Pretraining Numerical Frequency and Number-Line in Language Models

Mohammed Ibrahim Awad [1 *] **Ahmed Elshehaby** [1 *] **Velibor Bojkovic** [1 †] **Hilal AlQuabeh** [1 †]

Abstract

Large language models exhibit compressed, non-uniform internal representations of numerical magnitude, but the pretraining factors associated with this geometry remain unclear. We study whether corpus-level integer statistics are related to the learned number-line geometry of these models. For four documented pretraining corpora, we count integers in $[0 : 10{,}000]$ and fit a magnitude-frequency power law, $\mathrm{count}(N) \propto N^{\alpha}$, where more negative α indicates steeper decay and less exposure to large magnitudes. For nine corresponding base models, we extract hidden states for numerical prompts, project them onto a one-dimensional number line with PCA, and estimate a scaling factor β, where smaller β indicates stronger compression. We first show that β is behaviorally meaningful: models with less compressed number-line geometry achieve higher likelihood-based number-comparison accuracy. We then find that flatter integer-frequency distributions, corresponding to less negative α, are associated with larger β, providing correlational evidence that pretraining integer statistics are reflected in LLM number representations.

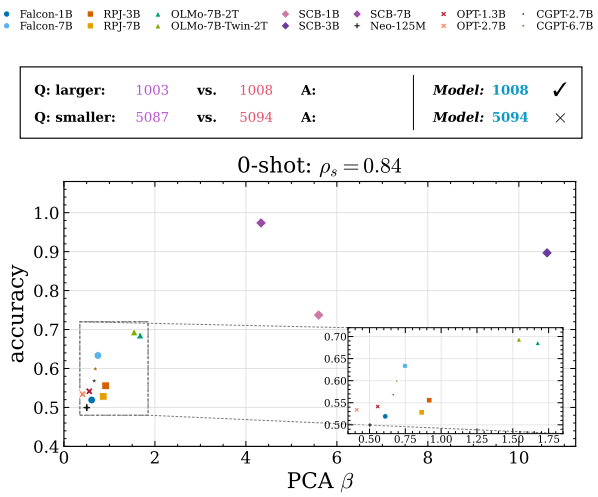

Figure 1. Relationship between the compression factor β and numerical comparison accuracy across models in the 0-shot setting. Each point corresponds to a model, with β measured from dimensionality reduction for number-line representations and accuracy computed on the number-comparison task. A clear positive correlation is observed ($\rho_s = 0.84$), indicating that models with less compressed numerical representations (larger β) achieve higher accuracy. The prompts examples above illustrate the pairwise comparison format. (refer to Section 4 for detailed experiments and results)

1. Introduction

Numerical reasoning remains an important challenge for Large Language Models (LLMs), since mathematical tasks often depend on how models internally represent, compare, and manipulate these numbers. This makes it important to understand how numerical values are encoded internally, and whether models develop an internal notion of a number line that may shape their reasoning on mathematical tasks.

The linear representation hypothesis (Park et al., 2024) states that high-level concepts are represented internally in LLMs as approximately linear directions or subspaces. This motivates the question of whether numerical magnitude is also organized in a simple linear way inside models. Supporting this possibility, Zhu et al. (2025) showed that numerical values can be recovered from LLM hidden states using linear probes. Being linearly recoverable does not necessarily mean that the complete internal number line is uniformly linearly spaced. Instead, recent findings suggest that LLMs encode numerical magnitude in a manner similar to human numerical cognition, in a more compressed non-linear manner, following the logarithmic mental number line hypothesis (Shah et al., 2023; AlquBoj et al., 2025), as illustrated in Figure 7.

More specifically, AlquBoj et al. (2025) showed that internal number representations in LLMs are not uniformly spaced, rather the spacing between number representations changes as numerical magnitude increases. They quantify

[*]Equal contribution [1]Mohamed bin Zayed University of Artificial Intelligence, Abu Dhabi, United Arab Emirates. Correspondence to: Velibor Bojkovic <velibor.bojkovic@mbzuai.ac.ae>, Hilal AlQuabeh <hilal.alquabeh@mbzuai.ac.ae>.

Mechanistic Interpretability Workshop at of the 43rd International Conference on Machine Learning, Seoul, South Korea, 2026.
Copyright 2026 by the author(s).

this compression factor using the Scaling Rate Index β. which measures how strongly the internal spacing form a logarithmic-like geometry, where smaller β means stronger compression.

While prior work identifies compressed number-line geometry in LLMs, two questions remain open: **1- what drives different compression rates across models, and 2- whether this geometry has practical relevance for numerical reasoning**. We address the latter question first by directly relating the compression factor β to model performance on simple multiple-choice numerical reasoning questions. We find that models with larger β, corresponding to less compressed and more evenly spaced internal number lines, tend to achieve higher accuracy, as shown in Section 4. This provides evidence that number-line geometry is not only a representational property, but is also related to numerical reasoning ability.

To answer the former question: what factors shape β across models? We investigate pre-training number frequency as one possible factor. This motivation is inspired by human numerical development, where exposure and formal training affect number-line judgments, and by recent evidence that pre-training frequency influences representation geometry in LLMs (Merullo et al., 2025).

To test this hypothesis, we connect corpus-level number statistics to model-level representation geometry, as summarized in Figure 2. Specifically, we estimate the frequency decay term α from integer counts in pre-training data using $f_{\text{count}}(N) \propto N^{\alpha}$, and compare it with the number-line compression factor β, estimated from one-dimensional projections of hidden-state number representations. Across nine LLMs trained on known datasets, we find a positive correlation between α and β: sharper frequency decay in the data (α more negative) is associated with stronger internal number-line compression (β smaller).

In short, we summarize our contributions as follows:

- We identify a general empirical trend showing that LLM accuracy on mathematical reasoning tasks tend to increase as the number-line compression factor β increases. This behavior is described in Section 4.
- We extract the empirical frequency distribution of integers $x \in [0, 10000]$ from four distinct open-source pretraining datasets summarized in Table 1.
- We fit a magnitude-based power-law model to the empirical integer frequency distribution and introduce the decay term α, where smaller values of α (more negative values) indicate that integer frequency decreases more rapidly as numerical value increases.
- We study nine LLMs that were trained only on their respective pre-training datasets listed in Table 1 and find a positive correlation between the compression factor β of these models and the decay term α of their respective pre-training dataset, providing empirical evidence that the integer-frequency distribution is related to the geometry of internal number representations.

2. Related Work

Many previous studies have explored whether language models can understand and reason about numbers. Early work on numeracy in LLMs showed that while models can capture some numerical information, they still struggle with numerical prediction and arithmetic reasoning (Spithourakis & Riedel, 2018; Wallace et al., 2019). More recent work has shown that LLMs exhibit human-like numerical-comparison effects, such as distance and ratio effects, suggesting that LLMs may capture behavioral patterns of numerical magnitude similar to those in human numerical cognition (Shah et al., 2023). Closely related to our motivation, Razeghi et al. (2022) showed that LLMs perform better on numerical reasoning tasks when the relevant terms occur more frequently in the pre-training data, suggesting that numerical ability might be influenced not only by abstract reasoning but also by the distribution of numbers during pre-training.

