (second harmonics $l = 2$) captures simplest bends on top of that. By making $(l = 2)$ space orthogonal to $(l = 1)$ we get $(l = 2)\ \perp$, whose ablation wouldn't affect the straight sphere $(l = 1)$

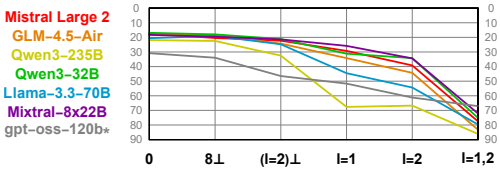

Figure 5: Ablation results. Median angular separation of the object called by a prompt and the neighbor named by an LLM. Ablating either the straight sphere model (first harmonics, $l = 1$), the quadrupole moments in the complement (second harmonics, $l = 2$), both, or neither, or the 8-dim complement of both in top-16 PCA

3.3 Ablation and Steering

To test the causal use of the sky sphere, we performed ablation and steering experiments. In ablation tests, we see that projecting just along the $\mathbb{R}^3$ spanned by a linearly fitted spherical model (using the first spherical harmonics, $l = 1$) hurts LLMs' abilities to name a proximal star or constellation rather moderately (on average). To make the error shift close to the $90°$ that a random chance would give, one needs to ablate the next harmonics $l = 2$ too, showing that the sky sphere is bent in neural space. As a control, instead of ablating the $\mathbb{R}^8$ spanned by the $l = 1, 2$ modes, we ablate the $\mathbb{R}^8$ orthogonal complement within the top-16 PCs and see almost no change in performance. Further details are in Appendix E.1.

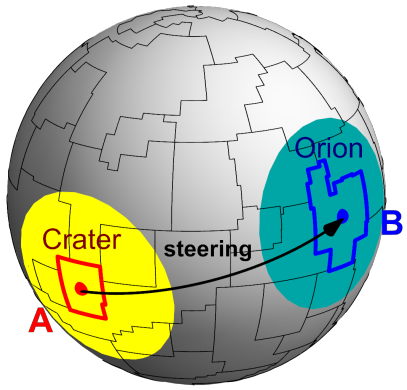

Figure 6: Residual stream steering. Fit the sky sphere into the residual stream; feed LLMs questions about neighbors of constellation A; replace residual stream components in sky model subspace with a different constellation B fit. Steering is successful if the neighbors provided by LLMs switch from vicinity of A to vicinity of B (or B itself)

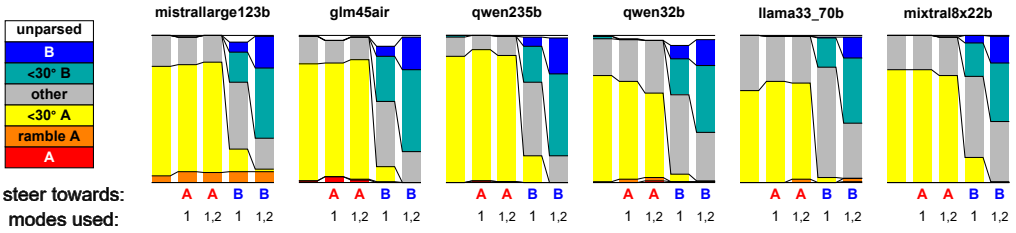

Figure 7: Steering experiment. No intervention (leftmost column) vs steering towards the original constellation A or a new target B by using either straight sphere model ($l = 1$), or accounting for second spherical harmonics ($l = 1, 2$). Steering towards the original constellation A doesn't change behavior (naming constellations near A), steering towards B switches to naming neighbors of B, more precise if higher harmonics are used.

For steering we use either the straight $l = 1$ or the bent $l = 1, 2$ fit of the sky sphere into residual stream, as in ablation above. We use prompts asking for a neighbor of constellation A. We may intervene by replacing residual stream vector components (spanned by the model fit) with a model representation of a different constellation B that is picked to be as orthogonal on the sky to A as possible (Figure 6). The results in Figure 7 show that without intervention, LLMs mostly provide a neighbor within 30° of A. However, if we change the vector to the prediction of the bent sphere model (using $l = 1, 2$ modes), LLMs' answers change mostly to either the target B or its neighbor, with neighbors of A practically eliminated. If we use the straight model instead ($l = 1$ modes), the effect is less accurate: LLMs are still steered away from A, but they land near the target B half as often. As a measure of control, if we use either modes to steer towards A itself, the performance remains similar to the tests with no intervention (leftmost columns). Further details are in Appendix E.2.

In conclusion, both ablation and steering experiments show that changes within the sky sphere space affect LLMs' output in great accordance with our interpretation, but only if our model allows the sphere to bend in neural space by higher spherical harmonics (adding just the second ones already shows strong results).

3.4 Competing representations

The unit sphere representation of the night sky is not the only possible one. In the work of Gurnee & Tegmark (2024), the model used for the continents and USA states was not globe-based, but a flat map with latitude and longitude as coordinates. Similarly, the sky may be represented as a flat 2D-chart instead of as a unit sphere. Tiblias et al. (2025) compared spherical, flat and cylindrical representations of cities. Separating these models is not entirely straightforward, since the North-South dimension is highly correlated between them. Thus even a completely absent model would have a significant correlation with data by proxy. This likely explains why in Tiblias et al. (2025) flat and cylindrical charts scored not much lower than the sphere (83–92% of sphere's score). To avoid this issue, we use a modified RDM-analysis by measuring how much the metric on test objects (that

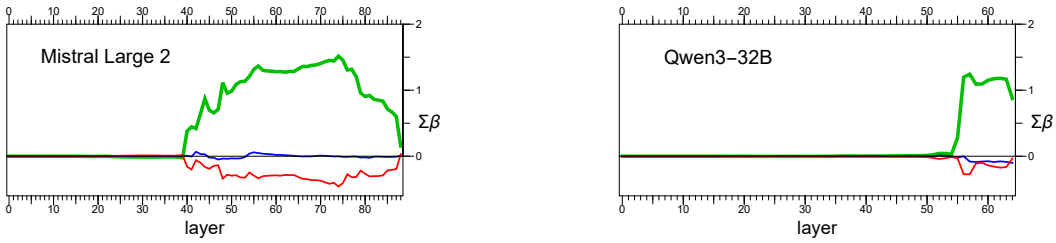

Figure 8: Prominence of the sphere (green thick), cylinder (red) and flat (blue) representations in the residual stream data, as assessed by their more distinctive components (see Appendix F.1 for details and other models).

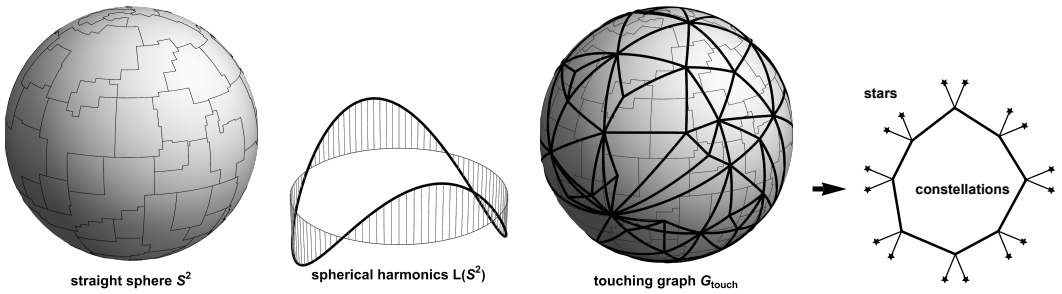

Figure 9: Some models compared with the neural representation. Straight sphere S^2, sphere with added higher harmonics $L(S^2)$ and the graph G_{touch} that joins constellations that touch each other and attaches other objects to the constellations they belong to.

we observe in the residual stream) is attributable to each model dimension separately, and in scoring ignore the confounded North-South dimension (see Appendix F.1). The results show that only the sphere has a consistently positive score (see, for instance, Figure 8).

Another alternative representation conceivable for the stars is a genuine 3D-representation. However, since distances to the stars alone span ~ 3 orders of magnitude (in our set), the radial component, if present, would have to be non-linear. We analyze the radial component, but don't find a significant (or even consistent across models) correlation with the distance to the star (Appendix F.4). This rules out our representation being secretly a 3D Euclidean one (even with adjusted, but still clear radial dependence). We find, however, that the radial component *is* influenced by other factors, most notably, the type of the object (constellations are placed further, stars are placed closer), and, to a lesser extent, by its frequency in texts and, possibly, brightness (Figure 21).

Lastly, we test our representation against variously derived spectral embeddings (embeddings by modes, that is, eigenfunctions of the Laplacian). Among them are embedding of the true sky sphere by its harmonics (straight sphere is a particular case where the modes decay to 0 past the linear $l = 1$ modes) and spectral embeddings of graphs that describe constellation adjacency: constellations are joined by an edge if they touch, other objects are joined to constellations that host them with edges of a different length. These are illustrated in Figure 9.

