A NIGHT AT THE LLM ZOO: DATA CONDITIONED FLOW MATCHING FOR GENERATING PRETRAINED LLM WEIGHTS

Anonymous authors
Paper under double-blind review

ABSTRACT

A challenge to Weight Space Learning as it expands from task specific and vision models to Large Language Models (LLMs) is the absence of a well curated and analyzed LLM population. To address this, we train such a model zoo for application to Weight Space Generation by systematically varying the data mixture composition between web text, books, mathematics, coding, and French across populations of 51M, 124M, and 355M parameter GPT-2 Models, each tier with count $N = 100$. We analyze these populations by cosine similarity, functional diversity, Leave-One-Mixture-Out modeling, and variance decomposition. We show that the zoo contains diverse models whose variation is structured by training mixture composition. We embed the training mixture composition to condition a Flow Matching model of *complete* pretrained LLM weights in Gram-PCA compressed weight space. By demonstrating zero-shot generalization to held out mixtures, measuring Continuous Ranked Probability Score skill, as well as finding minimal changes in downstream test-set perplexity and general capability benchmark scores, we show that our generator generalizes on the simplex of data mixtures to produce models that perform similarly to those in the zoo *ex nihilo*. Finally, we produce all source code and artifacts to support the continued expansion of Weight Space Learning and Generation into the LLM domain.

1 INTRODUCTION

The success of modern deep neural networks relies heavily on large-scale computation, and the past decade has produced a rapidly growing population of pretrained models, from computer vision systems to large language models (LLMs). Despite the substantial computational cost invested in training these models, the information contained in their learned parameters remains comparatively underexplored for understanding how neural networks evolve during training or for constructing new models. A central difficulty is that publicly available checkpoints are highly heterogeneous: they often differ in architecture, scale, initialization, training data, and optimization procedure, while many of these details are only partially documented. As a result, collections of existing models do not provide a controlled population from which the structure and dynamics of LLM weight space can be readily studied.

A recent surge of research on weight-space learning addresses this problem by treating the parameters of trained neural networks as data (Han et al., 2026; Wang et al., 2026). However, existing studies remain concentrated on relatively small vision and classification models, or on parameter-efficient representations such as Low-Rank Adaptation (LoRA) weights, leaving the full parameter space of pretrained language models comparatively unexplored (Charakorn et al., 2026; Huang et al., 2026). This limitation can be attributed to the difficulty of learning meaningful structure from collections of LLMs whose training conditions and architectural differences are uncontrolled. Among the few works that extend full-weight generation to natural-language models, DeepWeightFlow (Gupta et al., 2026) trains a population of 100 BERT encoders from scratch and models their parameters; however, these models are task-specific regressors trained on a single supervised objective rather than pretrained language models which consider data from multiple diverse sources. Similarly, Soro et al. (2026) studies full LLM weights in the context of model merging and highlights the difficulty of obtaining compact and meaningful representations of large language-model parameters. Whether

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.

systematic and exploitable structure exists in the full parameter space of models trained under a general language-modeling objective therefore remains an open question.

We argue that addressing this question requires a controlled model zoo specifically designed for studying pretrained LLM weight space. Rather than assembling unrelated checkpoints from public repositories, such a zoo should systematically control the factors that generate variation across models, making it possible to associate changes in model parameters with known changes in pretraining conditions. Motivated by this challenge, we construct a structured zoo of pretrained language models containing more than 300 models trained under systematically varying pretraining data compositions. This controlled population provides a foundation for studying how LLM weights vary across training distributions, characterizing the geometry of their parameter space, and ultimately determining whether this structure is sufficiently regular to support the analysis and generation of full language-model parameters.

Furthermore, we train conditional flow matching models (Lipman et al., 2023; Liu et al., 2023a) to generate the weights of GPT style models (Radford et al., 2019) at 51M, 124M and 355M parameters, conditioned on the pretraining data mixture. Earlier work on parameter generation has mostly focused on small vision models or low rank adapters (Peebles et al., 2022; Soro et al., 2025b; Wang et al., 2025; Charakorn et al., 2025; Huang et al., 2026), whereas we generate complete pretrained language models. Following Gupta et al. (2026), we compress each zoo with Gram PCA, train the flow in this compressed space, and decode the samples back into full GPT-2 weights, which we evaluate directly without any further training. The generated models perform close to the original trained models and generalize well to data mixtures held out from training. They stay closer to the real models than a nearest neighbor retrieval baseline, and their CRPS skill (Gneiting & Raftery, 2007) shows that conditioning on the mixture clearly helps. Our results show that the generated weight spaces generalize zero shot to new data mixtures.

The primary contributions of our work are:

- We build a controlled zoo of 300 GPT-2 models at 51M, 124M and 355M parameters, in which the only factor that varies between models is the composition of the pretraining data.
- We show that the weights in this zoo are structured by data mixture. Models trained on the same mixture are strongly aligned, models trained on different mixtures are nearly orthogonal, and the mixture explains more than 85% of the variance among repeatedly drawn models.
- We train a conditional flow matching generator that produces complete pretrained language models for unseen data mixtures across model parameter scales. The generated models achieve an average perplexity change of 4.1 to 5.1%, compared to 5.8 to 6.0% for a nearest neighbor retrieval baseline, and perform on par with our real models on HellaSwag (Zellers et al., 2019).

2 Related Work

Weight-space learning. A growing body of work treats trained parameters as a data modality in their own right (Han et al., 2026; Wang et al., 2026). Early studies predicted model properties such as accuracy directly from weights (Unterthiner et al., 2021; Eilertsen et al., 2020). Later work designed architectures that respect the symmetries of weight space, including permutation-equivariant layers (Navon et al., 2023; Zhou et al., 2023), graph metanetworks (Kofinas et al., 2024; Lim et al., 2024), and set-based encoders (Andreis et al., 2024). A parallel line learns weight representations with autoencoders, yielding hyper-representations used for both property prediction and sampling (Schürholt et al., 2021; 2022a; Schürholt et al., 2024). These methods are developed and evaluated on small vision classifiers and implicit neural representations, not on pretrained language models. In addition, weight-space learning relies on populations of models rather than isolated checkpoints. Controlled model zoos vary seeds and hyperparameters while holding architecture fixed (Schürholt et al., 2022b), and later work studies how zoo composition affects learned representations (Falk et al., 2025a) or extends the setting to heterogeneous public checkpoints (Falk et al., 2025b). These zoos are primarily vision-based. We adopt the same controlled setting for language models, varying only the pretraining data composition rather than optimization or other hyperparameters.

Weight generation. Generating parameters instead of optimizing them dates back to hypernetworks (Ha et al., 2017) and to graph hypernetworks that predict parameters for unseen architectures (Knyazev et al., 2021). Generative models over checkpoints followed, including diffusion over optimization trajectories (Peebles et al., 2022), diffusion (Soro et al., 2025b; Wang et al., 2025) weights, autoregressive parameter generation (Soro et al., 2025a), and flow matching in a re-basined weight space (Gupta et al., 2026). In the language domain, generation is largely confined to low-rank adapters, where a hypernetwork maps a task description or a batch of unlabeled prompts to LoRA matrices (Charakorn et al., 2025; Liang et al., 2025; Charakorn et al., 2026; Huang et al., 2026). Such methods produce a small update on top of a frozen backbone and therefore do not model pretrained parameters themselves. The exception is Gupta et al. (2026), which trains a population of BERT-sized encoders and generates their full weights, but that population is trained on a single supervised regression task rather than on general-purpose language modeling.

Controlled suites and data composition. Open suites release language models and intermediate checkpoints to study training: MultiBERTs varies random seed (Sellam et al., 2022), Pythia varies scale while keeping data and ordering fixed (Biderman et al., 2023), and PolyPythias extends this to multiple pretraining runs (van der Wal et al., 2025); LLM360 (Liu et al., 2023b) and OLMo (OLMo et al., 2024) similarly release open training artifacts. These suites are designed to isolate factors such as scale, seed, or training step and therefore largely keep the pretraining distribution fixed. The complementary literature instead varies data composition: DoReMi learns domain weights (Xie et al., 2023), data mixing laws relate mixture proportions to language-modeling loss (Ye et al., 2025), RegMix predicts performance from sampled mixtures (Liu et al., 2024), and DCLM compares curation strategies under a standardized recipe (Li et al., 2024). These works study how data composition affects performance, but not how it is reflected in the resulting parameters. We retain the weights and ask whether the pretraining mixture is recoverable from weight space.

3 METHODS

3.1 PROBLEM SETUP & NOTATION

Let $\mathcal{T}_1, \ldots, \mathcal{T}_K$ denote K pretraining corpora over distinct text domains (see Table 12 for details), where $K = 5$. Let $\pi \in \Delta^{K-1} = \Delta^4$ be a mixture over these domains, with π_d denoting the weight on domain $\mathcal{T}_d$. Each pretraining mixture π produces a distribution of model parameters θ, where all models have the same number of parameters D. Given a model zoo $\Theta = \theta_i i = 1^N$, where each model θ_i was pretrained on a different mixture π_i, we study how the model weights change as the pretraining mixture changes. Our goal is to determine whether these changes follow a regular pattern that can be utilized to learn a conditional generator $f\psi(z \mid \pi)$ to generate plausible model parameters for mixtures that were not seen during training. In practice, rather than generating parameters directly in the high-dimensional parameter space $\mathbb{R}^D$, we learn this mapping in the latent space introduced in Section 3.3.

3.2 LLM ZOO CONSTRUCTION

Each zoo holds $N = 100$ members over 85 distinct mixtures. Twenty are *anchors*: the five one-hot vertices of Δ^4, each replicated over four independent branches at the same π. The remaining eighty are *singletons*, at unique interior mixtures drawn from $\mathrm{Dirichlet}(1)$ and rejection-sampled at a minimum pairwise L_1 gap of 0.15. The anchors are the only mixtures drawn more than once, hence the only place a within-mixture quantity can be estimated.

We construct the zoo to capture variation in the weight spaces directly attributable to pretraining compositions such that all members share architecture, optimizer, schedule, token budget and initialization, and only π_i and a branch seed vary. For the first $(1 - \beta)S$ of S optimizer steps a single model trains on a shared stream, forming a *trunk*, which is then copied into N branches; branch i trains for the remaining βS steps on π_i, under a schedule spanning the full trajectory so that warmup does not restart. We set $\beta = 0.15$ and a budget of 20 tokens per parameter (Hoffmann et al., 2022), and build three zoos at 51M (Mini), 124M (Small) and 355M (Medium) parameters, the latter two being published GPT-2 configurations Radford et al. (2019). Across these scales only a single

mixture plan is drawn, so across scales member i carries an identical π_i. Additional details on Zoo construction can be found throughout Appendix A.

3.3 Zoo Encoding and Weight-Space Representation

We follow the Gram-PCA compression technique for extending weight space generation beyond $\mathcal{O}(100\text{M})$ developed by Gupta et al. (2026), representing a given zoo tier with population $N = 100$ in dimension $k = N - 1 = 99$ codes for zoo diversity analysis, and $k = 50$ codes for generative modeling experiments. Details are explored in Appendix B (Gupta et al., 2026).

We choose rank $k = N - 1 = 99$ for reporting population data, while generative experiments are done at $k = 50$. The cost of this truncation is measured at $\sim 2\%\Delta\text{PPL}$ in Table 4. Due to our method of zoo training, detailed in Section 3.2 and Appendix A, we do not need to re-basin the population (see Appendix A.5)(Frankle et al., 2020). Finally, a notable limitation of this Gram-PCA technique is that it must be fitted on every member, even in LOMO or zero-shot settings. Appendices B.2, A.5 and B.3 expand on these points.

3.4 LLM Zoo Weight Generation

We model $p(z \mid \pi)$ with a rectified flow (Liu et al., 2023a), the linear-interpolant case of conditional flow matching (Lipman et al., 2023). For a member code z_1 and Gaussian source $z_0 \sim \mathcal{N}(0, \sigma_0^2 I_k)$ we take $z_t = (1-t)z_0 + tz_1$ and regress a velocity field on the constant target $z_1 - z_0$,

$$\mathcal{L}(\psi) = \mathbb{E}_{t\sim\mathcal{U}[0,1],\,(z_1,\pi)\sim\mathcal{Z},\,z_0} \left\| v_\psi\big(z_t, t, e_\psi(\pi)\big) - (z_1 - z_0) \right\|_2^2, \tag{1}$$

where v_ψ is an MLP and e_ψ a two-layer network embedding π into a vector concatenated to the velocity network's input (Perez et al., 2018), trained jointly with the flow (Gupta et al., 2026; Soro et al., 2025b; Saragih et al., 2025); a learned null token with conditioning dropout gives classifier-free guidance (Ho & Salimans, 2021). Sampling integrates $\dot{z}_t = v_\psi$ by the explicit Euler method, and the resulting latent PCA-code is decoded and evaluated without further optimizer steps. Further details and ablations can be found in Appendix B.6.