Beyond task-based performances, other works have studied how numerical values are represented in the hidden layers of LLMs. Some studies suggest that LLMs internally encode number values in their hidden states, and that these values can be recovered using linear probes (Zhu et al., 2025). However, this does not necessarily mean that the entire number line is uniformly spaced. Recent works have shown that different LLMs can learn similar numerical representations, suggesting that some numerical geometry may emerge consistently across datasets and architectures (Fu et al., 2026). At the same time, other studies also show that LLMs may represent numbers using the base-10 pattern and string-like information, rather than treating all numbers as pure numerical magnitude (Levy & Geva, 2025; Marjieh et al., 2025), suggesting that maybe the numerical internal representation, which may be structured, is not necessarily uniform, simple, or purely based on magnitude

Recent work by AlquBoj et al. (2025) has shown that LLMs encode numerical representation in a compressed and non-uniform manner similar to the logarithmic mental number line observed in human cognition. Their work quantifies this compression using the Scaling Rate Index β, which measures how the internal spacing of numerical representations changes as the magnitude of the number increases. In parallel, work on pre-training frequency has suggested that the statistics of the pre-training corpus can shape LLMs' internal representations. Merullo et al. (2025) showed that concepts that appear more frequently in the pre-training

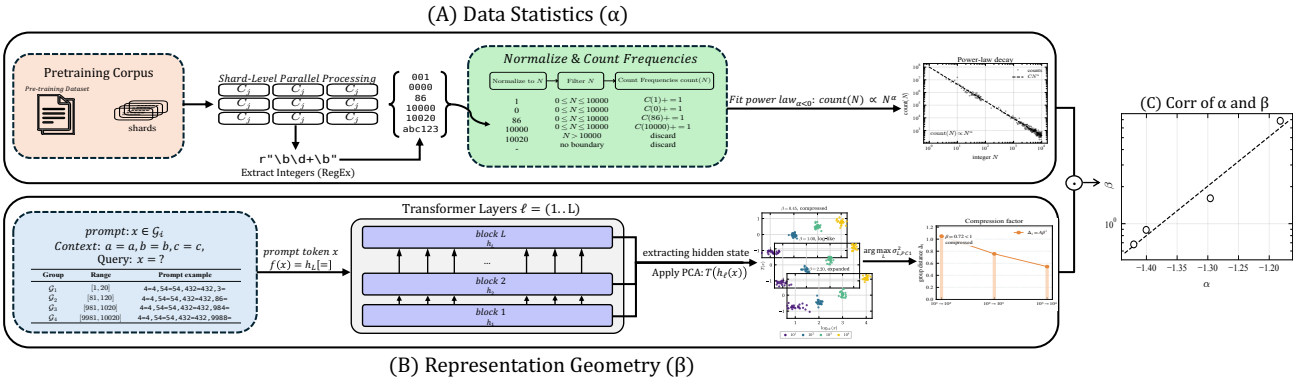

Figure 2. Overview of the proposed pipeline. (A) We estimate the magnitude-frequency exponent α by extracting integers from pretraining corpora, normalizing them, and fitting a power-law to $\mathrm{count}(N)$. (B) We obtain the compression factor β by extracting hidden representations from transformer layers, projecting them via PCA, and analyzing spacing across numerical magnitudes. (C) We then analyze the relationship between corpus-level statistics (α) and representation geometry (β) across models.

data are more likely to form cleaner linear representations inside LLMs. This suggests that the frequency of numbers in the pre-training dataset not only influences their performance on numerical tasks but also influences their internal geometrical representation of numbers.

Our work, inspired by these directions, studies whether the distribution of integers in pre-training datasets is related to the compressed numerical line geometry observed in LLMs. Since natural language datasets are generally highly non-uniform and often follow Zipf-like patterns (Zipf, 1949; Newman, 2005), it is natural to expect the number frequency distribution to follow similar patterns. However, instead of fitting the classical Zipf's rank-frequency law, we study how the frequency of a number changes as a function of its numerical power law, estimating the decay term α (Clauset et al., 2009). This α captures how rapidly number frequency decreases as numerical magnitude increases. We link this pre-training data distribution to the internal geometry of numbers in LLMs and investigate the relationship between the decay term α and the compression factor β.

3. Background

Previous work has introduced approaches to evaluate compression rate and analyze numerical representations in LLMs through the use of dimensionality reduction techniques and linear probing (AlquBoj et al., 2025). The broader idea is investigating whether LLMs encode the number-line in an intuition similar to humans–logarithmic, sublogarithmic, and super-logarithmic.

As a result, the investigation starts by analyzing the hidden representations across the model layers, examining the geometric structure of numerical magnitudes, and the general underlying trends across these layers. For that reason, Principal Component Analysis (PCA), a dimensionality reduction technique, is being introduced to map the hidden representations to a one-dimensional number-line that best reflects their underlying numerical features. Similarly, using Spearman rank correlation and nonlinear regression, these approaches reveal whether LLM representations exhibit properties of human numerical cognition, including order preservation and a compression effect, where the distances between consecutive numbers decrease with increasing magnitude. The general experimental setup introduced in (AlquBoj et al., 2025) for analyzing numerical representations in LLMs serves as the basis, in this paper, for studying the emergence of a number-line structure and the associated compression behavior.

3.1. Low-Dimensional Projection and Rank Correlation

As discussed earlier, analyzing the structure of the hidden representations in the model through PCA is the first stage. Moreover, the input number process in LLMs is a map into a high-dimensional latent space, where each input x is associated with an internal representation $f(x) \in \mathbb{R}^d$. Analyzing the geometry of these representations over a set of inputs X provides insight into how the model organizes numerical magnitudes. In particular, it allows us to examine whether numerical values align along an implicit number-line and whether this representation exhibits uniform spacing or compression. This mapping is denoted by f_{LLM}, which captures the internal encoding of the hidden numerical representations of the model. Extending this idea, projecting the hidden representations onto a one-dimensional space via $T : \mathbb{R}^d \to \mathbb{R}$, using PCA:

$$f_{\mathrm{LLM}}(x) := T(f(x)), \tag{1}$$

where $f(x)$ denotes the high-dimensional internal representation of the input x, and T maps these representations onto a one-dimensional space.

For any two inputs $x, y \in X$, the distance between their projections following the mapping T using Euclidean norm (AlquBoj et al., 2025), is as follows:

$$d(x, y) = \|f_{\text{LLM}}(x) - f_{\text{LLM}}(y)\| \tag{2}$$

which serves as the basis for evaluating both monotonicity and scaling properties in the following analysis.