We also test corpus derived embeddings. A naive one G_{text} builds a graph where objects are joined iff they occur within 10 word window in a corpus. Karkada et al. (2026) proposed a better, more theoretically motivated way to construct embeddings from co-occurrence in training corpus (details in Appendix F.3), we denote this model Q_{text} and include it in our comparison.

For all embeddings we compute the induced distances between data points and correlate this metric with the one observed in the residual stream (see details in Appendix F.2). Figure 10 shows that pure sphere S^2 and a sphere with added harmonics $L(S^2)$ fit data better than any considered model. This result leaves open the question of how exactly LLMs learn this representation and what structures and statistics they use for that purpose.

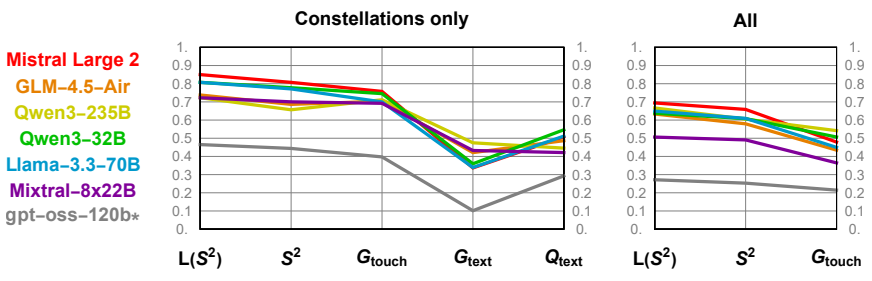

Figure 10: Pearson correlation of (metrics induced by the) observed representation in the residual stream with various toy and theoretical models (see text or Table 8). Straight sphere S^2 and the sphere with added harmonics $L(S^2)$ are on top.

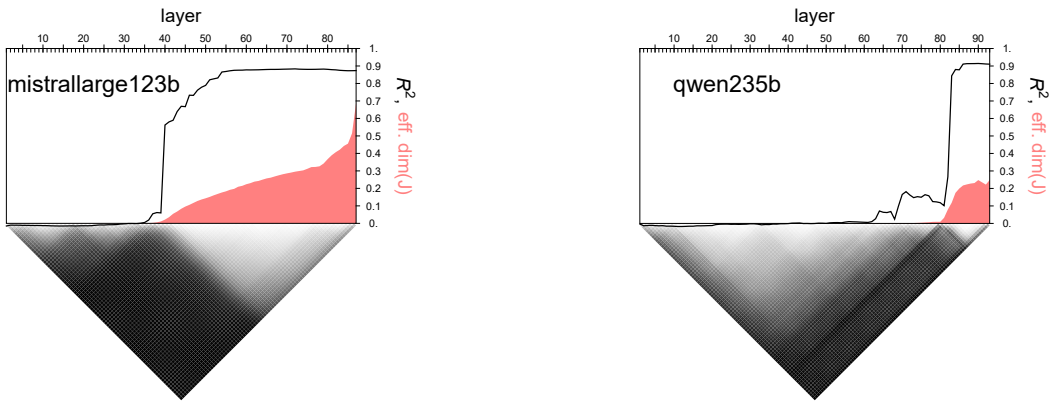

Figure 11: Comparison of R^2-signal of sky in top-16 PCA (black line) and effective dimension of J_l-lens (red shade, relative to model dim). On the bottom, (a half of) the similarity matrix of J-lenses between different layers. Workspace is visible as the rightmost block.

4 Connection to the J-space

Recently Gurnee et al. (2026b) reported on the discovery of a structure within LLMs that seems to hold intermediate concepts the model uses within a given pass, but does not output at the end. It was captured via a J-lens, that is the Jacobian of the forward pass from a given layer l to the last one (averaged over prompts and positions):

$$J_l := \mathbb{E}(\frac{\partial h_{\text{final}}}{\partial h_l})$$

where h_x is the residual stream vector of layer x. Comparing these J-matrices across the layers, Gurnee et al. (2026b) observed that the layers split into blocks: initial "sensory" half that prepares the data for processing, second "workspace" half, where higher-order processing is performed, and the last few "motor" layers that extract the answer logits from the output.

We checked how our representation relates to the J-space of tested LLMs. Remarkably, the layers with a strong sky signal seem to occupy a similar layer span to the J-space: in most LLMs both span roughly the second half, but in Qwen models both are confined to the very end (compare charts on Figure 11). More technical details and caveats are in Appendix G.

5 Open questions

We only analyzed the representation in the residual stream, which leaves several aspects unclear. Is the mechanism that constructs the representation more interpretable than pure memorization? Given that the models know celestial coordinates far better than the accuracy of our representation (see Figure 13), what is its role for the models? Gurnee et al. (2026a) identified an attention head that manipulates a one-dimensional feature manifold to determine where to add a line break. Can we describe how the sky sphere is used mechanistically? Given our failure to tie the sky representation to textual co-occurrence as in Karkada et al. (2026), what mechanism does it arise from?

These questions may be related to the emergence of feature manifolds like the celestial sphere during training, for instance on model suites with released intermediate checkpoints (Biderman et al., 2023; Groeneveld et al., 2024). How does the prominence of the feature manifold correlate with the model's ability to accurately answer questions about positions of stars? What other important capabilities emerge at the same time?

We would also like to understand what determines the range of layers where the sky sphere lives. This range co-varies with the layer span of the J-space across model families (see Section 4 and Figure 11), and of particular interest is whether this points to a certain layer window responsible for particular capabilities in LLMs.

Finally, an important open problem for the field is to find unsupervised ways of discovering feature manifolds in LLMs.

AI USE STATEMENT

In this work, we used generative AI tools for generating synthetic data sets (for doubling prompt template set size), implementing methods (all coding and executing was done by an AI agent), designing and providing feedback on research methodology or experiments (while most metrics and tests were specified by authors, occasionally it was provided/adjusted by the coding agent; crucially, the methodology of section F.1 was devised in a back and forth with AI, the effective dimension metric (Equation 5) was proposed by AI alone), cleaning and reformatting datasets (retrieval of night sky object lists, their coordinates, alternative names). We used different agents to reproduce code and verify experiments.

We have not used generative AI tools for developing theoretical models or conceptual frameworks, proposing or refining hypotheses, supporting qualitative and thematic data analysis, interpreting results (the last two were frequently done unprompted by the coding agent, but not used). Also AI tools weren't used for formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs and assisting with translation, as these are not relevant to this paper.

Additionally, we used generative AI tools for condensing the abstract and finding a few more relevant prior work publications. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The code used to gather and analyze the data is available at github.com/AnonymusPurposes/Decodable-sky

REFERENCES

Mostafa Abdou, Artur Kulmizev, Daniel Hershcovich, Stella Frank, Ellie Pavlick, and Anders Søgaard. Can language models encode perceptual structure without grounding? a case study in color. In *Proceedings of the 25th conference on computational natural language learning*, pp. 109–132, 2021.

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

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

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. Pythia: A suite for analyzing large language models across training and scaling. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pp. 2397–2430. PMLR, 2023.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, et al. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. URL https://transformer-circuits.pub/2023/monosemantic-features.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2309.08600.

Joshua Engels, Isaac Liao, Eric J Michaud, Wes Gurnee, and Max Tegmark. Not all language model features are linear. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2405.14860.

Kit Fraser-Taliente, Subhash Kantamneni, Euan Ong, Dan Mossing, Christina Lu, Paul C Bogdan, et al. Natural language autoencoders produce unsupervised explanations of LLM activations. *Transformer Circuits Thread*, 2026. URL https://transformer-circuits.pub/2026/nla/index.html.

GLM-4.5 Team. Glm-4.5: Agentic, reasoning, and coding (arc) foundation models, 2025. URL https://arxiv.org/abs/2508.06471.

Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, et al. OLMo: Accelerating the science of language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15789–15809, 2024.

Wes Gurnee and Max Tegmark. Language models represent space and time. In *International Conference on Learning Representations*, volume 2024, pp. 2483–2503, 2024.

Wes Gurnee, Emmanuel Ameisen, Isaac Kauvar, Julius Tarng, Adam Pearce, Chris Olah, and Joshua Batson. When models manipulate manifolds: The geometry of a counting task. *arXiv preprint arXiv:2601.04480*, 2026a.

Wes Gurnee, Nicholas Sofroniew, Adam Pearce, Mateusz Piotrowski, Isaac Kauvar, Runjin Chen, Anna Soligo, Paul Bogdan, Euan Ong, Rowan Wang, et al. Verbalizable representations form a global workspace in language models. *arXiv preprint arXiv:2607.15495*, 2026b.