3.5 LLM Zoo and Generated LLM Diversity

3.5.1 LLM Zoo Diversity

We adopt the reporting conventions of vision model zoos developed by Schürholt et al. (2022b); Unterthiner et al. (2021); Eilertsen et al. (2020). However, some are adjusted due to our trunk-branch approach. Our zoo also contains only a single optimizer and model architecture, only varying the data supply, $\pi \in \Delta^4$, across the population at a given scale.

Branch deltas. Every θ_i inherits the trunk, so $\|\theta_i - \theta_j\|$ are small relative to θ_i. We work instead with the *branch delta* $d_i = \theta_i - \theta_{\text{trunk}}$, member i's task vector (Ilharco et al., 2023), and read all pairwise quantities off the $N \times N$ Gram $G_{ij} = \langle d_i, d_j \rangle$ through

$$\|\theta_i - \theta_j\|^2 = G_{ii} + G_{jj} - 2G_{ij}, \qquad \|\theta_i - \theta_{\text{trunk}}\|^2 = G_{ii}, \tag{2}$$

in which the reference cancels. Displacements, not raw weights, are a numerical requirement here, not a convenience: on raw weights the first identity is catastrophic cancellation.

Cosine pools. The pairwise diversity statistic is $\cos_{ij} = G_{ij}/\sqrt{G_{ii}G_{jj}}$, read against the null for independent directions in $\mathbb{R}^D$, $|\cos| \sim D^{-1/2}$. We report it in four disjoint pools, created from anchor pairs sharing a π, anchor pairs with different π, interior (or singleton) to interior, and anchor to singleton. These pairings' displacement scales differ materially enough that they cannot be simply combined. Raw cosines confound structure induced by π with the drift inherited from the trunk, which we refer to as the mean displacement $\bar{d} = N^{-1}\sum_i d_i$. Projecting out this shared drift yields the centered Gram,

$$\tilde{G} \;=\; PGP, \qquad P = I - \tfrac{1}{N}\mathbf{1}\mathbf{1}^\top, \tag{3}$$

we report both forms when possible.

Between- and within-π variance. The four replicate branches per anchor give a genuine within-group term, admitting a one-way decomposition with $d = 5$ groups of $s = 4$,

$$\mathrm{SS}_B = s \sum_g \|m_g - \bar{m}\|^2, \qquad \mathrm{SS}_W = \sum_g \sum_r \|\theta_{gr} - m_g\|^2, \qquad \mathrm{SS}_T = \mathrm{SS}_B + \mathrm{SS}_W, \quad (4)$$

for m_g the group mean and $\bar{m}$ the population mean, making $\eta^2 = \mathrm{SS}_B/\mathrm{SS}_T$ the share of anchor variation attributable to mixture identity (Fisher, 1925). Since SS_B carries $d - 1$ degrees of freedom of within-group noise, η^2 is biased upward. Treating the five vertices as fixed effects, we report the unbiased estimate of the between-mixture variance and the corresponding variance fraction (Searle et al., 1992),,

$$\hat{\sigma}^2_{\text{dom}} = \frac{\mathrm{SS}_B - (d-1)\,\mathrm{MS}_W}{ds}, \qquad \mathrm{MS}_W = \frac{\mathrm{SS}_W}{d(s-1)}, \qquad \eta^2_{\text{corr}} = \frac{\hat{\sigma}^2_{\text{dom}}}{\hat{\sigma}^2_{\text{dom}} + \mathrm{MS}_W}, \quad (5)$$

alongside η^2 to address this. A Gram is a sum over coordinates, so equation 4 is exactly additive over disjoint coordinate regions; we quote the transformer blocks separately from the embeddings throughout, since the embedding share of D falls by $\sim 3.4\times$ across our three scales.

Leave-One-Mixture-Out. equation 4 needs repeated draws at a single π, so it is defined only at the five anchors, the vertices of Δ^4. These are the most mutually distinct mixtures the design contains. The interior of Δ^4 admits the out-of-sample form of the same question: withhold a mixture, predict its branch delta from π alone, and score that prediction against simply predicting the mean of the retained members (Allen, 1974; Arlot & Celisse, 2010). We call this Leave-One-Mixture-Out (LOMO),

$$R^2_{\text{LOMO}} \;=\; 1 \;-\; \frac{\sum_F \left\| d_F - \hat{d}_F^{(-F)} \right\|^2}{\sum_F \left\| d_F - \bar{d}^{(-F)} \right\|^2}, \quad (6)$$

where $\hat{d}^{(-F)}$ is an ordinary least-squares fit on the rows outside fold F and $\bar{d}^{(-F)}$ is the mean of those same rows, both refit for every fold.

The folds are mixtures, not members. An anchor's four branches share a π exactly, so withholding one member leaves three siblings at an identical key and the prediction degenerates to retrieval — the leakage grouped cross-validation exists to prevent whenever records are clustered.

We fit several methods of varied complexity to this objective and compare their performances against our flow generator: a linear fit on π, a linear fit on π with its entropy feature $H(\pi)$, and a quadratic fit on π. Through classical multidimensional scaling from $\mathbb{R}^k$, these same statistics transfer to $\mathbb{R}^D$ (Gower, 1966) (see Appendix B.5).

Population spectra. We report two distinct spectra: the five *centered domain means*, whose rank is bounded by $d - 1 = 4$ by construction, and all N *members* at $k = 99$. Both are summarized by the participation-ratio effective rank: $\mathrm{eff}(\lambda) \;=\; \frac{\left(\sum_a \lambda_a\right)^2}{\sum_a \lambda_a^2}$, quoted absolutely with its k stated. We do not report the normalized variant eff/k, because it is not k-invariant.

Functional diversity. Neither weight nor latent space metrics directly capture functional diversity (Wang et al., 2024; Zeng et al., 2025). Therefore, we also report diversity on downstream tasks in the form of next-token-prediction disagreement rates under greedy decoding (Germann, 2003).

3.5.2 Generated LLM Diversity

Throughout, a *cell* is one held-out mixture, the unit over which every generative comparison is paired, and an *arm* is one source of latent codes, z, evaluated on those cells. Generative models of weights can memorize their training checkpoints, particularly when trained on few samples (Zeng et al., 2025; Rao & Moyer, 2026), although implicit regularization can delay this (Bonnaire et al., 2025). Since each of our generators sees at most 85 mixtures, we test for memorization explicitly.

Continuous Ranked Probability Score. Mean-squared and mean-absolute error are not proper scoring rules for this question: both are optimized by a point mass, so a generator that has memorized

Table 1: Branch delta cosine similarity by pool, raw and after removing the drift inherited from the shared trunk equation 2. The pools are disjoint. The random-direction nulls are $\leq \mathcal{O}(1\text{E-}4)$ across scales.

		Mini (51.5M)		Small (124.4M)		Medium (354.8M)	
pool	pairs	raw	drift rem.	raw	drift rem.	raw	drift rem.
within-anchor (same π)	30	0.9153	0.8920	0.9024	0.8717	0.8505	0.8000
between-anchor (different π)	160	0.2564	−0.0697	0.2743	−0.0468	0.2423	−0.0160
singleton–singleton	3160	0.6047	0.1721	0.6172	0.1193	0.6060	0.0844
anchor–singleton	1600	0.3534	−0.1381	0.3568	−0.1276	0.3132	−0.1167
all pairs	4950	0.5141	0.0684	0.5236	0.0387	0.5011	0.0205

the conditional mean outscores the correct conditional. We therefore report the Continuous Ranked Probability Score (CRPS),

$$\mathrm{CRPS}(F, y) \;=\; \mathbb{E}\,|X - y| \;-\; \tfrac{1}{2}\,\mathbb{E}\,|X - X'|, \qquad X, X' \overset{\text{iid}}{\sim} F, \tag{7}$$

whose second term is the diversity credit the point statistics lack (Gneiting & Raftery, 2007); Appendix B.7 gives the screen that admitted it and the measurement that rejects the alternatives. CRPS needs only one observation per condition, and at $k = 50$ our flow is light enough that dense sampling is affordable. We evaluate equation 7 in code space per component, and in function space, where the generated cohort at a held-out mixture is a predictive distribution over each domain's perplexity and the real model is a single observation. We report the skill score $1 - \mathrm{CRPS}_{\text{model}}/\mathrm{CRPS}_{\text{marginal}}$ against the zoo's own marginal, bypassing π entirely. As such, a CRPS skill of zero indicates that conditioning is useless. Alongside CRPS skill, we also report round-trip errors of compression and decompression through z (see Appendices B.7 and B.4).

Functional round trip. Latent-code and weight space metrics do not directly correspond to functional capacity; therefore, we sample a latent at the requested mixture, decode it to $\mathbb{R}^D$, load it into GPT-2 and measure the $\Delta\mathrm{PPL}\%$ on five held-out domain slices (see Appendix B.9). These held out splits source data in equal proportion to avoid confounding scores. Errors are $\%\Delta\mathrm{PPL}$: per domain, $100 \cdot |\mathrm{ppl}_{\text{gen}}/\mathrm{ppl}_{\text{real}} - 1|$, formed per domain and then averaged.

Retrieval and Linear Baselines One stress test of the generator is to demonstrate the capability to zero shot generalize to held out π_i mixtures. The bar is 1-NN *retrieval*, which returns the real Zoo member whose mixture is nearest the request by L_1. A generator that has memorized its training rows and returns the member nearest the request in L_1 reduces exactly to this baseline, so beating retrieval requires more than copying. For a comparison between such a generator and retrieval system to be level, the retrieval pool must also be denied access to the neighborhood of the held out π_i.

The cheapest competing account is that θ is simply a function of π. We test least-squares surfaces on π and a weight *soup* (Wortsman et al., 2022), written in equation 9, which fits nothing at all, its coefficients being the requested mixture itself, so it is simply task arithmetic over the five vertex means (Ilharco et al., 2023). Three further measurements corroborate the retrieval comparison with independent failure modes, elaborated on in Appendices B.8 and B.12 (Meehan et al., 2020).

4 Experiments

4.1 Zoo Analysis

Cosine similarity. Raw pairwise cosine similarities are uniformly high; removing the drift inherited from the shared trunk separates them, leaving branch deltas strongly aligned within a mixture and near-orthogonal across mixtures at every scale (Table 1).

Spectra. Effective rank at $k = 99$ grows faster in the transformer blocks than in the embeddings (Table 19), so variance not captured by $\pi \in \Delta^4$ grows with model size.

Table 2: One-way between/within decomposition over the twenty anchors ($d = 5$ mixtures $\times$ $s = 4$ replicate branches, $\mathrm{df} = 4/15$), by coordinate region; η^2 and η^2_{corr} are equation 4 and equation 5. "of D" is the region's share of the parameter vector, which shifts from embeddings to blocks with scale.

	Mini			Small			Medium		
region	η^2	η^2_{corr}	of D	η^2	η^2_{corr}	of D	η^2	η^2_{corr}	of D
whole model	0.9416	0.9224	100%	0.9220	0.8965	100%	0.8661	0.8230	100%
embeddings	0.9455	0.9276	51.0%	0.9451	0.9271	31.7%	0.9377	0.9172	14.8%
blocks only	0.9112	0.8822	49.0%	0.8754	0.8353	68.3%	0.8186	0.7610	85.2%

Table 4: Every source of weights on the same 85 held-out mixtures, in $\%\Delta$PPL over the five-domain profile, averaged per mixture. The *replicate ceiling* (two real models at the same π) and $k = 50$ rank truncation provide an estimated lower bound. Retrieval is blinded to the same neighbourhood as the generator (see Table 30 for vertex mixtures.).