To further examine the preservation of the natural order of the numerical values, this is extended by studying the monotonic properties of the function f_{LLM}. Specifically, for $x_1 < x_2$, this requires $f_{\text{LLM}}(x_1) < f_{\text{LLM}}(x_2)$, or the reverse inequality if the direction is flipped, ensuring that the numerical magnitudes maintain their relative order after projection (AlquBoj et al., 2025).

To quantify this property, Spearman rank correlation plays an essential role in obtaining it. Given two vectors $X, Y \in \mathbb{R}$, let $R(X)$ and $R(Y)$ denote their rank-transformed version, where each element is replaced by its rank in the sorted sequence. The Spearman correlation coefficient, denoted by ρ, is defined as the covariance between the rank vectors normalized by their standard deviations:

$$\rho = \frac{Cov(R(X), R(Y))}{\sigma(R(X)) \cdot \sigma(R(Y))} \tag{3}$$

Spearman's ρ provides a nonparametric measure of the relationship between two sequences by comparing their relative rankings. Assesses whether the increase/decrease of one variable is contingent on the change of the other. The provided value ρ is the absolute value $|\rho|$, thereby treating both increasing and decreasing monotonic relationships interchangeably.

3.2. Number-Line Compressions and β Parameter

The quantification of numerical representations in LLMs is characterized through the notion of number-line compression, measuring the difference in distance between numerical magnitudes while evolving and increasing. Let $f(x)$ denote the internal representation within the mode. To analyze scaling across orders of magnitude, we consider inputs of the form $x_i = 10^i$ and define $y_i = f(x_i)$.

Following prior work, we examine the differences between consecutive representations (AlquBoj et al., 2025):

$$y_{i+1} - y_i = A \cdot \beta^i \tag{4}$$

To estimate the scaling parameter β, A and β are fitted to the observed differences $y_{i+1} - y_i$, by minimizing the least-squares objective.

$$\min_{\alpha,\beta} \sum_{i=1}^{n} \left((y_{i+1} - y_i) - A\beta^i\right)^2 \tag{5}$$

This objective in Equation (5) models how the difference between representations of consecutive inputs vary across scaled, in other words, according to the numerical scaling regime. Therefore, to capture the spacing and the geometric structure, the scaling parameter β, obtained via geometric regression, characterizes how the spacing between consecutive representations evolves along a monotonic sequence.

To elucidate, The scaling parameter β governs the evolving of spacing between representations across scales.

- if $\beta = 1$, the differences $y_{i+1} - y_i$ remain approximately constant, implying that equal multiplicative changes in input, for example 10^i to 10^{i+1}, correspond to equal additive distances in the representation space.
  *Therefore, f is a **logarithmic** function on x_i*
- if $\beta > 1$, the differences increase with i, indicating that the distances between larger numbers expand, meaning super-logarithmic behavior.
  *Therefore, f is a **super-logarithmic** function on x_i*
- Conversely, if $\beta < 1$, the differences decrease with i, indicating compression in which the distances between larger numbers become increasingly packed, which means sub-logarithmic behavior.
  *Therefore, f is a **sub-logarithmic** function on x_i*

For a better understanding of the role of the scaling parameter β, the study of the mapping f_{LLM} is carried out on exponentially spaced inputs. Recall that $x_i = 10^i$, the index i corresponds to logarithmic scale of the input. From Equation (4), we have $y_{i+1} - y_i = A\beta^i$, and hence $y_{i+1} - y_i \propto \beta^i$. Reformulating this in terms of x_i, we obtain the following.

$$\beta^i = \beta^{log_{10}(x)} = x^{log_{10}(\beta)} \tag{6}$$

Thorough analysis of Equation (6) gives an intuitively clear understanding of different scaling regimes:

- if $\beta = 1$, the difference remains constant, producing a logarithmic scaling as shown in Equation (4)
- if $1 < \beta < 10$, the mapping results in faster than logarithmic but slower than linear, corresponding to sublinear behavior (i.e., super-logarithmic but sublinear).
- if $\beta = 10$, this results in a linear mapping:

$$\beta^i = 10^i = 10^{\log_{10}(x_i)} = x_i^{\log_{10}(10)} = x_i \tag{7}$$

- if $\beta > 10$, the growth becomes faster than linear, indicating superlinear behavior, where the distances between the representations expand at an increasing rate.

Building upon this and to further interpret the behavior of compression and scaling in a more conventional setting, consider the case where inputs are linearly spaced, i.e., $x_i = i$. Under the same formulation of Equation (4), the parameter β characterizes the growth of the sequence directly.

- $\beta = 1$ corresponds to linear growth.
- $\beta < 1$ corresponds to sublinear (concave (Rockafellar, 1970)) behavior in which increments decrease.
- $\beta > 1$ corresponds to superlinear (convex (Rockafellar, 1970)) in which increments increase.

This complements the previous analysis in Equation (7) by providing an interpretation of β in terms of a standard growth regime.

Algorithm 1 Monotonicity (ρ) and Compression (β)

Require: Log-spaced $\{G_i\}$ with $x_i = 10^i$, model $\mathcal{M}$
Ensure: Monotonicity score ρ, scaling parameter β
$\{h_\ell(x)\}_{\ell=1}^{L} \leftarrow \mathcal{M}(x),\ \forall x \in \bigcup_i G_i$
for each layer ℓ **do**
  $T_\ell \leftarrow \mathrm{PCA}_1(\{h_\ell(x)\})$
  $\tilde{f}_\ell(x) \leftarrow T_\ell(h_\ell(x))$
  $\sigma_\ell^2 \leftarrow \mathrm{Var}(\tilde{f}_\ell(x))$
  $\rho_\ell \leftarrow \mathrm{Spearman}(x, \tilde{f}_\ell(x))$
  for each group G_i **do**
    $\bar{f}_\ell(i) \leftarrow \mathbb{E}_{x \in G_i}[\tilde{f}_\ell(x)]$
  end for
  $\mathcal{L}_\ell(A, \beta) = \sum_{i=1}^{n-1} \left((\bar{f}_\ell(i+1) - \bar{f}_\ell(i)) - A\beta^i\right)^2$
  $\beta_\ell \leftarrow \beta^\star$, where $(A^\star, \beta^\star)$ solves $\min_{A,\beta>0} \mathcal{L}_\ell(A, \beta)$
end for
$\ell^* \leftarrow \arg\max_\ell \sigma_\ell^2$
$\rho \leftarrow \rho_{\ell^*}$
$\beta \leftarrow \beta_{\ell^*}$
return ρ, β

4. LLMs Numerical Reasoning Ability and β

Following up on LLM conceiving of the number-line, further experiments were implemented to figure out if there is a direct impact of β for each LLM and their ability to perform certain types of arithmetic questions that evaluate the understanding of the number-line, where the difficulty of answering those questions contingent on the number-line's encoding–sublogarithmic, logarithmic, or super-logarithmic. The prompt questions are comparison-based questions that are structured to evaluate LLMs ability to compare between two numbers. These tasks assess the model's ability to distinguish between numerical magnitudes and reason about their relative value comparatively with another.