Albert Q. Jiang et al. Mixtral of experts, 2024. URL https://arxiv.org/abs/2401.04088.

Dhruva Karkada, Daniel J Korchinski, Andres Nava, Matthieu Wyart, and Yasaman Bahri. Symmetry in language statistics shapes the geometry of model representations. *arXiv preprint arXiv:2602.15029*, 2026.

Adam Karvonen, James Chua, Clément Dumas, Kit Fraser-Taliente, Subhash Kantamneni, Julian Minder, Euan Ong, Arnab Sen Sharma, Daniel Wen, Owain Evans, and Samuel Marks. Activation oracles: Training and evaluating LLMs as general-purpose activation explainers. *arXiv preprint arXiv:2512.15674*, 2025.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International conference on machine learning*, pp. 3519–3529. PMLR, 2019.

Dominik Kundel. Openai harmony response format, 2025. URL https://developers.openai.com/cookbook/articles/openai-harmony.

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

Mistral AI. Mistral large 2 (instruct, 2411). https://huggingface.co/mistralai/Mistral-Large-Instruct-2411, 2024a. Blog: https://mistral.ai/news/mistral-large-2407/. Accessed 2026-07-16.

Mistral AI. Cheaper, better, faster, stronger: Mixtral 8x22b. https://mistral.ai/news/mixtral-8x22b/, 2024b. Model: https://huggingface.co/mistralai/Mixtral-8x22B-Instruct-v0.1. Accessed 2026-07-16.

Alexander Modell, Patrick Rubin-Delanchy, and Nick Whiteley. The origins of representation manifolds in large language models. *arXiv preprint arXiv:2505.18235*, 2025.

OpenAI. gpt-oss-120b & gpt-oss-20b model card, 2025. URL https://arxiv.org/abs/2508.10925.

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

Federico Tiblias, Irina Bigoulaeva, Jingcheng Niu, Simone Balloccu, and Iryna Gurevych. Hypothesis-driven feature manifold analysis in LLMs via supervised multi-dimensional scaling. *arXiv preprint arXiv:2510.01025*, 2025.

A MODELS

Table 1 lists the open-source models used for most of tests in this work. Weights were loaded with transformers via AutoModelForCausalLM.from_pretrained(repo, revision=<commit>). Each revision was the repository's main HEAD at access time (July 2026).

Table 1: Models used in this work

Model huggingface repo used	**Type**	**Size**	**Layers**
Qwen3-32B, (Qwen Team, 2025) Qwen/Qwen3-32B 9216db5	Dense	32B tot	64
Qwen3-235B, (Qwen Team, 2025) Qwen/Qwen3-235B-A22B-Thinking-2507-FP8 f07f63f	MoE	235B tot 22B act	94
gpt-oss-120b, (OpenAI, 2025) openai/gpt-oss-120b b5c939d	MoE	117B tot 5B act	36
Llama-3.3, (Llama Team, AI @ Meta, 2024) unsloth/Llama-3.3-70B-Instruct (unsloth mirror) 99cd0d2	Dense	70B tot	80
Mixtral-8x22B, (Mistral AI, 2024b), (Jiang et al., 2024), mistralai/Mixtral-8x22B-Instruct-v0.1 cc88a6c	MoE	141B tot 39B act	56
Mistral Large 2, (Mistral AI, 2024a) mistralai/Mistral-Large-Instruct-2411 ba78820	Dense	123B tot	88
GLM-4.5-Air, (GLM-4.5 Team, 2025) zai-org/GLM-4.5-Air a24ceef	MoE	106B tot 12B act	46

B OBJECTS

Our test set consists of 3 groups. The first contains the 85 stars of apparent magnitude 2.5 or brighter under their proper names, which, together with their coordinates, are taken from IAU WGSN catalog. The second group has 15 miscellaneous, arbitrarily chosen night sky objects like prominent galaxies, nebulae and clusters (see Table 2), whose coordinates are pulled via astropy's SkyCoord.from_name method.

Table 2: Miscellaneous objects included in this work

GALAXIES	**NEBULAE**	**CLUSTERS**
Andromeda Galaxy	Orion Nebula	Pleiades
Triangulum Galaxy	Crab Nebula	Hercules Cluster
Whirlpool Galaxy	Ring Nebula	Omega Centauri
Sombrero Galaxy	Lagoon Nebula	Beehive Cluster
Large Magellanic Cloud	Carina Nebula	
	Tarantula Nebula	

Lastly, we often include 88 official constellations. For purposes of our analysis we define and compute the "location" of constellations on the night sky as their area centroids: $\sim 2 \cdot 10^5$ points are sampled uniformly over the unit sphere, those that fall within the constellation (determined by astropy's get_constellation) are averaged, and the result is projected back onto the unit sphere. We have two arguments to excuse the arbitrary nature of this choice. The first is that successful models in our tests don't perform worse on constellations compared to objects with less subjective point representatives (this analysis is not included), so the data couldn't have been spoiled too much by our choice of designation. The second argument is that in our tests we got median angular error down to 10° at best, so this noise would likely hide any imperfections of our choice anyway, given that the typical radius of a constellation is 10°–15°, so any reasonable ways to pick a center would differ by a much smaller amount.

Table 3: Default list of prompt templates

in the night sky X is not far from	the closest constellation to X is
in the night sky X shares space with	in star atlases X lies close to
in the night sky X is a short hop from	in the night sky X appears near
in the sky charts X is right next to	in the night sky X is located near
in the night sky X is close beside	sky watchers locate X next to
astrophotographers find X by first locating	in sky atlases X is adjacent to
in the night sky X is right next to	the closest bright star to X is
the neighboring constellation to X is	stargazers locate X right beside
on the celestial sphere X is situated near	sky atlases place X close to
on sky maps X appears beside	sky atlases put X close to
on star maps X lies beside	sky charts show X beside
nearby in the sky to X is	on star charts X sits beside
sky maps place X near	

C PROMPTS

The prompts we used for most tests are listed in Table 3. The set of prompts was generated by a combination of composing, weeding out and adjusting by hand, and using generative AI to expand existing sets. Originally, the prompts were more diverse, but the better performance was shown if

- the prompt asked about the nearest vicinity (not more general geometric relations)
- it made clear that it talks about specifically the night, in some way.

Thus, only prompts of this type made the final list.

To show the significance of prompt phrasing being focused on night sky proximity, we run the same test on a more generic set of prompts listed in Table 4. The results (Appendix D.5) show that, in this case, the representation appears weaker, never touches the top-8 PCA components (despite residing there entirely under default prompts), and decays quicker.

Table 4: Generic prompts

the age of X is estimated to be	X is	the magnitude of X is
the discovery of X is due to	X emits	X is composed mainly of
in mythology, X represents	X is known for	the temperature of X is
X was first recognized by	X is studied by	the luminosity of X is
the name of X comes from	X is believed to	astronomers use X for
last time X was seen in	X was named after	X is important for
X was first observed by	X is classified as	X is an example of
X was cataloged by	X is described in	X is mentioned by
brightness of X is		

D CHARTS AND VALIDATION

Here we explain the primary analysis of this work. All 25×188 prompts are fed into the model in the auto-completion (no thinking) regime and for each layer we record the top 128 PCA components of the residual stream on the last token of the prompt.

D.1 PLOTS

The presence of the sky sphere representation is validated with the following leave-one-out (LOO) analysis. We restrict to the top-4 or top-8 PCA space (still using all prompts). The model prediction

for an object X is defined then via linear regression (with prompts including X being held out now) onto the "true" night sky globe space — $\mathbb{R}^3$, where all objects are placed on the unit sphere $S^2 \subset \mathbb{R}^3$ according to their true celestial coordinates (for constellations see Appendix B). The accuracy of that prediction (averaged over prompts for each object) is measured with the R^2-score (variance explained by the model) and median angular error α.

D.2 80/20 SPLIT TEST

Another variation test we use holds out random 20% of the objects in the dataset for validation and records r^2 score and median angular error similar to LOO pipeline described in Appendix D.1. That is, remaining 80% of the objects are used to build OLS-optimal projection of top-8 PCA space to model $\mathbb{R}^3 \supset S^2$, then the (prompt-averaged) vectors of the 20% test set is scored by this model projection. We run 11 splits for each model and record median results in Table 5.

Table 5: Model performance in 80/20 split test.

LLM	layer	Med. r^2	Med. ang. error	p-value
Qwen3-235B	88	0.899	10.6°	$< 10^{-5}$
GLM-4.5-Air	46	0.886	11.1°	$< 10^{-5}$
Mistral Large 2	72	0.876	11.3°	$< 10^{-5}$
Llama-3.3	72	0.833	12.9°	$< 10^{-5}$
Qwen3-32B	63	0.816	12.7°	$< 10^{-5}$
Mixtral-8x22B	52	0.729	17.9°	$< 10^{-5}$
gpt-oss-120b	29	0.239	44.6°	$< 10^{-5}$

The p-value was calculated empirically by randomly re-matching objects with stream vectors, running the same test and checking how often it will beat the r^2 score originally produced. With 10^5 permutations none of the models ever matched the original score, implying $p < 10^{-5}$.