	Mini (51M)		Small (124M)		Medium (355M)	
source of weights	$\%\Delta$PPL	$\times$ ceil	$\%\Delta$PPL	$\times$ ceil	$\%\Delta$PPL	$\times$ ceil
replicate ceiling (seed noise)	1.01	1.0	1.09	1.0	0.97	1.0
$k = 50$ rank truncation floor	1.80	1.8	1.93	1.8	2.46	2.5
flow (ours)	**4.05**	**4.0**	**4.17**	**3.8**	**5.11**	**5.3**
retrieval (1-NN, matched blinding)	5.75	5.7	5.79	5.3	6.01	6.2
quadratic surface on π	6.54	6.5	7.27	6.7	7.88	8.1
linear surface on π	9.64	9.5	11.28	10.4	12.66	13.0
bare trunk (no branching)	12.72	12.6	11.48	10.5	11.42	11.7
global mean (ignores π)	12.43	12.3	12.66	11.6	13.57	13.9
soup (Eq. equation 9)	14.87	14.7	15.45	14.2	22.29	22.9
vertices-only flow (Tab. 30)	25.09	24.8	35.18	32.3	35.77	36.7

Variance Explained by π. Model-level $\eta^2 \geq 0.85$ at every scale (Table 2), but the twenty anchors it is measured on sit at the vertices of Δ^4, bounding the influence of π from above, not measuring it on the interior where a generator operates.

The LOMO fits of equation 6 measure the interior instead (Table 20). A quadratic surface on π is the strongest predictor at every scale, capturing roughly 15 to 35% of interior variance. Only designs carrying $H(\pi)$ predict the vertex folds, possibly because entropy is zero exactly at a one-hot mixture. This range sits far below the $R^2 = 0.948$ the same quadratic basis reaches on c0–c4 (Appendix B.4).

This gap is also a result, it indicates that the whole-delta fit is dragged down by components that carry little variance and are largely member-private, meaning that the predictable portions are the high-variance ones.

Functional diversity. Weight-space distance is not functional difference, so we also record the argmax decoded token at 327,680 held-out next-token positions per member, a comparison used to separate genuine ensembles from single-trajectory oversampling (Izmailov et al., 2019). Members with different mixtures disagree roughly twice as often as replicates of the same mixture, at every scale. Our flows incur a real $\%\Delta$PPL cost but sit nearest the replicate noise floor of any source tested, including the 1-NN lookup estimator, with the closed-form surfaces on π and the weight soup of equation 9 progressively worse (Table 4).

Table 3: Pairwise next-token disagreement: fraction of 327,680 held-out positions at which two members predict a different argmax token, over the same four disjoint pools as Table 1. Bold marks the extreme pools of each column.

pool	Mini	Small	Medium
within-anchor (same π)	0.166	0.179	0.189
between-anchor (different π)	0.411	0.396	0.373
singleton–singleton	0.195	0.183	0.184
anchor–singleton	0.345	0.328	0.312
all pairs	0.250	0.236	0.232
between/within anchor	2.48	2.21	1.97

Table 5: Gold-continuation log score on HellaSwag, 2,000 questions, zero shot, ten held-out mixtures per scale, in nats/token; higher is better. *real* is the members pretrained at that mixture; *flow* is the leave-one-mixture-out generator. Δ columns are paired over the five mixtures of a stratum and give the mean difference and its t (df $= 4$), with $|t| \geq 2$ the separation threshold used throughout. *rank* 50 is the same real members' codes truncated to rank 50, the reconstruction floor, which involves no generator. The accuracy view of the same runs is Table 32.

	stratum	real	flow	flow − real (t)	flow − rank 50 (t)
Mini (51M)	interior	−5.198	−5.178	+0.0204 (+3.40)	+0.0074 (+1.62)
	vertices	−5.228	−5.140	+0.0877 (+0.77)	+0.0870 (+0.76)
Small (124M)	interior	−4.685	−4.648	+0.0360 (+4.27)	+0.0085 (+1.52)
	vertices	−4.771	−4.611	+0.1601 (+0.97)	+0.1597 (+0.97)
Medium (355M)	interior	−4.148	−4.141	+0.0068 (+0.62)	−0.0150 (−3.12)
	vertices	−4.273	−4.101	+0.1720 (+0.86)	+0.1720 (+0.86)

Table 7: All 85 held-out mixtures, one LOMO generator per cell, with rows indicating n samples/cell. *CRPS* equation 7 scores the whole cohort against the one real model at that mixture, which a round-trip error cannot do without being minimized by a point mass (Appendix B.7). Absolute round-trip errors are in Table 4.

	CRPS, %ΔPPL			retrieval − flow, %ΔPPL			marginal − flow, CRPS		
	flow	marginal	**skill**(↑)	gap	t	won	gap	t	won
Mini (51M, n=32)	3.10	8.34	**+0.628**	+1.70	+4.63	61/85	+5.23	+10.27	84/85
Small (124M, n=32)	3.26	8.38	**+0.611**	+1.62	+5.38	61/85	+5.12	+11.02	83/85
Medium (355M, n=16)	3.51	8.83	**+0.603**	+0.89	+3.13	43/85	+5.32	+12.44	82/85

4.2 Performance of Generated LLMs

Downstream tasks. We score HellaSwag at 2,000 questions on ten held-out mixtures per scale, five one-hot vertices and five interior, against the real members pretrained at those mixtures (Zellers et al., 2019). At these scales and a $20D$ token budget the models sit within a few points of chance and accuracy is quantized at $1/2000$, so the accuracy column can show parity without being able to distinguish it from a failure to measure (Table 32, Appendix B.13). We therefore read the log-probability assigned to the correct continuation, which is continuous, proper, and paired question by question (Table 5).

4.3 Diversity of Generated LLMs

With 80 of 85 mixtures observed once, π is nearly a unique key, so a generator could satisfy every conditional statistic by simply storing one point per mixture. Since no dispersion statistic can detect this (Appendix B.7), we instead use a structural control: a copier *is* 1-NN retrieval, and under matched blinding it can at best tie retrieval, never beat it.

Table 6: Next-token disagreement of generated models on *interior* pools (vertices are never pooled with the interior). Flow samples disagree with their target more than the rank-50 floor but less than 1-NN retrieval and other π, and less than original zoo members (Table 3). Vertex results are in Table 30.

Pool	Mini	Small	Medium
Flow vs. its target	0.1322	0.1331	0.1452
1-NN retrieval vs. target	0.1680	0.1596	0.1651
Flow vs. other π	0.2056	0.1954	0.1967
Rank-50 floor	0.0931	0.0891	0.0935
Flow cohort spread	0.0825	0.1018	0.1257

Conditioning on π recovers about three fifths of the available skill over the Zoo's π-blind marginal, flat across the ladder (Table 23), and the flow's exact likelihood correctly identifies the requested mixture among all 85 candidates for 94–97% of held-out members (Appendix B.8). Generated models therefore behave like models pretrained on the requested mixture, but respond to it less strongly than real models do: their perplexity slopes against the requested domain weight are only 42–54% of the real slopes (Table 29), and their codes are neither drawn from the same distribution as real codes nor spread with the correct conditional width.

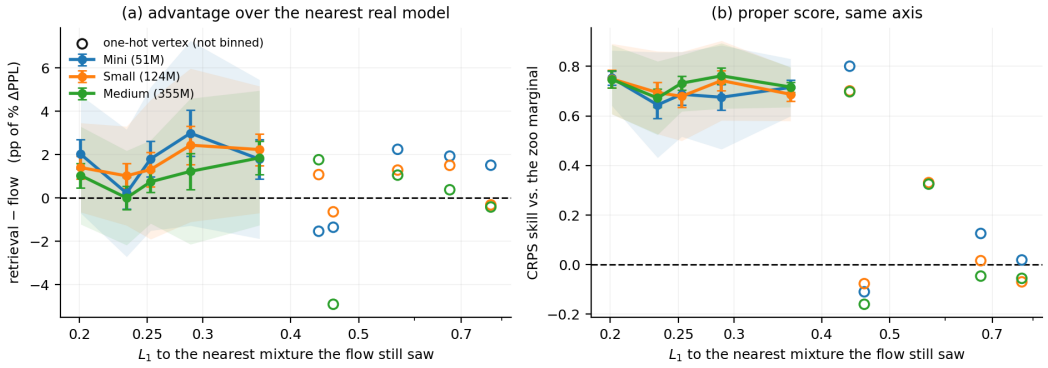


Figure 1: Advantage over blinded retrieval (a) and the proper score (b) against L_1 distance to the nearest mixture seen by the generator. Filled markers are equal-count bins over the 80 *interior* cells with binned standard errors. The five vertices are open squares and are excluded from binning.

4.4 Unseen Mixtures and Reduced Integration Steps

Over the 80 interior mixtures the margin over blinded retrieval is positive at every scale and attenuates from Mini to Medium (Table 22). Figure 1 shows both positive advantage across distance bins at all three scales and a steady CRPS skill of ~ 0.70. At the five one-hot vertices both quantities drop, with median skill below 0.15.

Table 8: An ablation on the number of Euler integration steps taken in generation of LOMO flows across all scales. Here round-trip error indicates the backward integration, followed by the reverse integration. Mean pairwise distances are reported as a dispersion reporting tool, additional metrics can be found in Table 26.

	round-trip rel. L_2			rel. L_2 vs. 200-step			mean pairwise dist.		
steps	Mini	Small	Med.	Mini	Small	Med.	Mini	Small	Med.
1	1.0077	0.7129	0.5826	0.4158	0.3786	0.3735	2.306	3.209	4.558
5	0.2413	0.1552	0.1208	0.0920	0.0715	0.0534	1.725	2.732	4.087
50	0.0281	0.0166	0.0122	0.0062	0.0050	0.0037	1.922	2.889	4.139
200	0.0070	0.0042	0.0030	0.0000	0.0000	0.0000	1.942	2.904	4.145

We find that our flow generator, is capable of quality generation in latent space in very few steps, despite the complexity of the structure. All other results hold the number of Euler integration steps to 50 unless explicitly stated otherwise.

5 Conclusion

We demonstrate that, in a controlled environment, the variance in pretrained language model weight spaces is structured by intentional variation in their pretraining data mixtures, and that this structure is resilient enough to permit our flow matching generative model of weights to generalize inside the simplex of data mixtures. We identify the performance boundaries of these generators, namely the vertices of that simplex, the noise floor of replication, and the baseline performance changes induced by a compression and decompression cycle. We find that generated models perform similarly to zoo members by latent and weight space measures, as well as by downstream task metrics, and that their performance persists through dramatic reductions in the number of generator integration steps. Our methods are limited by the scale of samples used, both in magnitude and population, as well as by the fact that the PCA compression basis is transductive, which limits opportunities for testing basis-blind generation. Finally, we contribute a large, diverse, and linearly mode connected model zoo trained under tight controls for application in weight space learning/generation and beyond. Our anonymized implementation is available here: A-Night-at-the-LLM-Zoo, and zoo artifacts will be published after peer review or on reviewer request.

AI USE STATEMENT

In this work, authors leveraged AI tools for the software implementation and adaptation of prior published and appropriately cited methods and the data analysis and charting software run on the produced artifacts. We did not use AI tools in the decision of methodology, supporting experiments, or analysis of results. Other required disclosure fields are not applicable to our work. We choose to disclose that AI tools were used in the formatting of tables and figures, and for drafting some portions of early manuscript versions. AI tools were also used to support the literature review and source discovery processes; however, authors have verified every cited source for correctness. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI, and will respond to queries regarding the provenance of artifacts, data, methods, or portions of the manuscript text itself.

REPRODUCIBILITY STATEMENT

To support the core scientific tenet of peer-reproducibility, we provide all software levered in the implementation and execution of this work. We support this by detailing relevant hyperparameters throughout the Appendix, including Appendix A.1 and Appendix A.2. Our anonymized implementation is available here: A-Night-at-the-LLM-Zoo, and zoo artifacts will be published after peer review or on reviewer request.

ETHICS STATEMENT

Since several large text corpora were used in training the zoo, we want to thank those who produced, compiled, cleaned, and made publicly available this data. We have cited their work appropriately, and have complied with any relevant licensing and usage terms. To the best of our knowledge, these open source datasets do not reveal human users' personal identifiable information or violate copyright restrictions.

REFERENCES

Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Leandro von Werra, and Thomas Wolf. Smollm - blazingly fast and remarkably powerful, 2024.

David M. Allen. The relationship between variable selection and data agumentation and a method for prediction. *Technometrics*, 16:125–127, 1974. URL https://api.semanticscholar.org/CorpusID:121351491.

Bruno Andreis, Bedionita Soro, Philip Torr, and Sung Ju Hwang. Set-based neural network encoding without weight tying. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=i3me9bCSCy.