Specifically, in each prompt, the model is given a set of numerical values and is required to identify the value that satisfies the comparison conditions, whether it is the largest or smallest number. For instance, given inputs $\{a, b\}$, the model must select $\min\{a, b\}$ or $\max\{a, b\}$. To further expand this experiment, the difference between $\mathrm{d}(a, b) = |a - b|$, which is called the gap, is categorized into 10 different gaps. Each gap corresponds to the difference based on its rank. For example, gap 1 means a digit difference. This setup directly evaluates whether the encoding of number-line does impact the ability for LLMs to reason about the magnitudes of numbers. Intuitively, if LLMs understand number-line in a logarithmic manner, the bigger the numbers get, the harder the comparison gets as the gaps between larger numbers get closer.

4.1. Generation of Number-Comparison Dataset

To evaluate whether the geometry of the learned number-line is reflected in numerical reasoning of LLMs, we construct a synthetic pairwise number-comparison dataset. The dataset is designed to control for numerical magnitude, absolute gap size, answer position, and query direction. Firstly, we partition comparison pairs into four magnitude regimes corresponding to powers of ten. These groups allow us to analyze how the model behavior varies as the numerical magnitude increases.

$$\mathcal{G}_1 = [10, 99], \qquad \mathcal{G}_2 = [100, 999],$$
$$\mathcal{G}_3 = [1000, 9999],\ \mathcal{G}_4 = [10000, 99999].$$

Within each magnitude group, we generate fixed-gap pairs of the form $(a, b) = (n, n + d)$ with two gap regimes:

$$SG_{small_gap} : d \in \{1, 2, 3, 4, 5\}$$
$$MG_{medium_gap} : d \in \{6, 7, 8, 9, 10\}$$

For each group and each gap value, we sample 32 starting values n using a fixed random seed, ensuring a balanced distribution across magnitude and gap size. In addition, we evaluate each model on a 0-4 shot prompt. For few-shot settings, demonstration examples are prepended to the prompt excluding 0-shot. To reduce dependence on a particular prompt configuration, we use multiple exemplar sets. Each group and gap regime yields 640 comparisons, or 1920 with three exemplar sets; across four groups and two regimes, this totals 15,360 comparisons per model-shot condition. We report the main 0–1 shot results in Figure 3, and provide the 2–4 shot results in Section C.3.

4.2. Experimental Setup for Number Comparison

The evaluation was carried out on 9 open-source language models summarized in Table 1. For the few-shot comparison analysis, we also include GPT-Neo-125M (Black et al.,

2021), OPT-1.3B, OPT-2.7B (Zhang et al., 2022), Cerebras-GPT-2.7B and Cerebras-GPT-6.7B (Dey et al., 2023); these models are not included in the pretraining corpus analysis because their exact pretraining datasets are not available in the same directly matched form.

To ensure robustness, both input orderings (a, b) and (b, a) are considered, along with prompt directions ("larger" and "smaller"), preventing models from relying on positional or heuristic biases. Rather than relying on free-form generation, we adopt a likelihood evaluation. Given a prompt q and two candidate outputs c_a and c_b, we compute the log-likelihood normalized in length for each candidate and select the one with the highest score, as follows:

$$s(c|q) = \frac{1}{|c|} \sum_{t=1}^{|c|} log\ p(c_t|q, c_{<t}) \qquad (8)$$
$$\hat{c} = \arg \max_{c \in c_a, c_b} s(c|q)$$

Ensure that the evaluation reflects the model's internal preference between numerical magnitudes and not decoding artifacts.

As a result, We observe that the performance of the models exhibits a strong positive correlation with the scaling factor β in all shot settings (see Figure 3). In particular, models with larger values β consistently achieve higher comparison accuracy, indicating that less compressed numerical representations correspond to better discrimination between numerical magnitudes. This trend holds across 0-4 shot configurations, with Spearman correlation coefficients remaining high, suggesting that there is a monotonic relationship between β and numerical reasoning.

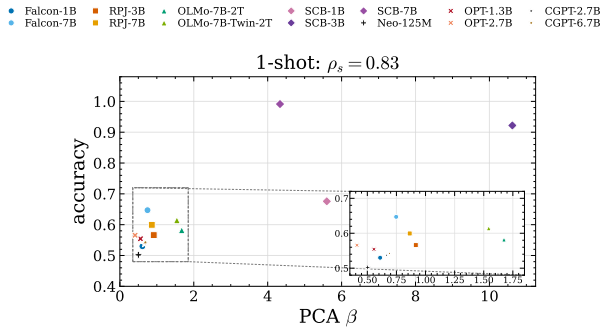

Figure 3. Accuracy on number-comparison task vs the PCA-based scaling factor β. Spearman correlation ρ_s indicates that models with larger β tend to achieve higher comparison accuracy.

Consequently, these results motivate us to do a thorough study into the factors that give rise to higher β values and the underlying mechanisms that influence them.

5. Power-Law Structure of Pretraining Data and its Effect on β

In order to have a complete and unbiased understanding of the underlying influence of the training dataset on the compression rate of number-lines, the access to the full datasets, which their models have been pre-trained on without any task-specific fine-tuning, instruction tuning, or post-training modifications, was crucial to further investigate these datasets under rigorous scrutiny. Therefore, the distinct chosen pre-training datasets (refer to Table 1), their models, 9 in total, are open-source and only pre-trained on them, are publicly accessible to analyze and count the exact occurrences of the integers $[0 : 10000]$. Building upon this, we fit power law to each dataset, and study the relationship between power law's exponent α and the compression factor β of the number-line in LLMs.

To conclude, we selected 9 models on these 4 distinct datasets, ensuring that these LLMs were trained on only the following pretraining datasets and were not finetuned or have additional reasoning methods on top. This is because there might be the possibility that the compression factor has been altered by an unexplainable factor, which we cannot account for while trying to find a correlation between β and α.

Datasets	Models	Desc
Falcon-RefinedWeb (Penedo et al., 2023)	falcon-rw-7b falcon-rw-1b	Filtered web pages from CommonCrawl
RedPajama-Data-1T (Weber et al., 2024)	RedPajama-INCITE-Base-3B-v1 RedPajama-INCITE-7B-Base	Mixed text corpus with web, books, Wikipedia, and code
Dolma (Soldaini et al., 2024)	OLMo-7B OLMo-7B-Twin-2T (Groeneveld et al., 2024)	Large open corpus with web, books, papers, wiki, and code
The Stack (Kocetkov et al., 2023)	starcoderbase-1b starcoderbase-3b starcoderbase-7b (Li et al., 2023)	Programming code dataset from public repositories

Table 1. Pre-training datasets and corresponding model families used in this study. All models are open-source and trained exclusively on their respective datasets.