D.3 DENSITY PCA CHARTS

Below the plots we show charts that illustrate the location of the sky sphere representation within PCA directions (blue and purple). Specifically, we center and whiten the sky data (so that their coordinates c_k are distributed with mean 0 and unit variance I_3), and record

$$s(l,i) = \sum_{1 \le k \le 3} \mathrm{Cov}^2(c_k, (wp)_{li})$$

where $(wp)_{li}$ is the i-th principal component p of l-th layer scaled by a weight w that either uniformly scales the layer so that the top variance is 1, or whitens the data so that every component has unit variance. The first version sets the red channel and the second sets the green via formula $1 - \sqrt{s(l,i)}$. The blue channel is defined like the green one (with whitened PCA), but instead of correlation with celestial coordinates c_k we correlate $(wp)_{li}$ to the "type" function that is 1 if the object X in the prompt is a star, -1 if X is a constellation, and 0 if X is neither. So the colors on the chart should be interpreted as in Table 6.

D.4 GPT-OSS ANOMALY

Figures 2 and 12 show that in gpt-oss-120b the sky sphere is much less prominent than in models of comparable size, and hides in lower PCA components. We note that it doesn't mean that "it is way worse at knowing celestial coordinates"; in fact, all models with thinking in our test managed to locate most non-constellation objects to within 1° (Figure 13) — way better than $10^\circ - 20^\circ$ error of our linear neural decoding.

This leaves many questions regarding the role of the representation we found in model's thinking, but also leaves its absence in gpt-oss unexplained. One way in which it differs from other models is that it was trained on the harmony response format (see Kundel (2025)), rather than raw text, so our raw prompts were out of distribution in that regard. However, when we wrapped prompts in

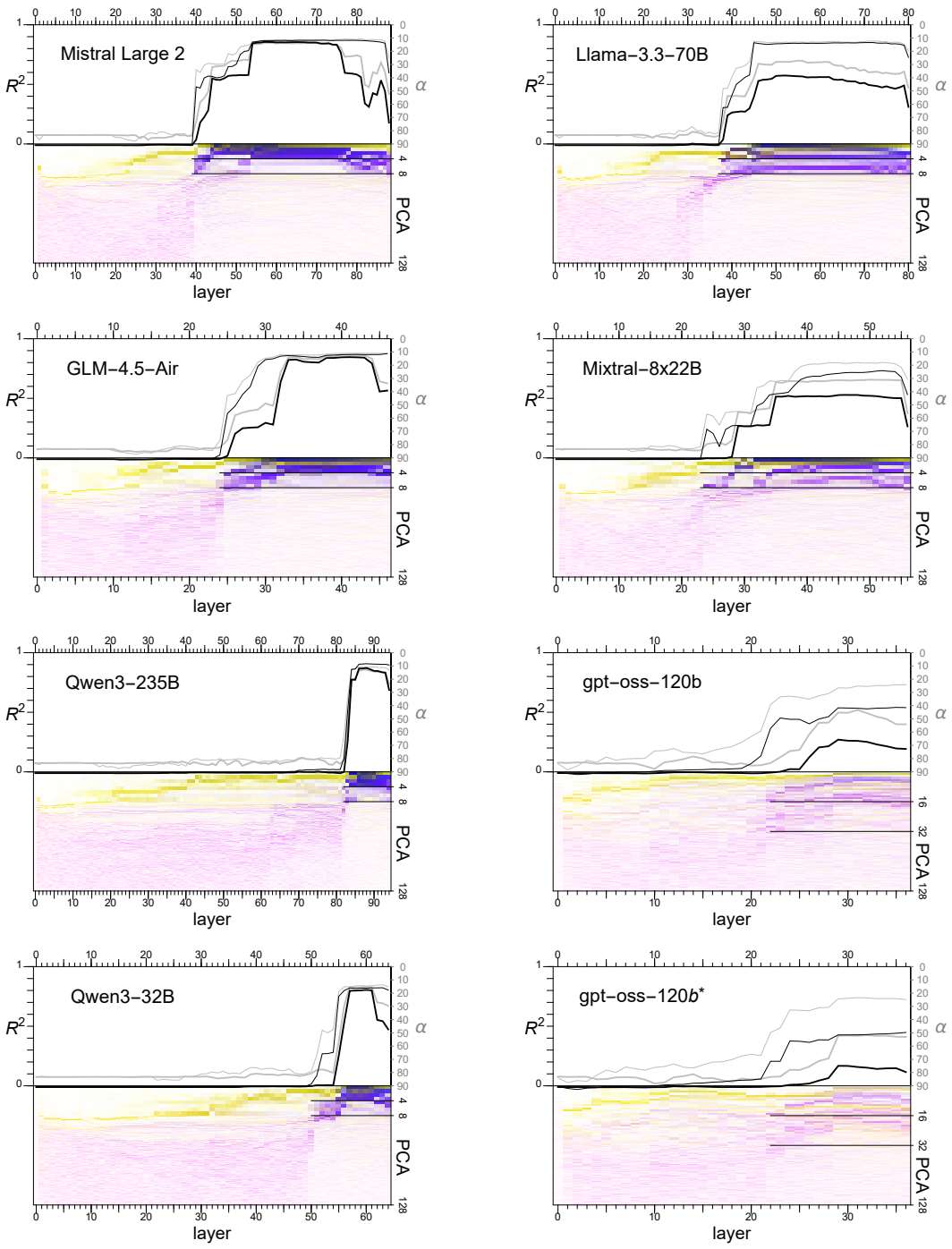

Figure 12: R^2-score (black) and median angular error (gray), in thin line for top-8 (top-32 for gpt), or in thick for top-4 (top-16 for gpt) PCA components for a sky sphere fit. Colored charts show which PC's correlate with the optimal projection (blue&purple) and with the object type (yellow) (see details in text Appendix D). The last gpt-oss-120b* chart for a special test for this outlier model, see Appendix D.4

the appropriate format, the decodability representation dropped even lower (Figure 14, same as last chart in Figure 12)

Table 6: Color key for Figure 12.

Color	Correlation with	PCA strength
Dark blue	Sky Sphere, Strong	Strong
Light blue	Sky Sphere, Weak	Strong
Dark purple	Sky Sphere, Strong	Weak
Light purple	Sky Sphere, Weak	Weak
Solid Yellow	Object Type, Strong	
Light yellow	Object Type, Weak	
White	Nothing	

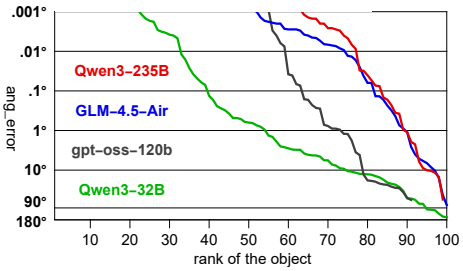

Figure 13: Accuracy on prompts "Give the right ascension and declination of X with no commentary.", log of angular error sorted in increasing order (results closer than .001° not shown).

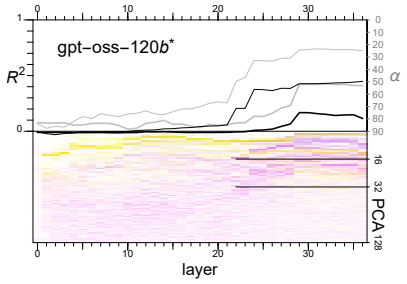

Figure 14: Chart from Figure 12 for gpt-oss on prompts with proper harmony response format.

D.5 Charts for proximity-focused vs generic prompts

Here we illustrate the effect that the phrasing of the prompts has on decodability of sky sphere representation. Note that the apples-to-apples comparison is to be done between the gray shade graph (sky proximity prompts) and thick black line (generic prompts), both being restricted to the same top-8 PCA space. In this case the signal for generic prompts stays essentially at 0. The R^2-score (defined as in Appendix D.1) only rises to the thin black line when the domain is expanded (thin line shows the results on top-32 PCA space), and even then it is far from the results on the focused prompts, and decays in later levels.

E Intervention

E.1 Ablation

We start with 128 PC components of the residual stream constructed in Appendix D, take the top 16 of them and fit in that space different models of the sky sphere. The first one, $S_1 : D \to \mathbb{R}^3$, is an ordinary sphere model, whose $\mathbb{R}^3$ coordinates x, y, z are also the first ($l = 1$) harmonic functions of the sphere. The second, $S_2 : D \to \mathbb{R}^5$, uses the five second harmonic functions ($l = 2$), such as xy, so adding these to the model captures some macroscopic bending of the sphere. We find fits $F_l(S_i)$ of these models into layers l of the residual stream by least squares error on all the prompts. For ablation, we project the residual stream along some subspaces built from $F_l(S_i)$ onto data centroids while keeping their orthogonal complement unchanged. We do this intervention at every token position (during prompt prefill as well as at each generated token) and at every layer where the top-16 PCs show the R^2-score of the sky representation of at least 80% of its maximum (see Figure 12).