Sylvain Arlot and Alain Celisse. A survey of cross-validation procedures for model selection. *Statistics Surveys*, 4(none), January 2010. ISSN 1935-7516. doi: 10.1214/09-ss054. URL http://dx.doi.org/10.1214/09-SS054.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pp. 2397–2430. PMLR, 2023.

Tony Bonnaire, Raphaël Urfin, Giulio Biroli, and Marc Mezard. Why diffusion models don't memorize: The role of implicit dynamical regularization in training. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=BSZqpqgqM0.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh,

Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

Rujikorn Charakorn, Edoardo Cetin, Yujin Tang, and Robert Tjarko Lange. Text-to-loRA: Instant transformer adaption. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=zWskCdu3QA.

Rujikorn Charakorn, Edoardo Cetin, Shinnosuke Uesaka, and Robert Tjarko Lange. Doc-to-loRA: Learning to instantly internalize contexts. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=iW1oBBO72S.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/69386f6bb1dfed68692a24c8686939b9-Paper.pdf.

CodeParrot. CodeParrot Clean. https://huggingface.co/datasets/codeparrot/codeparrot-clean. Hugging Face dataset, accessed September 25, 2026.

Gabriel Eilertsen, Daniel Jönsson, Timo Ropinski, Jonas Unger, and Anders Ynnerman. Classifying the classifier: dissecting the weight space of neural networks. *Proceedings of the European Conference on Artificial Intelligence (ECAI 2020)*, 325:1119–1126, 2020.

Damian Falk, Konstantin Schürholt, and Damian Borth. The impact of model zoo size and composition on weight space learning, 2025a. URL https://arxiv.org/abs/2504.10141.

Damian Falk, Konstantin Schürholt, Konstantinos Tzevelekakis, Léo Meynent, and Damian Borth. Learning model representations using publicly available model hubs, 2025b. URL https://arxiv.org/abs/2510.02096.

Ronald Aylmer Fisher. *Statistical Methods for Research Workers*. Oliver and Boyd, Edinburgh, 1925.

Jonathan Frankle, Gintare Karolina Dziugaite, Daniel Roy, and Michael Carbin. Linear mode connectivity and the lottery ticket hypothesis. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 3259–3269. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/frankle20a.html.

Ulrich Germann. Greedy decoding for statistical machine translation in almost linear time. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 72–79, 2003. URL https://aclanthology.org/N03-1010/.

Tilmann Gneiting and Adrian E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102:359 – 378, 2007. URL https://api.semanticscholar.org/CorpusID:1878582.

John C. Gower. Some distance properties of latent root and vector methods used in multivariate analysis. *Biometrika*, 53:325–338, 1966. URL https://api.semanticscholar.org/CorpusID:9968018.

Saumya Gupta, Scott Biggs, Moritz Laber, Zohair Shafi, Robin Walters, and Ayan Paul. Deepweightflow: Re-basined flow matching for generating neural network weights. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=fOwsr1VTi8.

David Ha, Andrew M. Dai, and Quoc V. Le. Hypernetworks. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=rkpACe1lx.

Xiaolong Han, Zehong Wang, Bo Zhao, Binchi Zhang, Jundong Li, Damian Borth, Rose Yu, Haggai Maron, Yanfang Ye, Lu Yin, and Ferrante Neri. A survey of weight space learning: Understanding, representation, and generation, 2026. URL https://arxiv.org/abs/2603.10090.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. URL https://openreview.net/forum?id=qw8AKxfYbI.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Oriol Vinyals, Jack W. Rae, and Laurent Sifre. Training compute-optimal large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Hao Huang, Jingtao Ding, Mengqi Liao, Xin Wang, Jinyang Ban, Yuan Yuan, Huaiyu Wan, and Yong Li. LoRAGen: Structure-aware weight space learning for loRA generation. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=mrafO7aTYj.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=6t0Kwf8-jrj.

Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization, 2019. URL https://arxiv.org/abs/1803.05407.

Boris Knyazev, Michal Drozdzal, Graham W Taylor, and Adriana Romero-Soriano. Parameter prediction for unseen deep architectures. In *Advances in Neural Information Processing Systems*, 2021.

Miltiadis Kofinas, Boris Knyazev, Yan Zhang, Yunlu Chen, Gertjan J. Burghouts, Efstratios Gavves, Cees G. M. Snoek, and David W. Zhang. Graph Neural Networks for Learning Equivariant Representations of Neural Networks. In *12th International Conference on Learning Representations (ICLR)*, 2024.

Jeffrey Li, Alex Fang, Georgios Smyrnis, Maor Ivgi, Matt Jordan, Samir Yitzhak Gadre, Hritik Bansal, Etash Kumar Guha, Sedrick Keh, Kushal Arora, Saurabh Garg, Rui Xin, Niklas Muennighoff, Reinhard Heckel, Jean Mercat, Mayee F Chen, Suchin Gururangan, Mitchell Wortsman, Alon Albalak, Yonatan Bitton, Marianna Nezhurina, Amro Kamal Mohamed Abbas, Cheng-Yu Hsieh, Dhruba Ghosh, Joshua P Gardner, Maciej Kilian, Hanlin Zhang, Rulin Shao, Sarah M Pratt, Sunny Sanyal, Gabriel Ilharco, Giannis Daras, Kalyani Marathe, Aaron Gokaslan, Jieyu Zhang, Khyathi Chandu, Thao Nguyen, Igor Vasiljevic, Sham M. Kakade, Shuran Song, Sujay Sanghavi, Fartash Faghri, Sewoong Oh, Luke Zettlemoyer, Kyle Lo, Alaaeldin El-Nouby, Hadi Pouransari, Alexander T Toshev, Stephanie Wang, Dirk Groeneveld, Luca Soldaini, Pang Wei Koh, Jenia Jitsev, Thomas Kollar, Alex Dimakis, Yair Carmon, Achal Dave, Ludwig Schmidt, and Vaishaal Shankar. Datacomp-LM: In search of the next generation of training sets for language models. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024. URL https://openreview.net/forum?id=CNWdWn47IE.

Zhiyuan Liang, Dongwen Tang, Yuhao Zhou, Xuanlei Zhao, Mingjia Shi, Wangbo Zhao, Zekai Li, Peihao Wang, Konstantin Schürholt, Damian Borth, Michael M. Bronstein, Yang You, Zhangyang Wang, and Kai Wang. Drag-and-drop llms: Zero-shot prompt-to-weights, 2025. URL https://arxiv.org/abs/2506.16406.

Derek Lim, Haggai Maron, Marc T. Law, Jonathan Lorraine, and James Lucas. Graph metanetworks for processing diverse neural architectures. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=ijK5hyxs0n.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Qian Liu, Xiaosen Zheng, Niklas Muennighoff, Guangtao Zeng, Longxu Dou, Tianyu Pang, Jing Jiang, and Min Lin. Regmix: Data mixture as regression for language model pre-training. *arXiv preprint arXiv:2407.01492*, 2024.

Xingchao Liu, Chengyue Gong, and qiang liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations*, 2023a. URL https://openreview.net/forum?id=XVjTT1nw5z.

Zhengzhong Liu, Aurick Qiao, Willie Neiswanger, Hongyi Wang, Bowen Tan, Tianhua Tao, Junbo Li, Yuqi Wang, Suqi Sun, Omkar Pangarkar, Richard Fan, Yi Gu, Victor Miller, Yonghao Zhuang, Guowei He, Haonan Li, Fajri Koto, Liping Tang, Nikhil Ranjan, Zhiqiang Shen, Xuguang Ren, Roberto Iriondo, Cun Mu, Zhiting Hu, Mark Schulze, Preslav Nakov, Tim Baldwin, and Eric Xing. Llm360: Towards fully transparent open-source llms. 2023b.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

manu. Project Gutenberg Dataset. https://huggingface.co/datasets/manu/project_gutenberg. Hugging Face dataset derived from Project Gutenberg, accessed September 25, 2026.

Casey Meehan, Kamalika Chaudhuri, and Sanjoy Dasgupta. A three sample hypothesis test for evaluating generative models. In Silvia Chiappa and Roberto Calandra (eds.), *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pp. 3546–3556. PMLR, 26–28 Aug 2020. URL https://proceedings.mlr.press/v108/meehan20a.html.

Aviv Navon, Aviv Shamsian, Idan Achituve, Ethan Fetaya, Gal Chechik, and Haggai Maron. Equivariant architectures for learning in deep weight spaces. *arXiv preprint arXiv:2301.12780*, 2023.

Behnam Neyshabur, Hanie Sedghi, and Chiyuan Zhang. What is being transferred in transfer learning? In *Advances in Neural Information Processing Systems*, volume 33, pp. 512–523. Curran Associates, Inc., 2020.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, Nathan Lambert, Dustin Schwenk, Oyvind Tafjord, Taira Anderson, David Atkinson, Faeze Brahman, Christopher Clark, Pradeep Dasigi, Nouha Dziri, Michal Guerquin, Hamish Ivison, Pang Wei Koh, Jiacheng Liu, Saumya Malik, William Merrill, Lester James V. Miranda, Jacob Morrison, Tyler Murray, Crystal Nam, Valentina Pyatkin, Aman Rangapur, Michael Schmitz, Sam Skjonsberg, David Wadden, Christopher Wilhelm, Michael Wilson, Luke Zettlemoyer, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. 2 olmo 2 furious, 2024. URL https://arxiv.org/abs/2501.00656.

Keiran Paster, Marco Dos Santos, Zhangir Azerbayev, and Jimmy Ba. Openwebmath: An open dataset of high-quality mathematical web text. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=jKHmjlpViu.

William Peebles and Saining Xie. Scalable Diffusion Models with Transformers . In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4172–4182, Los Alamitos, CA, USA, October 2023. IEEE Computer Society. doi: 10.1109/ICCV51070.2023.00387. URL https://doi.ieeecomputersociety.org/10.1109/ICCV51070.2023.00387.

William Peebles, Ilija Radosavovic, Tim Brooks, Alexei Efros, and Jitendra Malik. Learning to learn with generative models of neural network checkpoints. *arXiv preprint arXiv:2209.12892*, 2022.

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.

Guilherme Penedo, Hynek Kydlíček, Vinko Sabolčec, Bettina Messmer, Negar Foroutan, Amir Hossein Kargaran, Colin Raffel, Martin Jaggi, Leandro Von Werra, and Thomas Wolf. Fineweb2: One pipeline to scale them all — adapting pre-training data processing to every language. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=jnRBe6zatP.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. Film: visual reasoning with a general conditioning layer. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018. ISBN 978-1-57735-800-8.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

Mingxing Rao and Daniel Moyer. Generalization and memorization in rectified flow. In Paolo Favaro, Zuzana Kukelova, Atsuto Maki, Anna Rohrbach, Konrad Schindler, and Federico Tombari (eds.), *Computer Vision – ECCV 2026*, pp. 537–554, Cham, 2026. Springer Nature Switzerland. ISBN 978-3-032-37383-0.

Daniel Saragih, Deyu Cao, Tejas Balaji, and Ashwin Santhosh. Flow to learn: Flow matching on neural network parameters, 2025. URL https://arxiv.org/abs/2503.19371.

Konstantin Schürholt, Dimche Kostadinov, and Damian Borth. Self-supervised representation learning on neural network weights for model characteristic prediction. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=F1D8buayXQT.

Konstantin Schürholt, Boris Knyazev, Xavier Giró-i Nieto, and Damian Borth. Hyper-representations as generative models: Sampling unseen neural network weights. In *Thirty-Sixth Conference on Neural Information Processing Systems (NeurIPS)*, September 2022a.

Konstantin Schürholt, Diyar Taskiran, Boris Knyazev, Xavier Giró-i Nieto, and Damian Borth. Model zoos: A dataset of diverse populations of neural network models. In *Thirty-Sixth Conference on Neural Information Processing Systems (NeurIPS) Track on Datasets and Benchmarks*, September 2022b.

Konstantin Schürholt, Michael W. Mahoney, and Damian Borth. Towards scalable and versatile weight space learning. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*. PMLR, 2024.

Shayle R. Searle, George Casella, and Charles E. McCulloch. *Variance Components*. John Wiley & Sons, 1992. ISBN 9780471621621. doi: 10.1002/9780470316856.