5.1. Integer-Frequency Analysis of Pretraining Corpora

To test whether corpus statistics may explain the geometry of learned number representations, we measure how often integers occur in several pretraining datasets. For each corpus, we count integer mentions in the range $0 \leq N \leq 10000$. This range matches the numerical scale used in our representation analysis and allows us to compare the frequency structure across datasets with β. For each dataset, we perform a full pass over the text and extract digit strings corresponding to non-negative integers. Each matched integer is normalized to its numeric value, for example, 001 becomes 1, so different textual occurrences of the same integer contribute to the same count. We then aggregate the counts into an empirical frequency distribution $\mathrm{count}(N)$,

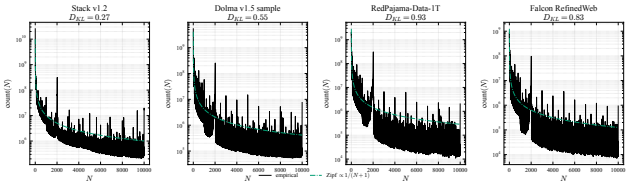

Figure 4. Empirical integer-frequency distributions for the pretraining datasets. D_{KL} measures how far each dataset deviates from this baseline (see Section B, Equation (9)).

where $\mathrm{count}(N)$ *denotes the number of times the integer* N *appears in the corpus.*

We model the empirical integer distribution (shown in Figure 4) as an approximately Zipfian power law with numerical magnitude: $\mathrm{count}(N) \propto N^{\alpha<0}$. Here, N denotes the integer value itself, not the frequency rank. Thus, α is a magnitude-frequency exponent: Smaller values of α (more negative values) indicate that the frequency decays more rapidly as the numerical magnitude increases; α reported in Table 2.

Equivalently, we estimate α by fitting a linear model in log–log space:

$$\log \mathrm{count}(N) = c + \alpha \log N + \epsilon_N,$$

for integers with nonzero counts. This exponent is used in our corpus-frequency hypothesis because both α and the representation compression factor β are defined with respect to the numerical magnitude.

Dataset	Count				Magnitude Fit	
	$[0, 10)$	$[10, 10^2)$	$[10^2, 10^3)$	$[10^3, 10^4]$	α	R^2
Stack v1.2	51.530	23.656	13.420	10.050	-1.18	0.91
Dolma v1.5 sample	18.879	11.297	4.701	5.725	-1.30	0.81
RedPajama-Data-1T	9.849	8.402	3.059	6.741	-1.40	0.68
Falcon RefinedWeb	4.633	4.017	1.426	2.222	-1.42	0.79

Table 2. Integer-frequency mass across magnitude bins for each pretraining dataset. Count entries are reported in billions. The final columns report the fitted magnitude-frequency exponent α from $\mathrm{count}(N) \propto N^{\alpha}$ and the corresponding log-log fit R^2. Smaller α indicates a steeper decay in exposure as numerical magnitude increases.

5.2. Number-Line Geometry and Its Relation to Integer-Frequency

After characterizing the frequency structure of integers in each pretraining corpus, we investigate whether these statistics are reflected in the internal geometry of numerical representations. For each model, we extract hidden representations for numerical inputs and apply PCA to analyze their geometric structure, reported in Table 3

Across models, we observe that PCA often produces a clear monotonic ordering of numerical values. Falcon and RedPajama models exhibit strong number-line structure, with best-layer correlations around $\rho \approx 0.92$–0.93. OLMo models also achieve high correlations, but with larger β values, indicating less compressed representations. StarCoderBase models exhibit the largest β values, corresponding to substantially expanded spacing at higher magnitudes. Figure 5 visualizes the PCA number-line projections summarized in Table 3, while Figures 10 and 11 provide the corresponding layer-wise ρ and β.

Model	Dataset	PCA			
		Layer	ρ	β	σ^2
Falcon-RW-1B	RefinedWeb	8	0.93 ± 0.01	0.61 ± 0.01	0.31 ± 0.00
Falcon-RW-7B		16	0.93 ± 0.01	0.75 ± 0.03	0.33 ± 0.01
INCITE-3B	RedPajama	16	0.93 ± 0.01	0.92 ± 0.06	0.39 ± 0.01
INCITE-7B		12	0.92 ± 0.01	0.86 ± 0.07	0.37 ± 0.01
OLMo-7B-2T	Dolma	20	0.92 ± 0.01	1.67 ± 0.23	0.49 ± 0.02
OLMo-7B-Twin-2T		13	0.92 ± 0.00	1.54 ± 0.12	0.50 ± 0.01
StarCoderBase-1B	Stack	10	0.82 ± 0.01	5.60 ± 2.86	0.37 ± 0.01
StarCoderBase-3B		16	0.81 ± 0.01	10.62 ± 6.07	0.37 ± 0.01
StarCoderBase-7B		20	0.85 ± 0.01	4.33 ± 1.10	0.37 ± 0.00

Table 3. PCA-based analysis of numerical representations across models, reporting the selected layer, monotonicity score ρ, compression factor β, and explained variance σ^2.

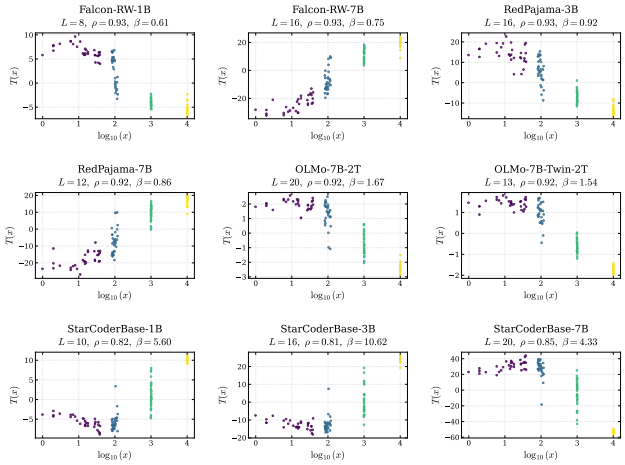

Figure 5. 1-D PCA projections of internal number representations $T(x)$ against $\log_{10}(x)$. Each plot shows the projected representation $T(x)$ against $\log_{10}(x)$. Monotonicity score ρ, and Scaling factor β reported for each model.

We now relate these observations to the frequency structure of the corresponding pretraining datasets. Let α denote the magnitude-frequency exponent:

$$\mathrm{count}(N) \propto N^{\alpha}.$$

Since the integer frequency decreases with magnitude, the effective decay rate is captured by $\alpha < 0$. Thus, smaller values of α (more negative values) correspond to faster decay (i.e., large numbers become rarer), while larger values (less

negative values) indicate a flatter distribution. According to our hypothesis, steeper decay (smaller α) reduces exposure to large numbers during pretraining, leading to stronger compression and therefore to smaller β. Conversely, flatter distributions (larger α) provide more exposure to large numbers, resulting in a weaker compression and larger β. Supported by empirical results in Tables 2 and 3.