The results in Figure 16 show that after ablating both $\mathrm{Span}(F_l(S_1), F_l(S_2))$, the median angular separation of the prompt object and LLM's named neighbor grows from the baseline of 17°–21° (aside from 30° of gpt-oss) to 72°-86°, not far from 90° expected under random chance. Ablating only the straight sphere ($l = 1$) gets us only halfway there, on average. As a control, we also ablate

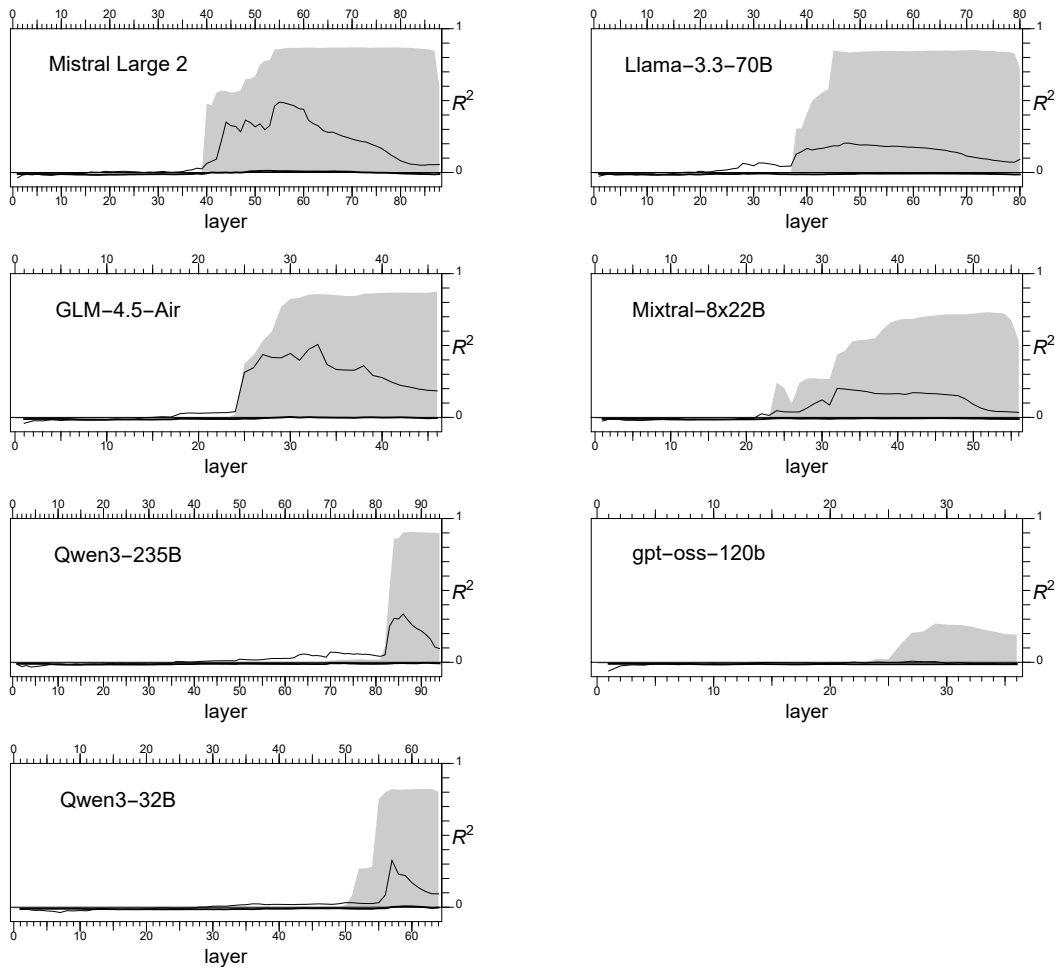

Figure 15: R^2 LOO scores for decodability of position on the sky sphere from residual stream of prompts focused on sky proximity (gray shade, top-8 PCA only) and generic prompts (thick black - on top-8 PCA components, almost 0; thin black - on top-32 PCA components).

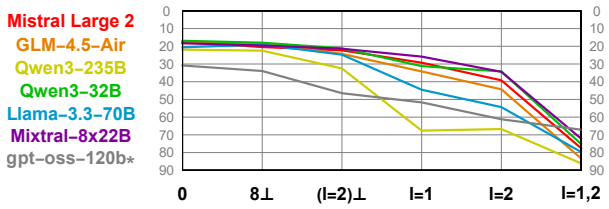

Figure 16: Ablation results. Median angular separation of the object called by a prompt and the neighbor named by an LLM. Ablating either the straight sphere model (first harmonics, $l = 1$), quadrupole moments (second spherical harmonics, $l = 2$), both, or neither (see Figure 4 for an illustration), or the 8-dim complement of both in top-16 PCA ($8\ \perp$), or second harmonics in the complement to the first ones ($(l = 2)\ \perp$)

the $\mathbb{R}^8$ orthogonal complement ($8\ \perp$ in Figure 16) of $\mathbb{R}^8$ spanned by $l = 1, 2$ modes in the top-16 PC space, with practically no change to performance.

One might notice that ablating just bend directions $l = 2$ harms the model more than the sphere itself $l = 1$. However, that is somewhat misleading, since ablating pure bends ($l = 2$ direction projected to the orthogonal complement of $l = 1$ directions) hurts the performance much less ($(l = 2)\ \perp$

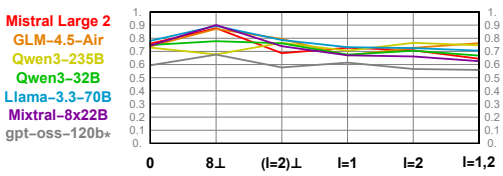

Figure 17: Fraction of usable responses across ablation regimes.

in Figure 16). Thus, the bigger impact of $l = 2$ ablation might not be indicative of their higher importance, but rise out of a) being not independent from $l = 1$ modes and hurting them during ablation, and b) spanning a 5-dimensional space vs the 3-dimensional space of $l = 1$ modes, and by that virtue alone deleting more information during ablation. However, we have not analyzed this further.

To parse the answers, we match LLMs output to our list (expanding the list of stars to include magnitude 4 and brighter) and compute the mean if a list of objects is provided (if list structure is not detected, the first mention is recorded; landmark answers like "north pole" or the original object outside of repetition of the question are discounted). We note that ablation did not noticeably affect the overall capacity of a model to name a usable object (Figure 17), so the effect in Figure 16 is specifically spatial disorientation rather than an overall breakdown.

E.2 Steering

The setup for steering is similar to that of Appendix E.1, but to be extra cautious, we don't modify the last layer, and during prefill intervene only on the last prompt position (and each subsequent generated-token input position). The least squares fit is also tweaked as follows:

$$F_l = \arg\min\Big(\min_{I_s, I_o \in \mathbb{R}^{16}} \sum_{\tau \in \text{temp}^*} \sum_{x \in D} |(F_l(S_{1,2}(x)) + I_s \mathbf{1}_{x \in \text{stars}} + I_o \mathbf{1}_{x \in \text{other}}) - P_l(\tau, x)|^2\Big) \quad (1)$$

Here the object x runs through the dataset D (see Appendix B); τ runs through prompt templates temp*, excluding the two asking for a constellation, since we use them for testing (Table 3); $P_l(\tau, x)$ are the top-16 residual stream components on the corresponding prompt in layer l and $S_{1,2}$ gives $l = 1, 2$ spherical harmonic functions on the unit sky sphere. The change we introduce is the arbitrary independent intercepts I_s, I_o for the classes "stars" and "other". That allows to use them for learning the common representation F_l without the potential of the type-encoding seen in Appendix H to skew the data.

Then we feed prompts from our dataset that ask which constellation is closest to a given one (A). To intervene, we replace the components in the "bent sphere" subspace $\text{Span}(F_l(S_{1,2}(\mathbb{R}^8)))$ by the bent sphere prediction $F_l(S_{1,2}(x))$. The target x is either a different constellation B (chosen so that the centroids of A and B are as orthogonally placed on the sky as possible), or the constellation A itself. Intervention is done in the same layers as in Appendix E.1 (minus the last one), those with the sky model signal of 80% or more of maximum. Alternatively, we write in only the prediction $F_l(S_1(x))$ of the non-bent model S_1. The results (Figure 18, a copy of Figure 7) show that such intervention switches LLMs from mostly naming constellations within 30° of the original A to mostly within 30° of the target B or B itself, but using only the non-bent version (only $l = 1$ modes used) is much less precise. Steering towards A itself results in minimal change of performance.