Thibault Sellam, Steve Yadlowsky, Ian Tenney, Jason Wei, Naomi Saphra, Alexander D'Amour, Tal Linzen, Jasmijn Bastings, Iulia Raluca Turc, Jacob Eisenstein, Dipanjan Das, and Ellie Pavlick. The multiBERTs: BERT reproductions for robustness analysis. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=K0E_F0gFDgA.

Bedionita Soro, Bruno Andreis, Song Chong, and Sung Ju Hwang. Instruction-guided autoregressive neural network parameter generation. In *Workshop on Neural Network Weights as a New Data Modality*, 2025a. URL https://openreview.net/forum?id=QutFK34ea1.

Bedionita Soro, Bruno Andreis, Hayeon Lee, Wonyong Jeong, Song Chong, Frank Hutter, and Sung Ju Hwang. Diffusion-based neural network weights generation. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL https://openreview.net/forum?id=j8WHjM9aMm.

Bedionita Soro, Aoxuan Silvia Zhang, Bruno Andreis, Jaehyeong Jo, Song Chong, and Sung Ju Hwang. LS-merge: Merging language models in latent space. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=VSDV0SWwOC.

Thomas Unterthiner, Daniel Keysers, Sylvain Gelly, Olivier Bousquet, and Ilya Tolstikhin. Predicting neural network accuracy from weights, 2021. URL https://arxiv.org/abs/2002.11448.

Oskar van der Wal, Pietro Lesci, Max Müller-Eberstein, Naomi Saphra, Hailey Schoelkopf, Willem Zuidema, and Stella Biderman. Polypythias: Stability and outliers across fifty language model pre-training runs. In *The Thirteenth International Conference on Learning Representations*, 2025.

Kai Wang, xu Zhao Pan, Zhuang Liu, Zelin Zang, Trevor Darrell, and Yang You. Neural network diffusion, 2024. URL https://openreview.net/forum?id=8Q6UmFhhQS.

Kai Wang, Dongwen Tang, Wangbo Zhao, Konstantin Schürholt, Zhangyang Wang, and Yang You. Scaling up parameter generation: A recurrent diffusion approach. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=65llKR17s4.

Zhangyang Wang, Peihao Wang, and Kai Wang. Position: Weight space should be a first-class generative AI modality. In *Forty-third International Conference on Machine Learning Position Paper Track*, 2026. URL https://openreview.net/forum?id=vu0ousWLde.

Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 23965–23998. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/wortsman22a.html.

Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy Liang, Quoc V Le, Tengyu Ma, and Adams Wei Yu. Doremi: Optimizing data mixtures speeds up language model pretraining. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=lXuByUeHhd.

Jiasheng Ye, Peiju Liu, Tianxiang Sun, Jun Zhan, Yunhua Zhou, and Xipeng Qiu. Data mixing laws: Optimizing data mixtures by predicting language modeling performance. In *The Thirteenth International Conference on Learning Representations*, 2025.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence?, 2019. URL https://arxiv.org/abs/1905.07830.

Boya Zeng, Yida Yin, Zhiqiu Xu, and Zhuang Liu. Generative modeling of weights: Generalization or memorization?, 2025. URL https://openreview.net/forum?id=QbSeApdgfi.

Allan Zhou, Kaien Yang, Kaylee Burns, Adriano Cardace, Yiding Jiang, Samuel Sokota, J Zico Kolter, and Chelsea Finn. Permutation equivariant neural functionals. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=fmYmXNPmhv.

A Model Zoo Construction

A.1 Model Architecture

All members of all Zoos are GPT-2 (Radford et al., 2019), built from the Hugging Face `GPT2LMHeadModel` with no pretrained weights and no download (Table 9). The Small and Medium configurations are exactly the published GPT-2 configurations in both width and depth.

The embedding fraction of D falls by $\sim 3.4\times$ across scales because the transformer blocks grow with n_{embd}^2 while the embedding table grows linearly. One consequence recurs in every region-wise statistic: a whole-model figure is largely a statement about embeddings at Mini and largely a statement about transformer blocks at Medium. To account for this evolving ratio, we report statistics for both the blocks and embeddings when appropriate.

Table 9: Model scales. D counts every trainable scalar, embeddings and biases included, and every model is trained on $20D$ tokens (Hoffmann et al., 2022). Tokens per block parameter is the effective data budget on the transformer blocks alone; it swings $1.74\times$ across the ladder at a fixed 20 tokens per total parameter.

	Mini	Small	Medium
n_{embd}	512	768	1024
n_{layer}	8	12	24
n_{head}	8	12	16
D	51,475,968	124,439,808	354,823,168
embedding share of D	51.0%	31.6%	14.8%
tokens per block parameter	40.8	29.3	23.5
identical at every scale			
vocabulary	50,257		
context length n_{ctx}	1024		
n_{inner}	$4\,n_{\text{embd}}$		
activation	GELU		
embeddings	input/output tied		
LayerNorm ϵ	10^{-5}		
initialization	$\mathcal{N}(0, 0.02^2)$, residual projections $\times\, 1/\sqrt{2L}$		
dropout	0 (residual, embedding, attention)		
attention	fused scaled dot-product		

A.2 Training

All N zoo members at a given scale share a common initialization and a single data stream for a fraction $(1-\beta)S$ of the total step budget, S. We refer to this phase of training as the *trunk*. After $(1-\beta)S$ steps, the zoo population diverges by training on their own mixture π_i for the remaining βS steps - we call this phase the *branch*. The learning-rate schedule is computed over the global step count spanning both phases, so branches inherit the optimizer state of the terminal trunk step. Table 10 contains optimizer hyperparameters and Table 11 the resulting step splits.

A.3 Pretraining Data and Zoo Composition

A *class* is a pretraining data mixture $\pi \in \Delta^4$, a point on the simplex over the five text domains of Table 12. Each domain is one streamed dataset. A document is assigned to the held-out evaluation split by hashing its own text rather than by position: the corpora contain exact duplicate documents at nearby offsets, so a positional split would let a training document's duplicate leak into the evaluation set.

Other language models of similar sizes frequently train beyond the $20D$ tok/param ratio, for example SmolLM 135M Allal et al. (2024), train on over $50\times$ our $20D$ mark. The result is better downstream performance by accepting deviation from Hoffmann et al. (2022).

Table 10: zoo training hyperparameters, identical at all three scales.

Optimizer	AdamW (Loshchilov & Hutter, 2019)
β_{Adam}	$(0.9,\ 0.95)$
Weight decay	0.1, applied uniformly
Gradient clipping	global norm 1.0
Precision	bf16 mixed
Peak learning rate	6×10^{-4}
Warmup	linear, first 1% of steps
Decay	cosine to 10% of peak (Brown et al., 2020)
Batch size	16
Gradient accumulation	8
Tokens per step	131,072
Token budget	$20D$ (Hoffmann et al., 2022)
Trunk seed	1234 (fixed across scales)
Branch seed	$10{,}000 + i$

Table 11: Trunk and branch step split. Counts are exact. $\beta = 0.30$ exists only as the Mini-scale calibration point of Appendix A.4.

Scale	β	total	trunk	branch
Mini	0.15	7,854	6,676	1,178
Mini	0.30	7,854	5,498	2,356
Small	0.15	18,988	16,140	2,848
Medium	0.15	54,142	46,021	8,121

Composition. Each zoo has $N = 100$ members over 85 distinct mixtures: 20 *anchors*, the five one-hot vertices of Δ^4 replicated over four independent branches, and 80 *singletons* drawn i.i.d. $\text{Dirichlet}(1)$ and rejection-sampled at a minimum pairwise L_1 gap of 0.15. On the realized drawing, the closest singleton pair is at $L_1 = 0.173$ and the median nearest-neighbor distance is 0.251. The anchor replicates are used to capture the noise floor from repeated training for a given π_i. Anchor replicates are exempt from the gap criterion by construction.

The mixture plan is drawn once and reused, so member i carries an identical π_i at all three scales. The three Zoos are consequently the same experiment at three sizes rather than three experiments, and a difference between scales is not a difference in mixtures. As such, the scale tiers do not represent independent replications in the typical sense.

A.4 Choosing the Branching Fraction

β was calibrated at the Mini scale ($\sim$ 51M), where a full $N = 100$ zoo was affordable twice-over, against the two candidate values $\beta = 0.15$ and $\beta = 0.30$. The selection criterion was to find the smallest value β that passes the domain-separation gate below, with the reasoning that β buys conditioning signal at the price of compute cost. Both values passed, so $\beta = 0.15$ was chosen in the interest of compute resource efficiency. Only $\beta = 0.15$ was used for the Small and Medium populations, and no $\beta = 0.30$ results are included in the main body of this manuscript.

A smaller sweep of $\beta \in \{0.15, 0.30, 0.60\}$ at the Mini scale with $N = 12$ per setting was run first, to bracket the range before committing a full population; its weight-space geometry is Table 13.

Mixture identity is checked by scoring every member on held-out text from all five domains, done on 64 held-out blocks per domain, context length 1024, and with evaluation RNG seed chosen independently of the training-mixture seeds. The domain separation gate is passed when anchor groups score well in their own domain and the variance between anchor groups exceeds the variance of the groups themselves. This is a loose proxy for the signal-to-noise ratio (SNR) of π. This criterion alone is maximized by memorization or the production of degenerate distributions of weights,

Table 12: Pretraining data domains. †Python-only because `codeparrot-clean` is Python-only. ‡French only, one `fineweb-2` per-language configuration. Evaluation split is SHA-1 of the UTF-8 document bytes, one in 64.

Domain	Dataset	Config	Column
web	`HuggingFaceFW/fineweb-edu` (Penedo et al., 2024)	`sample-10BT`	text
code†	`codeparrot/codeparrot-clean` (CodeParrot)	—	content
math	`open-web-math` (Paster et al., 2024)	—	text
books	`manu/project_gutenberg` (manu)	—	text
multilingual‡	`HuggingFaceFW/fineweb-2` (Penedo et al., 2025)	`fra_Latn`	text

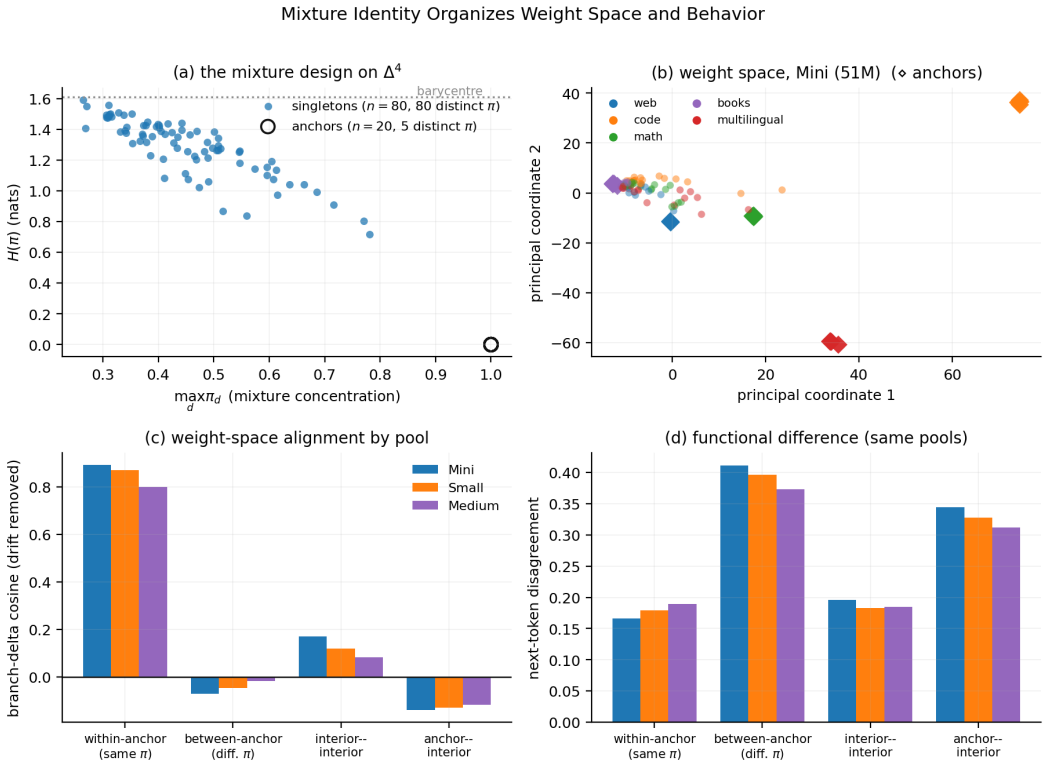


Figure 2: **(a)** The mixture samples shown as entropy, $H(\pi)$ vs the maximum component-wise concentration in π_i; the design alone, on Δ^4, with no weights involved. **(b)** The Mini population by top-2 Gram-PCA components, colored by the $argmax\ \pi$ category. **(c)** The four disjoint pairwise pools of Table 1 in full $\mathbb{R}^D$ weight space. Cosine similarities are read off the displacement Gram equation 2 and are therefore exact at every coordinate, without rank truncation. **(d)** The same four pools in function space, as next-token disagreement (see Table 3).

and as such it was considered alongside a displacement score w.r.t the trunk artifact (introduced in Sec 3.5.1), reported against β in Table 13. We therefore select $\beta = 0.15$, which is balanced against reasonable trunk displacement and significant compute savings for zoo training and further experiments. Section 4.1 reports the diversity of the population this produced.