Stack v1.2 has the flattest integer-frequency decay among the matched corpora, with signed exponent $\alpha = -1.18$. Its corresponding StarCoderBase models also exhibit the largest PCA scaling factors, with mean $\beta = 6.85$ across the three model sizes. Dolma shows an intermediate decay term ($\alpha = -1.30$), and the corresponding OLMo models have intermediate scaling factors ($\beta = 1.67,\ 1.54$). In contrast, RedPajama and Falcon RefinedWeb have steeper frequency decay ($\alpha = -1.40$ and $\alpha = -1.42$), and their associated models exhibit smaller scaling factors.

As shown in Figure 6, the matched corpus–model pairs show a monotonic trend: less negative α values correspond to larger β values. This supports the hypothesis that flatter numerical frequency distributions in pretraining data are associated with more expanded number-line representations, while steeper frequency decay is associated with stronger compression.

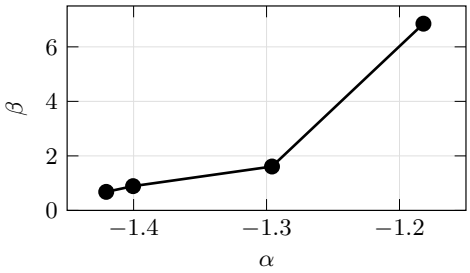

Figure 6. Relationship between the signed corpus frequency exponent α and the PCA-based numerical scaling factor β. Integer counts are fit as $C(N) \propto N^{\alpha}$, so more negative α indicates faster frequency decay. Each point aggregates the matched model family for one pretraining corpus.

6. Conclusion

We studied how LLM number-line geometry relates to integer-frequency structure in pretraining data. Across multiple model families, numerical representations align along a dominant PCA direction, forming a measurable internal number line. The spacing of this line varies systematically: models trained on corpora with flatter integer-frequency decay exhibit larger scaling factors β, while corpora with steeper decay yield more compressed representations.

These results suggest that number representations are shaped by corpus-level statistics, not only scale. In particular, the signed exponent α from $\mathrm{count}(N) \propto N^{\alpha}$ predicts variation in the geometry of the learned number-line. The number-comparison probe in Section 4 further shows that less compressed representations are associated with higher comparison accuracy, indicating measurable behavioral consequences.

7. Limitations

Our analysis is limited by access to exact pretraining data. Many models are trained on only partially documented mixtures, so dataset counts are approximate; Falcon RefinedWeb, for example, is only a proxy for Falcon's full training distribution. We also focus on digit-based prompts and PCA projections of hidden states, which do not cover all numerical reasoning phenomena, such as dates, units, or written number forms. Future work should test broader numerical formats and tasks.

References

AlquBoj, H. V., AlQuabeh, H., Bojkovic, V., Hiraoka, T., El-Shangiti, A. O., Nwadike, M., and Inui, K. Number representations in llms: A computational parallel to human perception, 2025. URL https://arxiv.org/abs/2502.16147.

Black, S., Gao, L., Wang, P., Leahy, C., and Biderman, S. GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow, March 2021. URL https://doi.org/10.5281/zenodo.5297715. If you use this software, please cite it using these metadata.

Clauset, A., Shalizi, C. R., and Newman, M. E. J. Power-law distributions in empirical data. *SIAM Review*, 51 (4):661–703, 2009. doi: 10.1137/070710111. URL https://doi.org/10.1137/070710111.

Davies, A. O., Nzoyem, R., Ajmeri, N., and Silva Filho, T. M. Language models do not embed numbers continuously. *arXiv preprint arXiv:2510.08009*, 2025. doi: 10.48550/arXiv.2510.08009. URL https://arxiv.org/abs/2510.08009.

Dehaene, S. The neural basis of the Weber–Fechner law: A logarithmic mental number line. *Trends in Cognitive Sciences*, 7(4):145–147, 2003. doi: 10.1016/S1364-6613(03)00055-X. URL https://doi.org/10.1016/S1364-6613(03)00055-X.

Dehaene, S., Izard, V., Spelke, E., and Pica, P. Log or linear? distinct intuitions of the number scale in western and amazonian indigene cultures. *Science*, 320(5880): 1217–1220, 2008. doi: 10.1126/science.1156540. URL https://doi.org/10.1126/science.1156540.

Dey, N., Gosal, G., Zhiming, Chen, Khachane, H., Marshall, W., Pathria, R., Tom, M., and Hestness, J. Cerebras-gpt: Open compute-optimal language models trained on the cerebras wafer-scale cluster, 2023. URL https://arxiv.org/abs/2304.03208.

El-Shangiti, A. O., Hiraoka, T., AlQuabeh, H., Heinzerling, B., and Inui, K. The geometry of numerical reasoning: Language models compare numeric properties in linear subspaces. In Chiruzzo, L., Ritter, A., and Wang, L. (eds.), *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pp. 550–561, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-190-2. doi: 10.18653/v1/2025.naacl-short.47. URL https://aclanthology.org/2025.naacl-short.47/.

Fechner, G. T. *Elemente der Psychophysik*. Breitkopf und Härtel, Leipzig, 1860. doi: 10.3931/e-rara-10879. URL https://doi.org/10.3931/e-rara-10879.

Fritz, A., Ehlert, A., and Balzer, L. Development of mathematical concepts as basis for an elaborated mathematical understanding. *South African Journal of Childhood Education*, 3(1):38–67, 2013.

Fu, D., Zhou, T., Belkin, M., Sharan, V., and Jia, R. Convergent evolution: How different language models learn similar number representations, 2026. URL https://arxiv.org/abs/2604.20817.

Groeneveld, D., Beltagy, I., Walsh, E., Bhagia, A., Kinney, R., Tafjord, O., Jha, A., Ivison, H., Magnusson, I., Wang, Y., Arora, S., Atkinson, D., Authur, R., Chandu, K., Cohan, A., Dumas, J., Elazar, Y., Gu, Y., Hessel, J., Khot, T., Merrill, W., Morrison, J., Muennighoff, N., Naik, A., Nam, C., Peters, M., Pyatkin, V., Ravichander, A., Schwenk, D., Shah, S., Smith, W., Strubell, E., Subramani, N., Wortsman, M., Dasigi, P., Lambert, N., Richardson, K., Zettlemoyer, L., Dodge, J., Lo, K., Soldaini, L., Smith, N., and Hajishirzi, H. OLMo: Accelerating the science of language models. In Ku, L.-W., Martins, A., and Srikumar, V. (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15789–15809, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.841. URL https://aclanthology.org/2024.acl-long.841/.

Gurnee, W., Ameisen, E., Kauvar, I., Tarng, J., Pearce, A., Olah, C., and Batson, J. When models manipulate manifolds: The geometry of a counting task, 2026. URL https://arxiv.org/abs/2601.04480.

Hasani, H., Banayeeanzade, M., Nafisi, A., Mohammadian, S., Askari, F., Bagherian, M., Izadi, A., and Baghshah, M. S. Mechanistic interpretability of large-scale counting in llms through a system-2 strategy, 2026. URL https://arxiv.org/abs/2601.02989.