Parsing was handled the following way. Echoes of the question were skipped. Apart from that, all parsed constellations until an end of the sentence are gathered and the distance to either A or B is computed as the mean of angular distances from all parsed constellations. Same is done when LLM provides a list of options as if to create a quiz question out of the prompt. In some cases the model doesn't give a direct answer and just rambles about properties of A, those answers were placed in a separate category.

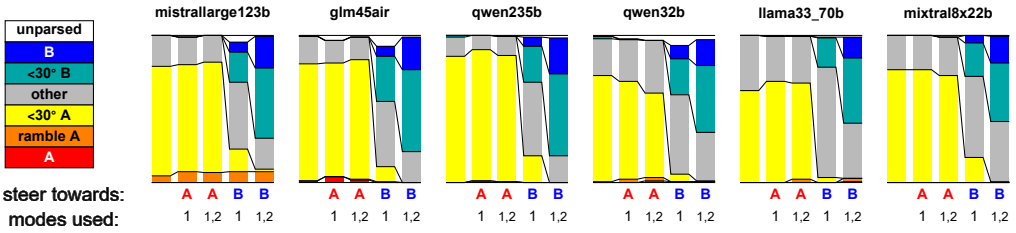

Figure 18: Steering experiment. No intervention (leftmost column) vs steering towards the original constellation A or a new target B by using either straight sphere model ($l = 1$), or accounting for second spherical harmonics ($l = 1, 2$). Steering towards the original constellation A doesn't change behavior (naming constellations near A), steering towards B switches to naming neighbors of B, more precise if higher harmonics are used.

F Testing alternative representations

We argue that the representation we found is indeed based on a spherical feature manifold, rather than a leak from a conceptually different but correlated representation. To that end, we pit our model against two other possible representations of the night sky objects. We present evidence that they are essentially absent from the top-8 PCA where the spherical representation resides. We also consider a variety of spectral embeddings based on constellation adjacency and co-occurrence in texts. A pure spherical fit outperforms all of them, leaving questions about the mechanisms through which the representation is learned.

F.1 Flat chart

Since the location on the night sky is typically documented as RA/Dec[1] coordinates, that are night sky equivalents of longitude and latitude, one might expect a representation that is linear in those coordinates (a flat 2-D map). This is the representation analyzed in Gurnee & Tegmark (2024) for the case of continents on the globe and states of USA. Another possibility tested in Tiblias et al. (2025) is a cylinder, a more spatially faithful representation, but still an approximation, built as a linear combination of one-dimensional structures.

It is not trivial to separate these models, since their North-South dimension is highly correlated[2]. To distinguish them, we model the metric as a direct sum of terms coming from individual dimensions of each of competing geometries, and score them only by the less-conflated terms.

Here are the details. First, we exclude constellations from the pool. Some of them cross the $\mathrm{RA} = 0^\circ \equiv 360^\circ$ discontinuity, making their placement on the flat chart ambiguous. The rest of the constellations are excluded too, to preserve the symmetry. Next, for each layer l we take top-8 PCA space of the residual stream[3], scale it to make the top variance be 1, and compute the RDM matrix M defined via

$$M_{ij} = |v_i - v_j|^2 \tag{2}$$

where v_i is the vector representing object i (averaged over prompt templates). We use the square of the distance so that such RDM would be additive for direct sums of representations, which we rely on. We perform a linear regression

$$M = \eta + \sum_\chi \beta_\chi M_\chi \tag{3}$$

where η is the minimized error, M_χ is the RDM matrix defined similarly for each of the coordinates listed in Table 7 (normalized to variance 1), and β_χ is the corresponding weight parameter. Table 7

[1] Right Ascension and Declination.

[2] Dec is almost identical to $x = \sin(\mathrm{Dec})$, RA is correlated with $y = \sin(\mathrm{RA})\cos(\mathrm{Dec})$, equatorial dimensions are highly correlated between sphere and cylinder.

[3] Fitting linearly the RA/Dec model mostly put it aligned with the sphere representation at the top, so we focused on that space.

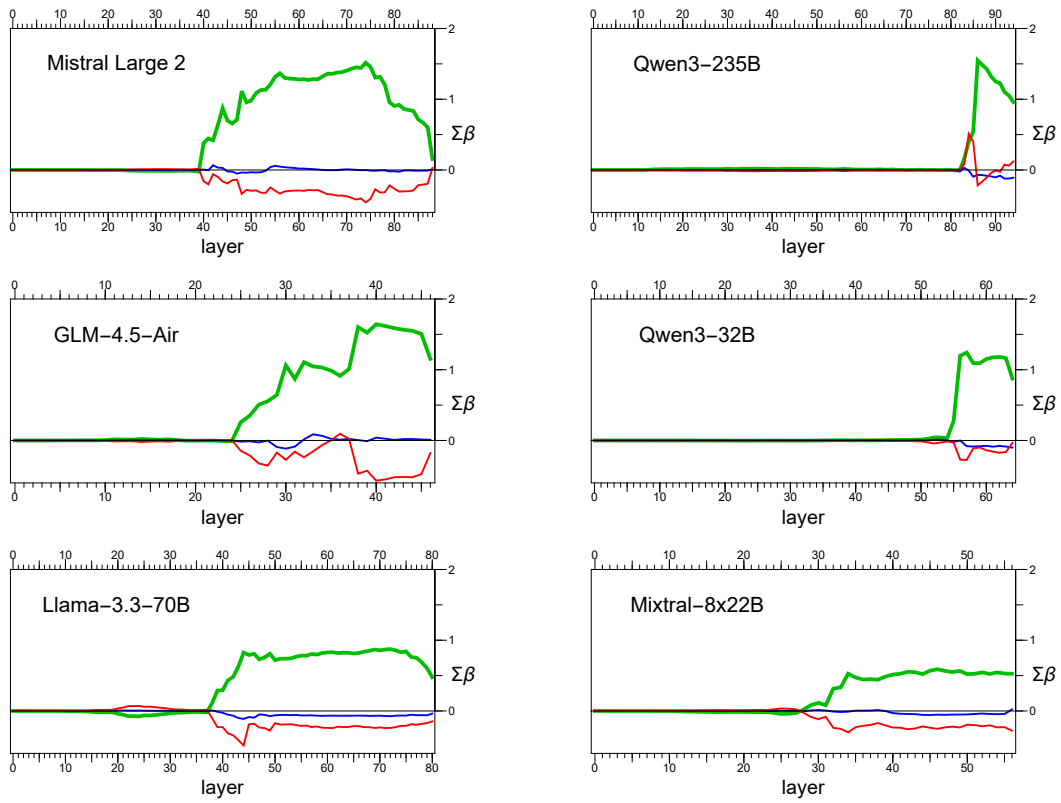

Figure 19: The metric RDM observed in the top-8 PCA space is regressed to RDMs that build up competing geometries (Equation 3). Candidates are scored by their weights, excluding the North-South direction (since it is shared or highly correlated). Sphere — green thick line on top, cylinder — thin red line, flat chart — thin blue line

is intentionally vague about whether cylinder takes the Dec-based coordinate from the sphere or from the flat map — it will not affect the score.

Table 7: Coordinate components of models: sphere vs cylinder vs flat

Sphere			Cylinder		Flat	
$y = \cos(\mathrm{Dec})\sin(\mathrm{RA})$	$z = \cos(\mathrm{Dec})\cos(\mathrm{RA})$	$x = \sin(\mathrm{Dec})$	$\sin(\mathrm{RA})$	$\cos(\mathrm{RA})$	Dec	RA

Since $x = \sin(\mathrm{Dec})$ and Dec are almost identical, the split between β_x and β_{Dec} is numerically unstable and meaningless in practice. To circumvent this issue, we exclude both from every model during scoring and record the total β using weights only of the remaining contrasting coordinates. For the cylinder and the sphere they still correlate a lot, making their split noisy, but even so, the sphere model is the only one that gets consistent positive scores. We take this as evidence of a complete absence of the other representations (a direct sum would've given a positive on average signal to every participating model even if some common components were ignored for scoring).

F.2 Correlation with other models

Having excluded the flat and cylindrical charts, we would like to see if there are models that would fit our data better than the literal sphere, possibly giving a hint for how it is learned. The models used are listed in Table 8 (see Figure 9 for illustration of the first three).