A.5 Common Trunks Canonicalize by Proxy

The branching fraction, β, also decides whether canonicalization is needed. At $\beta \to 1$ the members are independently trained runs sharing only an architecture, which is the regime in which weight-space symmetry concerns become unavoidable: independently trained networks could occupy distinct but functionally equivalent basins within the loss landscape, related by permutations of hidden units. A coordinate-wise population statistic would then mix mixture-induced variation with permutation

Table 13: Weight-space geometry against β, measured at Mini with $N = 12$. The between/within distance ratio is what a PCA basis is built from, and the largest β is the worst of the three on it.

	$\beta = 0.15$	$\beta = 0.30$	$\beta = 0.60$
between/within distance ratio	**4.08**	3.98	2.88
displacement from trunk	16.5%	28.7%	55.9%

artifacts, and a basis fitted on the population would inherit that potentially contaminated mixture. At $\beta \to 0$ the members are one model plus small terminal perturbations, and the mixture cannot have influenced the weights at all. Neither limit supports the line of inquiry this paper pursues.

An intermediate β is not a compromise between those failure modes, but an avoidance of both. The mechanism is the shared trunk itself, as all N members enter the branch phase from identical parameters and only diverge over the final βS steps. Thus the population is generated within a single basin by construction, not aligned into one afterwards. The alternative is to branch early or not at all and then re-basin the population before fitting a basis, as prior weight-space generative work has done (Gupta et al., 2026); permutation alignment has been found to buy little for models descended from a shared initialization (Neyshabur et al., 2020), and networks that share the first phase of training are linearly mode connected thereafter (Frankle et al., 2020); the shared trunk is the strongest available form of that condition.

Following Frankle et al. (2020) we score the linear interpolant $\theta(\alpha) = (1 - \alpha)\theta_i + \alpha\theta_j$ and report the excess of its loss over the chord, maximized over the *interior* of the path. The Linear Mode Connectivity (LMC) barrier is negative at every scale and in both pools: every interpolant lies at least 0.0067 nats *below* the chord (Table 14), which empirically justifies the decision not to canonicalize the zoo model parameters. Figure 3 draws the whole path over 40 scored pairs at each scale so total 120 scored pairs, no interpolant rises above the chord anywhere.

A single loss basin does *not*, however, mean that the population is interchangeable within it. Table 14 separates the two: connectivity is scale-stable, but whether *averaging* two members helps is not, and between different mixtures it reverses by 355M, where the midpoint beats both endpoints in only 5 of 20 pairs.

Table 14: Linear-interpolation barrier, in nats of mean-over-domain log perplexity, maximized over the interior of the path. Negative values mean the interpolated model is *better* than the chord, so there is no barrier at any scale. The *midpoint − best endpoint* row is the model-averaging statement and is independent of the barrier.

	Mini		Small		Medium	
	within	between	within	between	within	between
worst interior barrier	−0.0067	−0.0669	−0.0105	−0.0642	−0.0161	−0.0271
mean dip below chord	−0.0165	−0.1568	−0.0239	−0.1587	−0.0353	−0.1251
midpoint − best endpoint	−0.0100	−0.0919	−0.0177	−0.0283	−0.0305	**+0.0541**
midpoint beats both endpoints	19/20	18/20	20/20	15/20	20/20	**5/20**

B Weight-Space Representation

B.1 Dual Gram PCA

The representation is the dual (Gram) PCA of Gupta et al. (2026). Each member is flattened to $\theta_i \in \mathbb{R}^D$ under a fixed layout, the population is centered as $\widetilde{\mathbf{W}}$, and the model-space Gram $\mathbf{G} = \widetilde{\mathbf{W}}^\top \widetilde{\mathbf{W}} \in \mathbb{R}^{N \times N}$ is eigendecomposed into $\mathbf{U}_k$, $\mathbf{\Lambda}_k$, giving codes

$$\mathbf{C} \;=\; \mathbf{U}_k \mathbf{\Lambda}_k^{1/2} \in \mathbb{R}^{N \times k}. \tag{8}$$

No $D \times k$ basis is ever materialized, and decoding a code is a weighted sum over the ensemble members themselves, which is what makes the representation affordable at $D \sim 10^8$. Remaining settings

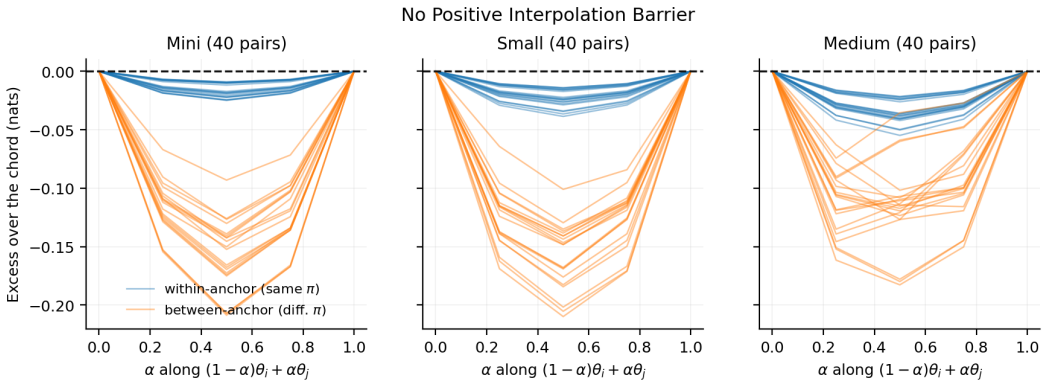


Figure 3: Excess of the interpolant's loss over the chord, in nats of mean-over-domain log perplexity, for all 40 scored pairs at each scale. Members are interpolated in the full $\mathbb{R}^D$ weight space, $(1-\alpha)\theta_i + \alpha\theta_j$ on raw weights. Chord endpoints are excluded from reporting.

are in Table 15. Biases are retained because GPT-2 carries one on every projection, and excluding one-dimensional parameters would make a generated model inherit some particular member's biases rather than generate them.

Table 15: Weight-space representation hyperparameters.

Rank k, population statistics	$N - 1 = 99$
Rank k, generative experiments	50 (Appendix B.2)
One-dimensional parameters	retained ($\sim 0.1\%$ of D)
Code normalization	one global scalar to unit training RMS
Synthetic weight-space noise	none

B.2 Why $k = 50$ for Generative Experiments

Every generative experiment departs from the $k = N - 1$ default and uses $k = 50$, because at the generator width we commit to it converged far faster. On an $8,000$ epoch sweep at the Mini scale, shown below in Table 16, $k = 50$ recovered more structure than $k = 99$ in the calibration case (Mini-scale).

Table 16: Conditional recovery at width $(64, 128, 64)$, Mini scale, varying only rank and duration; 100% is a correct conditional. The only $k = 99$ configuration that converges used a network seven times wider, which Appendix B.6 shows to be collapse-prone.

	500 epochs	8,000	20,000
$k = 50$	27.6%	90.2%	$\mathbf{101.1}\%$
$k = 99$	1.8%	21.7%	—

The truncation of k comes with a cost on reconstruction quality, measured in Table 17 as $\sim 2\ \Delta\mathrm{PPL}\%$ across scales.

B.3 The Representation Is Transductive

The dual basis is spanned by the zoo and is fitted on all N members, including any member later used for evaluation. Holding out these members from the basis fitting as well dramatically worsens their displacement relative to included members. The net effect is that the statistical measures become more likely to show the flow generator as memorizing. Developing a compression method that is comparably scalable but free of this weakness remains a challenge for future work.

Table 17: Round-trip error decomposed into what the representation permits and what the generator adds, %ΔPPL over 85 held-out cells.

	Mini	Small	Medium
flow round-trip	4.05	4.17	5.11
− truncation floor	1.80	1.93	2.46
= generator's own excess	**2.25**	**2.24**	**2.65**

B.4 Structure and Noise Inside the Models

The mixture nearly determines where a model sits in the PCA latents component coefficient (or 'code') space, but the map from π to that location is strongly nonlinear. On the leading five components, π explains $R^2 = 0.568$ of the variation linearly, 0.786 once the scalar $H(\pi)$ is added, and 0.948 under a full quadratic basis. A closed-form quadratic ridge regression therefore reaches $R^2 = 0.948$ at no training cost, which is why its decoding to weights is the baseline every generative result in this paper is reported against.

That baseline is only well posed because conditional *location* and conditional *width* occupy different parts of the spectrum (Table 18). The leading five components carry 84% of the retained trace and 0.4% of the within-mixture variance; components c5–c24 carry 80% of the within-mixture variance and almost none of the trace. Per anchor group the separation is sharper still, and the components carrying a group's within-variance differ by group: each anchor's four branches diverge along a nearly private set of directions, which is what member-specific noise looks like once it has been given its own principal axes.

Table 18: Code structure at $k = 99$, pooled over the anchor groups. The rank must be quoted with this table, because the λ shares are shares of the *retained* trace. "within/ens." compares the twenty anchors' within-group variance against all 100 members' total, so it exceeds 1 wherever the anchor groups are more dispersed than the ensemble. η^2 is the anchors-only between-group share, bounded in $[0, 1]$ by construction; five one-hot groups saturate a one-way ANOVA on Δ^4, so read it as a statement that the within-mixture noise is small in that block, not as evidence that π acts linearly — the $R^2(\pi)$ column measures that.

Block	λ share	within/ens.	of all within	η^2	$R^2(\pi)$
c0–c3	0.803	0.001	0.4%	0.9998	0.578
c4	0.038	0.004	0.1%	0.9934	0.368
c5–c9	0.036	0.969	15.0%	0.4390	0.182
c10–c24	0.036	4.178	65.2%	0.0283	0.013
c25+	0.087	0.514	19.4%	0.0144	0.001

We therefore read conditional location on c0–c4, conditional width on c5–c24, and score on c0–c24. Components beyond c24 are excluded from scoring: they behave as member-private noise that no generator can or should reproduce, and retaining them lets tail mismatch dominate the comparison. Statistics are reported in blocks rather than per component because individual components are identified only where the eigenvalue gaps are large — the third and fourth components form a near-degenerate pair whose axes rotate between runs.

B.4.1 Task Arithmetic Baseline

As used in Table 4, the *model soup* linear baseline described in Sections 3.5.2 can be written as,

$$\theta_{\text{soup}}(\pi) \;=\; \theta_{\text{trunk}} \;+\; \sum_{j=1}^{5} \pi_j \, \bar{d}_j, \qquad \bar{d}_j = \tfrac{1}{4} \sum_{r=1}^{4} d_{jr}, \tag{9}$$

which fits nothing at all- it is simply task arithmetic over the five vertex means (Ilharco et al., 2023). Additional measurements are elaborated on in Appendices B.8 and B.12 (Meehan et al., 2020).

Table 19: Member spectrum by coordinate region. Absolute effective rank equation **??**; k is stated because the normalized variant is not k-invariant. "var. share" is the region's share of the total centered variance, which is the composition confound in one number — compare it to the region's share of D in Table 2.

	Mini (8 blk)		Small (12 blk)		Medium (24 blk)	
region, $k = 99$	eff. rank	var. share	eff. rank	var. share	eff. rank	var. share
whole stack	4.466	100%	5.637	100%	8.725	100%
blocks only	7.198	13.4%	8.872	35.8%	11.346	64.7%
embeddings only	**4.026**	86.6%	**4.084**	64.2%	**4.961**	35.3%
same eigenvalues, embeddings only, read at lower k:						
$k = 49$	3.669		3.709		4.419	
$k = 24$	3.449		3.465		4.085	

B.5 Supplemental LOMO Results

A flow of this size 85 samples of 50-dimensional latents trains quickly, allowing complete control over the generators training data in experiments. The design is LOMO (see Section 3.5.2). This holdout does not impact the PCA basis, which is fit to all samples.