Heinzerling, B. and Inui, K. Monotonic representation of numeric attributes in language models. In Ku, L.-W., Martins, A., and Srikumar, V. (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 175–195, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-short.18. URL https://aclanthology.org/2024.acl-short.18/.

Kocetkov, D., Li, R., allal, L. B., LI, J., Mou, C., Jernite, Y., Mitchell, M., Ferrandis, C. M., Hughes, S., Wolf, T., Bahdanau, D., Werra, L. V., and de Vries, H. The stack: 3 TB of permissively licensed source code. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=pxpbTdUEpD.

Levy, A. A. and Geva, M. Language models encode numbers using digit representations in base 10. In Chiruzzo, L., Ritter, A., and Wang, L. (eds.), *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pp. 385–395, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-190-2. doi: 10.18653/v1/2025.naacl-short.33. URL https://aclanthology.org/2025.naacl-short.33/.

Li, R., Allal, L. B., Zi, Y., Muennighoff, N., Kocetkov, D., Mou, C., Marone, M., Akiki, C., Li, J., Chim, J., Liu, Q., Zheltonozhskii, E., Zhuo, T. Y., Wang, T., Dehaene, O., Lamy-Poirier, J., Monteiro, J., Gontier, N., Yee, M.-H., Umapathi, L. K., Zhu, J., Lipkin, B., Oblokulov, M., Wang, Z., Murthy, R., Stillerman, J. T., Patel, S. S., Abulkhanov, D., Zocca, M., Dey, M., Zhang, Z., Bhattacharyya, U., Yu, W., Luccioni, S., Villegas, P., Zhdanov, F., Lee, T., Timor, N., Ding, J., Schlesinger, C. S., Schoelkopf, H., Ebert, J., Dao, T., Mishra, M., Gu, A., Anderson, C. J., Dolan-Gavitt, B., Contractor, D., Reddy, S., Fried, D., Bahdanau, D., Jernite, Y., Ferrandis, C. M., Hughes, S., Wolf, T., Guha, A., Von Werra, L., and de Vries, H. StarCoder: May the Source Be with You! *Transactions on Machine Learning Research*, 2023. URL https://mlanthology.org/tmlr/2023/li2023tmlr-starcoder/.

Marjieh, R., Veselovsky, V., Griffiths, T. L., and Sucholutsky, I. What is a number, that a large language model may

know it?, 2025. URL https://arxiv.org/abs/2502.01540.

Merullo, J., Smith, N. A., Wiegreffe, S., and Elazar, Y. On linear representations and pretraining data frequency in language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=EDoD3DgivF.

Newman, M. E. J. Power laws, pareto distributions and Zipf's law. *Contemporary Physics*, 46(5):323–351, 2005. doi: 10.1080/00107510500052444. URL https://doi.org/10.1080/00107510500052444.

Park, K., Choe, Y. J., and Veitch, V. The linear representation hypothesis and the geometry of large language models. In Salakhutdinov, R., Kolter, Z., Heller, K., Weller, A., Oliver, N., Scarlett, J., and Berkenkamp, F. (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 39643–39666. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/park24c.html.

Park, S., Ryu, S., and Choi, E. Do language models understand measurements? In Goldberg, Y., Kozareva, Z., and Zhang, Y. (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 1782–1792, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-emnlp.128. URL https://aclanthology.org/2022.findings-emnlp.128/.

Penedo, G., Malartic, Q., Hesslow, D., Cojocaru, R., Alobeidli, H., Cappelli, A., Pannier, B., Almazrouei, E., and Launay, J. The refinedweb dataset for falcon llm: Outperforming curated corpora with web data only. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 79155–79172. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/fa3ed726cc5073b9c31e3e49a807789c-Paper-Datasets_and_Benchmarks.pdf.

Petrak, D., Moosavi, N. S., and Gurevych, I. Arithmetic-based pretraining improving numeracy of pretrained language models. In Palmer, A. and Camacho-collados, J. (eds.), *Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023)*, pp. 477–493, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.starsem-1.42. URL https://aclanthology.org/2023.starsem-1.42/.

Razeghi, Y., Logan IV, R. L., Gardner, M., and Singh, S. Impact of pretraining term frequencies on few-shot numerical reasoning. In Goldberg, Y., Kozareva, Z., and Zhang, Y. (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 840–854, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-emnlp.59. URL https://aclanthology.org/2022.findings-emnlp.59/.

Rockafellar, R. T. *Convex Analysis*. Princeton University Press, Princeton, 1970. ISBN 9781400873173. doi: doi:10.1515/9781400873173. URL https://doi.org/10.1515/9781400873173.

Shah, R., Marupudi, V., Koenen, R., Bhardwaj, K., and Varma, S. Numeric magnitude comparison effects in large language models. In Rogers, A., Boyd-Graber, J., and Okazaki, N. (eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 6147–6161, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.383. URL https://aclanthology.org/2023.findings-acl.383/.

Siegler, R. S. and Opfer, J. E. The development of numerical estimation: Evidence for multiple representations of numerical quantity. *Psychological Science*, 14(3):237–243, 2003. doi: 10.1111/1467-9280.02438. URL https://doi.org/10.1111/1467-9280.02438.

Soldaini, L., Kinney, R., Bhagia, A., Schwenk, D., Atkinson, D., Authur, R., Bogin, B., Chandu, K., Dumas, J., Elazar, Y., Hofmann, V., Jha, A., Kumar, S., Lucy, L., Lyu, X., Lambert, N., Magnusson, I., Morrison, J., Muennighoff, N., Naik, A., Nam, C., Peters, M., Ravichander, A., Richardson, K., Shen, Z., Strubell, E., Subramani, N., Tafjord, O., Walsh, E., Zettlemoyer, L., Smith, N., Hajishirzi, H., Beltagy, I., Groeneveld, D., Dodge, J., and Lo, K. Dolma: an open corpus of three trillion tokens for language model pretraining research. In Ku, L.-W., Martins, A., and Srikumar, V. (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15725–15788, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.840. URL https://aclanthology.org/2024.acl-long.840/.

Spithourakis, G. and Riedel, S. Numeracy for language models: Evaluating and improving their ability to predict numbers. In Gurevych, I. and Miyao, Y. (eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2104–2115, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1196.

URL https://aclanthology.org/P18-1196/.

Tiblias, F., Bigoulaeva, I., Niu, J., Balloccu, S., and Gurevych, I. Hypothesis-driven feature manifold analysis in LLMs via supervised multi-dimensional scaling. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=vCKZ40YYPr.

Wallace, E., Wang, Y., Li, S., Singh, S., and Gardner, M. Do NLP models know numbers? probing numeracy in embeddings. In Inui, K., Jiang, J., Ng, V., and Wan, X. (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 5307–5315, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1534. URL https://aclanthology.org/D19-1534/.