We compare these embeddings to the one observed in the residual stream: we compute metrics (RDMs) for each layer (distance squared between template-averaged residual vectors, top-128 PCA)

Table 8: Models correlated with neural representation

Label	description
S^2	The literal sphere: objects are embedded by their coordinates on the night sky (centroids for constellations).
$L(S^2)$	A sphere with higher harmonics: since our representation is bent (Appendix E.1, E.2), it makes sense to add other modes: n-th coordinate is given by $\phi_n e^{-k\lambda_n}$ where λ_n and ϕ_n are n-th eigenvalue and eigenfunction, and $k \approx .2$ is the decay rate.
G_{touch}	The graph built by joining constellations that share a border and attaching other objects to the constellations they reside in (the attaching length is different from the length of edges between constellations; the optimal length ratio is found by a sweep). The graph is embedded by the graph Laplacian eigenfunctions similar to the previous point. This structure is motivated by other instances of hierarchical data representation in LLMs.
G_{text}	spectral graph embedding as above, but the graph is based on whether objects appear close together in corpus[4]: those that appear within 10 words of each other. Counts are gated by requiring astronomical context to weed out occurrences with a different meaning (Andromeda TV show, "microscope" the instrument rather than constellation).
Q_{text}	An application of a general co-occurrence based embedding theoretically derived and experimentally supported by Karkada et al. (2026) (details in Appendix F.3). Gated similarly to the previous item.

with the caveat that we move "constellation", "star" and "other" datasets so that their centroids coincide. We average these RDMs across layer with strong sky signal (at least 80% of maximal R^2 in top-16 PCA). Then we record correlation of this metric with the one derived from the above models, where for spectral embeddings we do a sweep for the rate k of exponential decay of higher modes. The results are summarized in Figure 20.

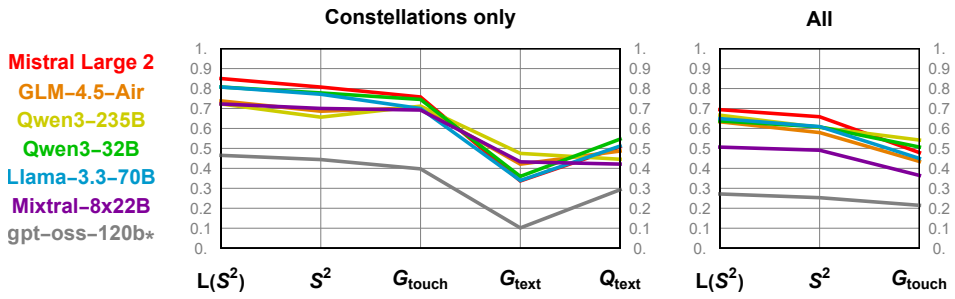

Figure 20: Pearson correlation of observed neural metric of the representation (RDM averaged over active layers) with metrics of other models (Table 8).

For co-occurrence based models we limit our scope to constellations only, since many obscure stars have poor statistics (and the constellations have a better signal on their own anyway). The true sphere tends to be in the lead, second only after a strictly more flexible model that includes all spherical harmonics. We have seen the importance of higher modes in ablation and steering experiments (Appendices E.1, E.2), but their inclusion does not improve correlation drastically. The constellation proximity graph gives comparable results to the plain sphere (even outperforming it in Qwen3-235B and GLM-4.5-Air), but only when restricted to constellations. In the regime with all objects, where this graph was meant to model a hierarchical representation, it falls behind the sphere (except gpt-oss without harmony format, not shown).

The absence of a model that would match data better than an ideal sphere (ground truth) indicates that we don't have a good candidate explanation of mechanisms by which the sky representation arises in models.

F.3 CO-OCCURRENCE METRIC

Karkada et al. (2026) explain geometrical arrangements of data in the neural space via the frequency of their co-occurrence in training data, with the most striking example being historical dates, events and headlines arranged along a Lissajous-like curve. They show that word-embedding algorithms like word2vec learn top eigenmodes of a normalized co-occurrence matrix M^*, approximately equal to the pointwise mutual information matrix:

$$M^*_{ij} = \frac{(P_{ij} - P_i P_j)}{\frac{1}{2}(P_{ij} + P_i P_j)} \approx \log\Big(\frac{P_{ij}}{P_i P_j}\Big)$$

where P_i is a (unigram) probability of token i and P_{ij} is co-occurrence probability of tokens i and j, so positive M^* corresponds to tokens i and j occurring close to each other more often than if they were drawn independently. According to Karkada et al. (2026), co-occurrence matrix M^* (with SVD $M^* = \Phi\Lambda^*\Phi^T = (\Phi\Sigma)(\Phi\Sigma)^T$, where $\Sigma := \sqrt{|\Lambda^*|}$) leads to learned embedding, equivalent to

$$W_{\textbf{Coocur}} = \Phi\Sigma.$$

We learn co-occurrence matrix $M*$ from the same corpus[5] as Karkada et al. (2026), with same gatings and precautions as listed for G_{text} in Table 8, then compute RDM (distances squared) matrix for that embedding, and correlate it with the one observed for residual vectors. The correlation is better than a naive "co-occurrence graph" model G_{text}, but worse than all models that we constructed from actual relations on the night sky. However, given that obscure constellations are much less frequently mentioned than months and years used in the original paper, our attempt to use their methodology might be held back by poor statistics and should not definitively rule out this mechanism.

F.4 3D CHART, RADIAL ANALYSIS

Another conceivable alternative representation is something akin to a genuine 3D map of the celestial objects that properly accounts for the distance to them rather than projecting them onto a unit sphere. However, expecting a literal undisturbed 3D map in this case is unreasonable, since the distances to the stars in our set span $\sim$ 3 orders of magnitude, including galaxies adds several more, and constellations just don't have a meaningful notion of distance to them. Even so, it raises the question of whether the radial direction in the observed representation does encode the distance to the object in some warped way. And if not — whether it encodes or is influenced by some other factors.

Figure 21 shows the correlations of the radial direction with various parameters. Black bars indicate the significance of the correlation ($p = 0.05$ corresponds to a half-height bar). Both are computed by pearsonr and spearmanr from scipy.stats. We see that (Spearman) correlations with distance (**dist**$^*_S$) are weak and disparate, with low significance (no model reaches $p = 0.05$). Given the textual origin of the data, the textual frequency (**zipf**) is the next factor that we test against, with more pronounced results for constellations (**zipf**$^\square$), but not for stars (**zipf***), placing more mentioned constellations slightly closer to the origin. Brightness is a parameter that is both physical and reflecting how noticeable it is to people. We see mostly positive but weak correlations of radial direction with brightness (**-mag**, negative apparent magnitude), and the significance mostly falls short of $p = 0.05$. The only really robust correlations we found were with the object type, placing constellations farther (**I**$(\square) > 0$) and stars closer (**I**$(*) < 0$).

To do this analysis, we pick the level with the best median angular error α in LOO analysis on the top-8 PCA subspace (computed as in Appendix D), and discard from the pool the objects whose MSE for the angle is more than double of the average, to eliminate outliers and get a cleaner signal. We use linear regression on the full dataset to project the top-8 PCA space onto the 3-dimensional space that hosts the unit sphere sky model. For the retained objects, we compute correlations of the length of their projection (averaged over prompts) with the following functions listed in Table 9.

[5] Wikipedia 2023 dump

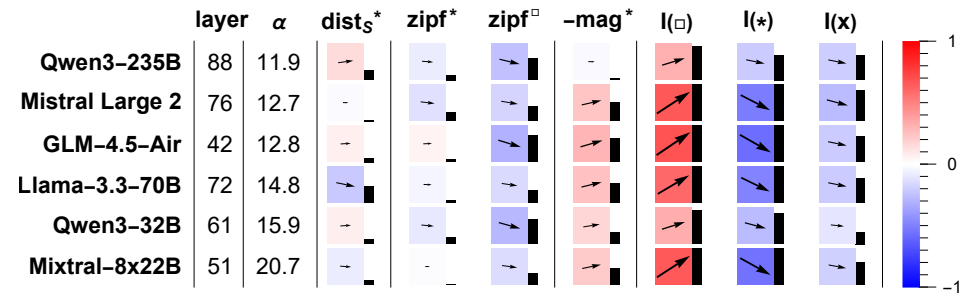

Figure 21: Correlations of radial direction with various functions (see Table 9) at the layer with least median angular error α. Black bars represent significance of the correlation, height is $1 - p^{\ln(.5)/\ln(.05)}$ ($p = 0.05$ gives height 0.5). Arrows indicate correlation direction for black and white version.

Table 9: Functions correlated with radial length.