Table 20: LOMO predictive R^2 of the branch delta on π equation 6, 85 folds over 100 members: **interior** scores the 80 singleton folds, **vertex** the 5 anchor folds; higher is better within $[-1, 1]$. Embedding and function-space regions are in Table 21.

		Mini		Small		Medium	
region	design	interior	vertex	interior	vertex	interior	vertex
whole model	π	−0.0874	−0.1919	0.0539	−0.2537	0.0297	−0.2815
	$\pi, H(\pi)$	0.0560	0.2209	0.1576	0.1874	0.0979	0.1002
	π quadratic	0.3251	0.1170	0.3420	−0.0043	0.2174	−0.2518
blocks only	π	0.0463	−0.0959	0.0295	−0.1354	0.0355	−0.2348
	$\pi, H(\pi)$	0.1079	0.1940	0.0846	0.1563	0.0847	0.1162
	π quadratic	0.2433	−0.0818	0.1871	−0.2009	0.1488	−0.3777

Table 21: The embedding and function-space regions of Table 20, on the same 85 folds and the same three designs. The embedding region is where the quadratic surface does best — it is the only region reaching $R^2 > 0.40$ anywhere — and function space is where π alone does worst.

		Mini		Small		Medium	
region	design	interior	vertex	interior	vertex	interior	vertex
embeddings	π	−0.1217	−0.2036	0.0728	−0.3084	0.0145	−0.3480
	$\pi, H(\pi)$	0.0427	0.2242	0.2142	0.2017	0.1331	0.0772
	π quadratic	0.3461	0.1413	0.4619	0.0867	0.4009	−0.0721
function space	π	0.0362	−0.4358	0.0073	−0.4089	0.0025	−0.4004
	$\pi, H(\pi)$	0.0900	0.1125	0.0577	0.0754	0.0434	0.0325
	π quadratic	0.1443	−0.4932	0.1129	−0.4811	0.0703	−0.6571

Because the vertex mixtures (anchors) are four-fold replicated, we can create five extrapolation points by holding out one such vertex from the flow's training. This is because the we not only target the four removed anchors, but also target their mean and spread as a group - this is the fifth point. By systematically repeating this for the vertecies of Δ^4, we can estimate the relative difficulties of each axis.

Table 22: The difficulty axis. Rows one to three are rank terciles of the L_1 distance from a request to the nearest mixture its generator still saw, cut on Mini and reused verbatim at every scale so the same mixtures fall in the same row; the boundaries are 0.240 and 0.291. Rows four and five split the same 85 mixtures by *type* instead, which the terciles cannot do. "gap" is retrieval minus flow, so positive favours the generator.

	Mini (51M)			Small (124M)			Medium (355M)		
stratum	flow	retr.	gap	flow	retr.	gap	flow	retr.	gap
lower tercile ($n = 29$)	2.60	3.63	+1.03	2.21	3.39	+1.18	3.20	3.72	+0.52
middle tercile ($n = 29$)	2.38	3.94	+1.56	2.48	3.61	+1.13	3.24	3.66	+0.41
upper tercile ($n = 27$)	7.40	9.96	+2.55	8.10	10.71	+2.61	9.18	10.99	+1.81
interior ($n = 80$)	2.66	4.43	+1.77	2.57	4.26	+1.68	3.39	4.37	+0.98
one-hot vertices ($n = 5$)	26.27	26.83	+0.56	29.79	30.38	+0.59	32.65	32.23	-0.42

Table 23: All 85 held-out mixtures with one LOMO generator per cell, producing 32 samples per cell at Mini and Small and 16 at Medium. CRPS equation 7 scores the whole cohort against the one real model at that mixture (Appendix B.7); skill is CRPS against the π-blind marginal. Absolute round-trip errors are in Table 4.

	CRPS, %ΔPPL			retrieval − flow, %ΔPPL			marginal − flow, CRPS		
	flow	marginal	**skill**	gap	t	won	gap	t	won
Mini (51M)	3.10	8.34	**+0.628**	+1.70	+4.63	61/85	+5.23	+10.27	84/85
Small (124M)	3.26	8.38	**+0.611**	+1.62	+5.38	61/85	+5.12	+11.02	83/85
Medium (355M)	3.51	8.83	**+0.603**	+0.89	+3.13	43/85	+5.32	+12.44	82/85

Vertex error is roughly ten times the interior's, and the vertices are also the most distant cells, so they pile into the upper tercile and a subset balanced on distance to the nearest surviving mixture can still over-represent them badly.

All three designs regress the branch delta $\Delta_i = w_i - w_{\text{trunk}}$ on features of π with no intercept, since $\sum_d \pi_d = 1$ means π already spans the constant: *linear* uses the five mixture weights; $+H(\pi)$ appends the single scalar $H(\pi) = -\sum_d \pi_d \log \pi_d$; *quadratic* uses the fifteen products $\{\pi_j \pi_k\}_{j \le k}$, which contain the linear span, so the three are nested.

Each is fitted by ordinary least squares and LOMO scored: every member sharing a π leaves together, so an anchor's four replicates cannot predict one another. The baseline is the training-set mean, refit per fold, which is why R^2 is bounded above by 1 but can go negative when a design predicts a held-out mixture worse than the mean of the ones it saw.

B.6 The Conditional Generator

B.6.1 Architecture and Training

The generator is a conditional rectified flow (Liu et al., 2023a; Lipman et al., 2023) over the normalized codes of Appendix B. Training pairs a Gaussian source z_0 with a data code z_1, takes the linear interpolant $z_t = (1 - t)z_0 + tz_1$ at $t \sim U(0, 1)$, and regresses the velocity field $v_\psi(z_t, t, \pi)$ onto the constant target $u = z_1 - z_0$; sampling integrates v_ψ from $t = 0$ to $t = 1$ by explicit Euler. The mixture enters through a two-layer MLP whose output is concatenated to the network input, the encoder-MLP pattern of FiLM (Perez et al., 2018), and a learned null token with conditioning dropout gives classifier-free guidance (Ho & Salimans, 2021). Table 24 gives the hyperparameters.

Despite the small size of the generator, it remains performant in part due to the high quality compression of weights into $\mathbb{R}^k$ (Gupta et al., 2026).

Since Table 25 shows that sample dispersion and conditional widths are not simply related, we rank solutions by the proper CRPS statistic and use $|1 - a_\pi|$ alongside a classifier two-sample test as guards. a_π is two-sided, since over-fitting on a real zoo could appear as a_π falling *below* 1 despite the pooled ratio still seeming healthy.

Table 24: Conditional flow hyperparameters. The generator is deliberately small: it is what makes an 85-cell leave-one-mixture-out sweep affordable, since each cell needs an independently trained flow.

Type	rectified flow (Liu et al., 2023a; Lipman et al., 2023)
Width	$(64,\ 128,\ 64)$
Parameters	$\sim$51k
Integration	50 explicit Euler steps
Conditioning	2-layer SiLU MLP, concatenated (Perez et al., 2018)
Conditioning dropout	0.15 (Ho & Salimans, 2021)
Guidance scale at evaluation	1.0
Source scale σ_0	1.0
Optimizer	Adam
Learning rate	3×10^{-4}
Weight decay	10^{-4}
Batch size	64
Epochs	20,000, always run in full; Table 16 sweeps it
Checkpoint selection	best validation loss, 15% of rows held out
Seed	1234
Training time	$<$ 2 minutes, CPU

Table 25: Flow source width, σ_0, sweep at established hyperparameter values (see Table 24). a_π is the sampled spread at one mixture over that mixture's own within-group spread ($[0, 1]$, higher is not always better). Column "pooled" is the corresponding ratio against the whole training set - as such they cannot be cleanly compared to the LOMO generators in Appendix **??**.

	Mini		Small	
σ_0	a_π	pooled	a_π	pooled
0.125	0.316	0.950	0.591	0.992
0.25	0.521	0.966	0.485	0.952
0.5	**0.877**	0.983	**0.868**	0.999
0.75	0.774	0.983	0.908	0.990
1.0 (committed)	0.707	0.976	0.681	0.966
2.0	0.712	0.959	0.743	0.969

Modulating every block rather than the input alone, in the manner of adaLN-Zero (Peebles & Xie, 2023), is worse than concatenation at our chosen width and better than it only with a wider residual stream. None of these improve measured capability. The decided configuration was selected on functional CRPS, and a_π is reported as a collapse guard only.

B.6.2 Step Ablations

B.7 Admissible Scores

As discussed throughout Section 3, a scoring rule is *proper* if its expected value is optimized by reporting the true conditional, and *strictly* proper if only by reporting it (Gneiting & Raftery, 2007). The obvious Mean Squared or Mean Absolute Errors (MSE or MAE) fail to rise to this bar, as they can be minimized by point-mass memorization on the mean alone. This is again the motivation for the CRPS score equation 7, which cannot be minimized by a point mass. To support CRPS and CRPS skill, we report round-trip and functional error measures.

B.8 Likelihood and Identification

A divergence between the real and predicted code distributions cannot be convincingly estimated from samples at our N: the k-nearest-neighbour estimator returns a negative value on these codes, which is impossible for a non-negative quantity. A continuous normalizing flow has an *exact* likelihood

Table 26: Small model scale ablation of varied Euler integration steps with 4 leave-one-mixture-out flows (32 samples). *Round-trip error* integrates a real latent backwards ($1 \to 0$) then forwards again ($0 \to 1$) and measures how far it lands from its start. Real latents should sit around 1 RMS, and this value is included to detect collapsed flows. This table expands on Table 8 with additional metrics and dispersion checks.

	round-trip error		vs. 200-step sample		dispersion	
steps	rel. L_2	$\times n$	rel. L_2	cosine	sample RMS	mean pair dist.
1	0.5902	0.590	0.2772	0.9617	1.050	4.432
2	0.3116	0.623	0.1346	0.9911	1.062	4.046
5	0.1272	0.636	0.0464	0.9989	1.058	3.982
10	0.0633	0.633	0.0217	0.9998	1.055	3.996
25	0.0252	0.630	0.0079	1.0000	1.054	4.011
50	0.0126	0.629	0.0034	1.0000	1.053	4.017
200	0.0031	0.628	0.0000	1.0000	1.053	4.022

Table 27: Small: all 85 LOMO cells, 8 samples per cell with identical seeds. a_π is the conditional dispersion ratio, where 1 is correct and < 1 is over-tight. We see that generated models survive drastic reductions in Euler integration steps in function space.

steps	ΔPPL %	CRPS %	CRPS skill	a_π
1	4.365	3.134	0.6259	0.839
10	4.165	3.252	0.6119	0.673
50	4.173	3.231	0.6143	0.668

instead, calculated by integrating along the path (Chen et al., 2018; Lipman et al., 2023),

$$\log p_\psi(z_1 \mid \pi) \;=\; \log p_0(z_0) \;-\; \int_0^1 \operatorname{tr}\big(\partial v_\psi / \partial z\big)\, dt, \tag{10}$$

evaluated with the exact Jacobian trace and cross-checked against the discrete-map log-determinant.

From this construction, two clean properties arise: firstly, a direct connection between held-out log-likelihoods and those of the zoo population through cross-entropy, and secondly, it allows direct rankings of candidates by $\log p_\psi(z \mid \pi)$. In both of these settings, π-blind approaches or ablations provide lower bounds to interpret against.

Table 28: Code-space density at $k = 50$. The mutual-information bound compares against a matched twin — same width, budget and source scale, conditioning ablated — so the difference is not a capacity comparison. Identification ranks the true π among all 85 candidates. The C2ST is quantised at $1/41$ on its held-out split, so the three scales are identical, at $40/41$.