Label	Function correlated	Domain
dist_S^*	Distance, from SIMBAD parallax; Spearman corr.	Stars only
zipf^*	Log frequency in text corpus, from wordfreq.zipf_frequency	Stars only
$\text{zipf}^\square$	Log frequency…, " constellation" stripped	Constellations only
$-\text{mag}^*$	Brightness as negative apparent magnitude, from SIMBAD	Stars only
$I(\square)$	Indicator function of "constellation" type	All
$I(*)$	Indicator function of "star" type	All
$I(\text{x})$	Indicator function of "other" type	All

G J SPACE

We train J-lens on WikiText-103 as in Gurnee et al. (2026b) by averaging the Jacobian of the feed forward map over prompts and token positions $t \le t'$:

$$J_l := \mathbb{E}_{t \le t'}\left(\frac{\partial h_{\text{final},t'}}{\partial h_{l,t}}\right)$$

To estimate the location of J-space we compute its effective relative dimension in each layer. Let σ_i be singular values of J_l. If n of them were equal and the rest were 0, then

$$\left(\sum \sigma_i\right)^2 / \sum \sigma_i^2 = (n\sigma)^2 / n\sigma^2 = n \tag{4}$$

provides the rank of J_l, so for arbitrary J_l this formula can serve as a proxy for the dimension of the subspace effectively attended to by J_l. So we define relative effective dimension as

$$\text{eff.dim}(J) := \left[\left(\sum \sigma_i\right)^2 / \left(\sum \sigma_i^2\right)\right] / d \tag{5}$$

where d is the total dimension (width of the model). We track these effective dimensions together with the R^2-signal of the sky representation in top-16 PCA in Figure 22. Triangles below the bottom axes of the plot are halves of similarity matrices for J_l (the bottom axis of the plot is the diagonal of the matrix). The similarity (i, j)-th entry is given by centered kernel alignment (CKA, (Kornblith et al., 2019)) between J_i and J_j (similar results can be obtained just with cosine similarity between centered $J_i^t J_i$ and $J_j^t J_j$). The J-space is visible as a distinguished block on the right, occupying roughly half of depth in most models, but only small fraction in Qwen models, just like the sky signal.

We note, however, the representation directions do not align with directions distinguished by the J-lens, interpreting this mismatch is a work in progress. One potential explanation could be that J-lens is tailored toward interpretability and related everything to the last layer, so if we probe J-space with the J-lens that is biased towards the output layers, it might undervalue purely internal

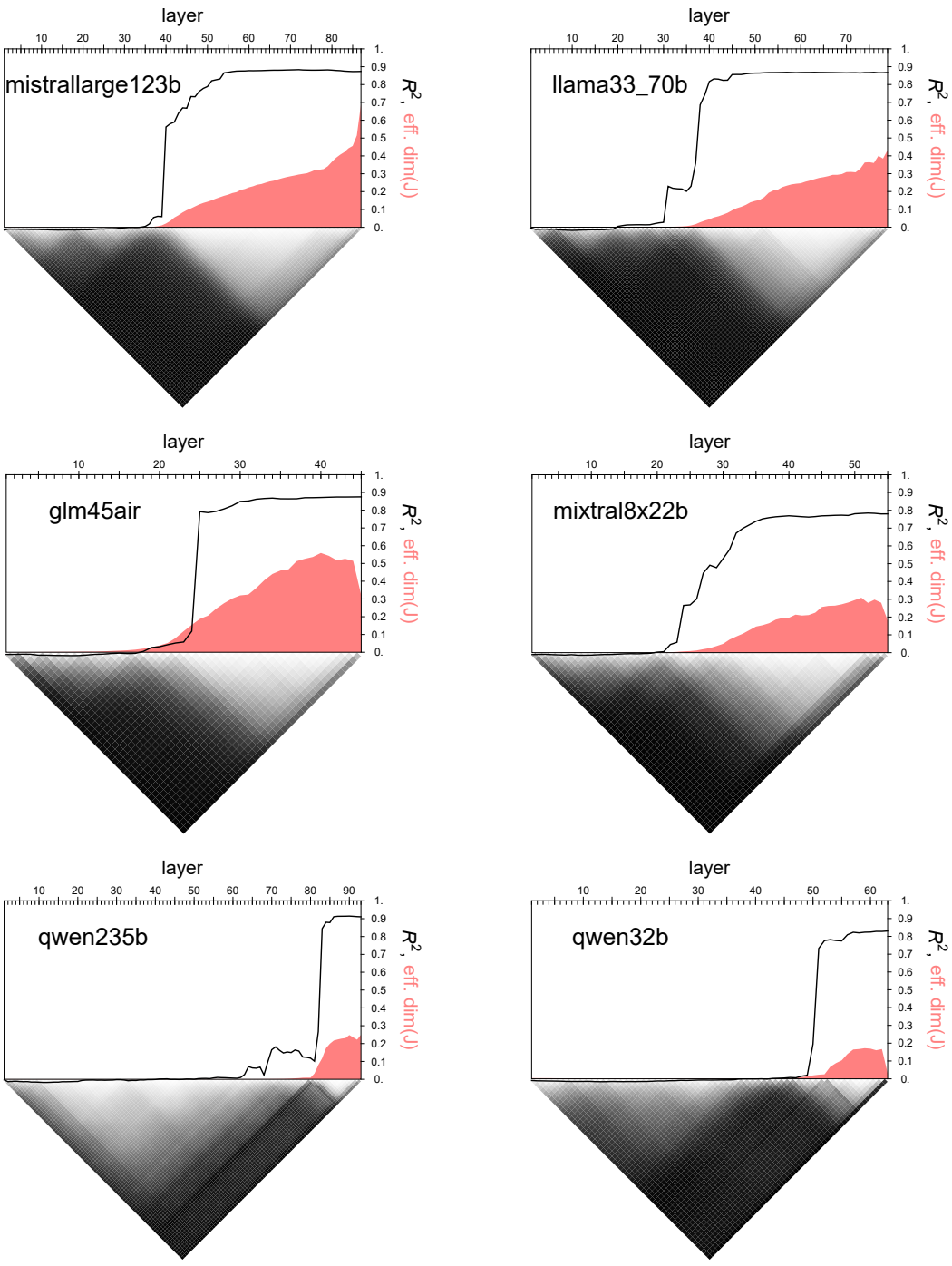

Figure 22: J-space correlation with sky representation across layers. Black line: R^2 sky signal in top-16 PCA. Red shade: relative effective dimension of J-space (Equation 5). Below: similarity of J-matrices between different layers (CKA (Kornblith et al., 2019)).

concepts whose conversion to the output layer doesn't survive prompt-averaging, and directions on our "mental map" representation might be better valued by layer-to-layer J-lens than layer-to-output.

H PCA COMPONENTS AHEAD OF SKY SPHERE

In many models, the sky sphere representation rises to the very top of PCA components (blue in Figure 12) on proximity focused prompts. However, even in the best examples it falls short from crystallizing as the top-3 components (see for example the top-8 PCA of Mistral model, Figure 23). This is mostly due to the direction (yellow) that separates the concepts "star" and "constellation" prevalent in our test data (see Appendix D for the definition of the yellow direction).

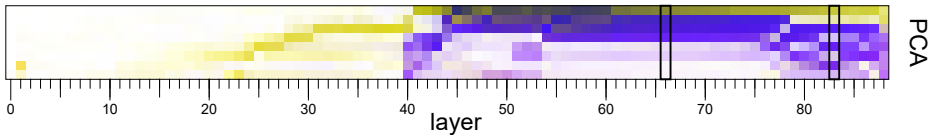

Figure 23: Partial copy of Mistral chart from Figure 12 with analyzed layers 66 and 83 marked.

Another way to analyze the competitor direction is to pick the top-4 components for a layer, where they are saturated with the sky sphere model, like level 66 in Mistral model, and isolate the direction orthogonal to the sky model (Figure 24). The prompts whose object X is a constellation (blue) land on one side, and those where X is a star (red) land on the other, with miscellaneous objects (black) spread in between. The three notable exceptions (highlighted in green) correspond to the cases where the template itself contains the word "constellation" or "star", landing them firmly in the corresponding group. One softer outlier (highlighted in gray) is the template

- "**astro**photographers find X by first locating", tilts towards stars,

which is in line with our interpretation: indirect "star" mention produces a mild bias.

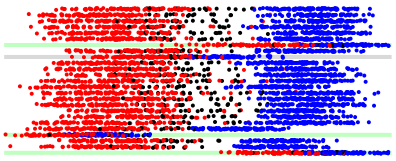

Figure 24: Mistral Level 66 top-4 PCA components minus the sky sphere:
object is **STAR**, **CONSTELLATION**, **NEITHER**



Figure 25: Mistral Level 83 top-2 PCA components minus the "type" direction:
last word is **IS**, **BESIDE**, **NEAR**, **other**

While the "star"-"constellation" direction is the only unsurpassed rival to the sky sphere in many models, we can also clearly see another direction surface up in layers 79-87 in Mistral. Plotting the direction, orthogonal to "star"-"const" direction in layer 83, reveals its meaning (Figure 25): it is focused on the last word of the prompt.

We note that while we provided the plots only for one layer in each case, the plots for all layers 54-75 and 79-87, respectively, are essentially the same.