	Mini	Small	Medium
$I(z;\pi)$ lower bound vs. matched twin (nats)	+18.5	+31.9	+4.9
identification, top-1 (chance 1.2%)	**96.0%**	**94.0%**	**97.0%**
identification, top-5	99.0%	98.0%	99.0%
median rank of true π (of 85)	1.0	1.0	1.0
tie fraction	0.000	0.000	0.000
C2ST, joint (z, π)	0.976	0.976	0.976
NN gen→train / train→train	0.802	0.730	0.935

Table 28 uses these facts to show that mixture identification appears scale-stable. Across candidates and scales, these flows identify the correct held out mixture $\sim 95\%$ of the time against a chance rate of about 1%. However, because ill fitted models may be able to inflate the information bound, we only refer to this result as a sign indicator. C2ST refers to a classifier two-sample test, where 0.50

represents that the generated and true distributions are truly indistinguishable. The CRPS and CRPS skill measures reported throughout also tell a similar story in the context of the functional similarity results, indicating that our generators are at least learning the conditional mean, but may not be fully capturing the variance. With this said, the prior result in Table 4 showing that our flows outperform even first-nearest neighbor lookups stands. Better capturing this variance remains an open problem.

B.9 The Functional Round-Trip

While latent and weight space measures and statistics can be deeply informative, they ultimately cannot truly capture functional performance on downstream application. To show that the generated models survive up projection into $\mathbb{R}^D$, we sample generated latents, decode them, and use the produced weights to populate a GPT-2 model of the appropriate size. We then measure test-set perplexity across five domain samples, as well as downstream performance on HellaSwag (Zellers et al., 2019; Radford et al., 2019). The relevant baselines are the replicate noise floor, and the rank-truncation floor (see Table 4).

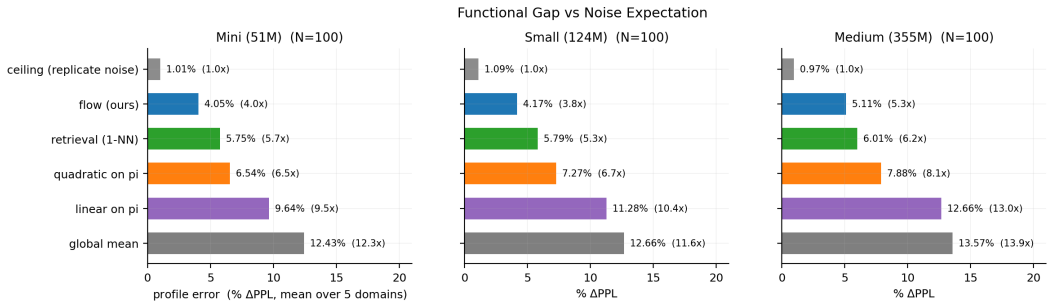


Figure 4: Table 4 on one axis per scale, with each bar's multiple of the replicate ceiling beside it. Every bar is a function-space quantity, $\%\Delta\mathrm{PPL}$ over the five-domain profile, but the arms reach it differently: the flow samples a code in z and decodes it to $\mathbb{R}^D$ before evaluation, while the π-surfaces, the global mean and retrieval predict the perplexity profile from π directly and never materialize weights. The shape of the ladder holds across a $6.9\times$ parameter range; what moves is the gap between the flow and retrieval.

The retrieval task along the barycenter-to-vertex traverse is discrete, with discontinuities indicating steps between π_i mixtures by 1-NN. Figure 5 shows this, and the contrast between 1-NN's discrete steps and the flows continuous progress is clear.

B.10 Dose-Response

Table 29: Ordinary least-squares slope of log perplexity on domain d against the requested π_d, with the Pearson correlation in parentheses, for the real zoo and for the generated cohort on the same requests. The final row is the mean over domains of the generated slope as a fraction of the real one.

	Mini (51M)		Small (124M)		Medium (355M)	
domain	real zoo	flow	real zoo	flow	real zoo	flow
web	-0.633	-0.328	-0.754	-0.352	-0.821	-0.325
code	-0.635	-0.401	-0.615	-0.349	-0.618	-0.334
math	-0.744	-0.458	-0.804	-0.436	-0.861	-0.414
books	-0.361	-0.083	-0.431	-0.063	-0.465	-0.058
multilingual	-1.056	-0.720	-0.919	-0.535	-0.895	-0.512
mean fraction of real slope	1.00	**0.54**	1.00	**0.46**	1.00	**0.42**

Dose-response curves indicate that across the web, code, math, and multilingual datasets the flows capture a significant proportion of the theoretically possible response to the dose of that dataset. The books category shows low response, which could be due to overlaps with other data classes. Overall,

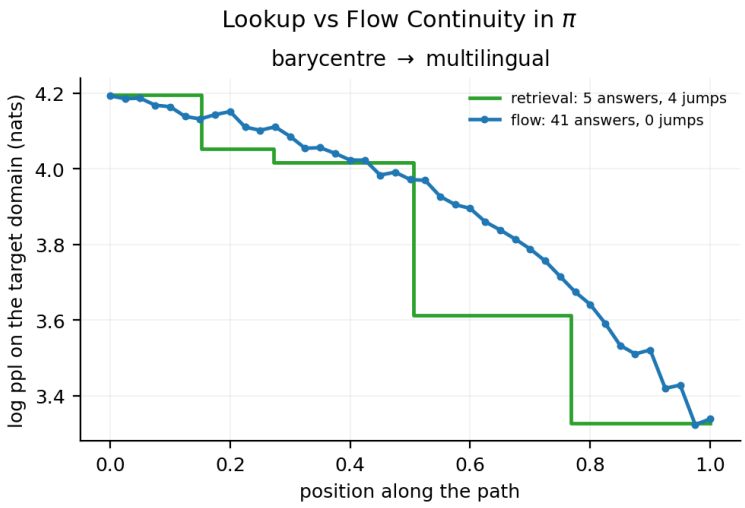


Figure 5: One traverse from the barycentre of Δ^4 to the multilingual vertex, 41 requests at two samples each, with samples at a request averaged before plotting. The abscissa traverses Δ^4, the ordinate is function space, and each of the flow's answers is a distinct code in z decoded to $\mathbb{R}^D$ before evaluation. Retrieval returns five distinct real members over the whole traverse; the generator returns 41 distinct models, and both arrive at very nearly the same place.

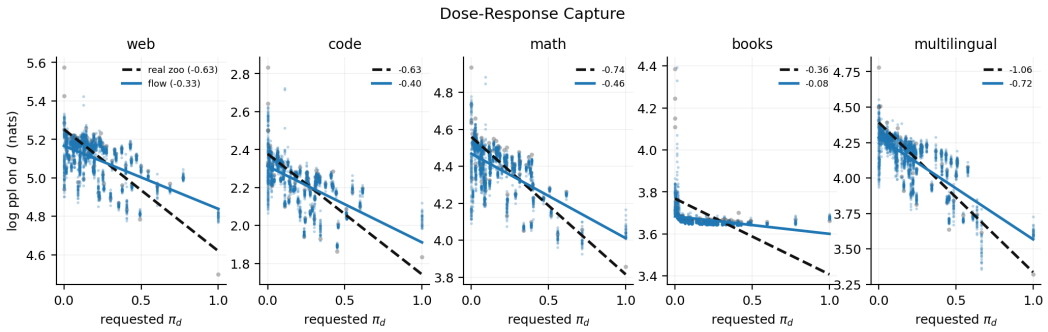


Figure 6: The dose–response of Table 29 at Mini, measured entirely in function space: a code is sampled in z at each requested π and decoded to $\mathbb{R}^D$ before any perplexity is taken. Faint points are the generated cohort (32 samples at each of 85 mixtures) with the real zoo's per-mixture profiles behind them; the lines carry the slopes of Table 29. The real slope is fitted per member while the plotted real points are per-mixture means, since an anchor's four replicates carry leverage their mean does not.

results show that the flows capture a significant proportion of the possible response to $\pi \in \Delta^4$ variations.

B.11 Next Token Prediction Disagreement

Here we provide auxiliary results studying the next-token prediction disagreement rates between generated models.

B.12 Memorization

With 100 members in a manifold of effective dimension ~ 4.5, and 80 of 85 mixtures observed once, π is very nearly a unique key, and a generator could in principle satisfy every conditional statistic by

Table 30: Generated model next-token disagreement from *vertex* pools. Vertices and interior are never pooled, organization and comparison follow structurally from π. Similar results for zoo members are shown in Table 6. A related Table 22 in the main body shows the same experiment run on the interior of Δ^4. We again find that generated models show higher disagreement rates than noise, but lower next token disagreement rates than original models share (see Table 3).

pool	Mini	Small	Medium
flow vs. its target	0.3036	0.2991	0.2939
1-NN retrieval vs. target	0.3136	0.2995	0.2887
flow vs. other π	0.2744	0.2503	0.2427
replicate floor (same π)	0.1657	0.1791	0.1892
rank-50 floor	0.0192	0.0133	0.0120
flow cohort spread	0.1302	0.1343	0.1585

storing one point per mixture. Four independent checks bear on it, so memorization would have to defeat all four.

1. The three-sample data-copying test. Following **?** we compare nearest-neighbour distances from *generated* samples to the training set against the same distances from *real held-out* members, via a Mann–Whitney statistic Z_U: a copier gives $Z_U \ll 0$, an underfit generator $Z_U \gg 0$. The third sample is the set of members the flow was blinded to, which remain rows of the basis so that their codes are exact; holding them out of the *basis* instead would inflate their distances and bias the verdict toward declaring copying (Appendix B.3).

Table 31: Three-sample data-copying test on the leading five components, at four blinded neighbourhoods and two scales. The mixture plan is identical across scales, so these are literally the same four mixtures. $Z_U < -3$ is copying, $Z_U > +3$ is underfitting, and $|Z_U| \leq 3$ is generated points sitting where real held-out data sits.

scale	n_{test}	gen→tr	test→tr	Z_U
Mini	9	0.352	0.435	−2.36
	4	0.511	0.213	+2.59
	9	0.357	0.613	−2.72
	9	0.331	0.292	−1.03
Medium	9	0.671	0.385	+1.99
	4	0.934	0.167	+3.39
	9	0.704	0.572	+1.94
	9	0.668	0.255	+4.03

No configuration at either scale triggers the copying verdict, and the failure mode that does appear is underfitting. At Mini the four statistics straddle zero; at Medium all four are positive and two exceed the underfitting threshold. The sign change rests on this test alone — the nearest-neighbour ratio stays below 1 at Medium (Table 28) and does not corroborate it — and the test is underpowered: its third sample is four to nine members and its signs disagree at Mini. So we do not lean on it. What can be said is that copying is not detected anywhere, and that a memorizing generator would produce a strongly negative Z_U that never appears.

2–4. Three further checks. A generator emitting training members verbatim has zero conditional spread, so its CRPS is dominated by the missing diversity term and its held-out log-likelihood is unbounded below — which makes the density of Appendix B.8 the sharpest available collapse control, reported as a control rather than a headline so that no claim depends on a divergence estimator being correct. The decisive check is functional. Emitting a copy of the nearest training member *is* the retrieval baseline, scored on the same cells under the same blinding, so a generator that copies cannot beat retrieval — it can only tie it. Any margin over retrieval is margin copying cannot produce, which is why that comparison is stated per cell and against extrapolation distance, never as a single average.

B.13 Downstream Benchmarks

Accuracy on a multiple-choice suite is quantized at $1/n$ and these models sit within a few points of chance, so Table 32 can show parity without being able to distinguish it from a failure to measure. MMLU is the limiting case and is reported nowhere: across the 435 models scored its accuracy takes ten or fewer distinct values, every one within five questions of 0.242, while the log-probabilities it assigns differ on all $2{,}000$ questions — the argmax over four single-letter continuations never moves, so the column reports the prior over those four tokens.

Table 32: HellaSwag accuracy for the runs of Table 5, $2{,}000$ questions, zero shot. *real* is the members pretrained at that mixture, averaged within a cell before the five mixtures of a stratum are averaged; *flow* is the leave-one-mixture-out generator at 8 samples per cell. Accuracy is quantized at $1/2000 = 0.0005$ and chance is 0.25.

	stratum	real	flow	Δ
Mini (51M)	interior	0.2521	0.2520	-0.0001
	vertices	0.2558	0.2519	-0.0039
Small (124M)	interior	0.2574	0.2580	$+0.0006$
	vertices	0.2616	0.2590	-0.0026
Medium (355M)	interior	0.2729	0.2715	-0.0014
	vertices	0.2720	0.2680	-0.0040

The log score of Table 5 is the sharper instrument. Scores are near-chance primarily due to the $20D$ training token choice. Models at similar scales, like HuggingFace's SmolLM 135M score $\sim 42\%$, but is trained on almost $250B$ tokens, compared to our $1B$ to $6B$ tokens (Allal et al., 2024).