Wang, Z., Jiang, Y., Zhou, R., Zhang, B., Zhang, F., Xu, Z., Zhang, Y., and Wang, J. Drivecode: Domain specific numerical encoding for llm-based autonomous driving, 2026. URL https://arxiv.org/abs/2603.00919.

Weber, M., Fu, D. Y., Anthony, Q., Oren, Y., Adams, S., Alexandrov, A., Lyu, X., Nguyen, H., Yao, X., Adams, V., Athiwaratkun, B., Chalamala, R., Chen, K., Ryabinin, M., Dao, T., Liang, P., Ré, C., Rish, I., and Zhang, C. Redpajama: an open dataset for training large language models. In Globerson, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J., and Zhang, C. (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 116462–116492. Curran Associates, Inc., 2024. doi: 10.52202/079017-3697. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/d34497330b1fd6530f7afd86d0df9f76-Paper-Datasets_and_Benchmarks_Track.pdf.

Zhang, S., Roller, S., Goyal, N., Artetxe, M., Chen, M., Chen, S., Dewan, C., Diab, M., Li, X., Lin, X. V., Mihaylov, T., Ott, M., Shleifer, S., Shuster, K., Simig, D., Koura, P. S., Sridhar, A., Wang, T., and Zettlemoyer, L. Opt: Open pre-trained transformer language models, 2022. URL https://arxiv.org/abs/2205.01068.

Zhang, X., Ramachandran, D., Tenney, I., Elazar, Y., and Roth, D. Do language embeddings capture scales? In Cohn, T., He, Y., and Liu, Y. (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 4889–4896, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.439. URL https://aclanthology.org/2020.findings-emnlp.439/.

Zhou, Z., Wang, J., Lin, D., and Chen, K. Scaling behavior for large language models regarding numeral systems: An example using pythia. In Al-Onaizan, Y., Bansal, M., and Chen, Y.-N. (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 3806–3820, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.218. URL https://aclanthology.org/2024.findings-emnlp.218/.

Zhu, F., Dai, D., and Sui, Z. Language models encode the value of numbers linearly. In Rambow, O., Wanner, L., Apidianaki, M., Al-Khalifa, H., Eugenio, B. D., and Schockaert, S. (eds.), *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 693–709, Abu Dhabi, UAE, January 2025. Association for Computational Linguistics. URL https://aclanthology.org/2025.coling-main.47/.

Zipf, G. K. *Human Behavior and the Principle of Least Effort: An Introduction to Human Ecology*. Addison-Wesley Press, Cambridge, MA, 1949. URL https://archive.org/details/in.ernet.dli.2015.90211.

A. Logarithmic Mental Line Hypothesis

The logarithmic mental line hypothesis proposes that numerical magnitudes are not always represented with uniform spacing. Instead, smaller numbers are represented with relatively larger separations, while larger numbers become increasingly compressed. This produces an approximately logarithmic order of magnitude, where equal ratios are represented more similarly than equal absolute differences. Such compression has been observed in human numerical cognition, especially in approximate estimation and early numerical development, and provides a useful refernce point for studying whether LLMs organize numbers in a smaller geometric structure.

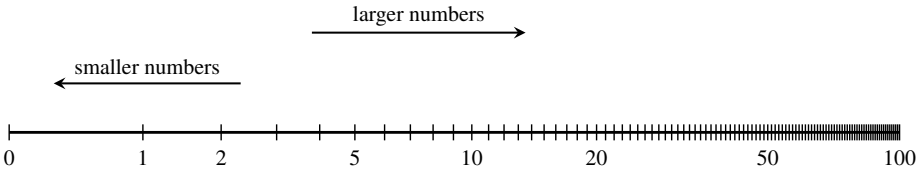

Figure 7. Logarithmically compressed mental number line. Image source (Fritz et al., 2013)

B. Additional Corpus-Frequency Diagnostics

In addition to estimating α, we compare the empirical distribution to a Zipf's law baseline:

$$q(N) = \frac{(N+1)^{-1}}{\sum_{k=0}^{10000}(k+1)^{-1}}.$$

We use $N+1$ to ensure the distribution is defined at $N=0$. Let

$$p(N) = \frac{\mathrm{count}(N)}{\sum_{k=0}^{10000}\mathrm{count}(k)}$$

be the normalized empirical distribution. To measure how different the empirical distribution is from this baseline, we compute the KL divergence:

$$D_{\mathrm{KL}}(p \parallel q) = \sum_{N=0}^{10000} p(N)\,\log\frac{p(N)}{q(N)} \tag{9}$$

This value serves as a secondary measure, indicating how closely the observed data follows a Zipf-like distribution.

From the observed results, Integer frequencies in natural text are not perfectly smooth and often exhibit systematic spikes due to round numbers, years, dates, and culturally salient values. To characterize these deviations, we analyze round-number categories:

$$N \equiv 0 \pmod{10}, \qquad N \equiv 0 \pmod{100}$$

C. Additional Details for the Number-Comparison Probe

This appendix provides additional details for the controlled number-comparison probe introduced in Section 4. The probe measures whether a model assigns higher likelihood to the numerically correct candidate in a pairwise comparison, rather than evaluating unrestricted mathematical generation.

C.1. Synthetic Task Construction

Each example contains two integers (a, b) and an instruction asking for either the larger or the smaller value. We generate pairs as

$$(a, b) = (n, n+d), \tag{10}$$

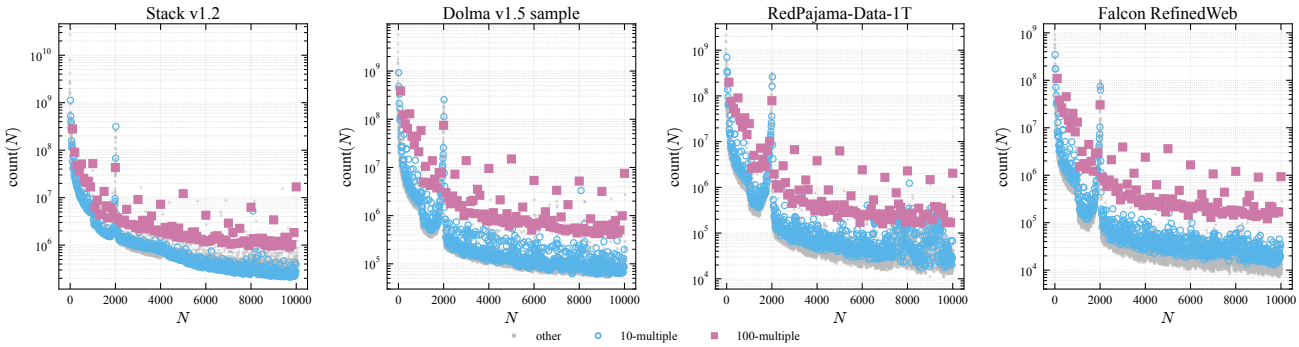

Figure 8. Integer-frequency distributions with round-number and year-like effects. Multiples of 10 and 100 show systematic spikes, with additional peaks around values such as 2000 likely reflecting frequent year mentions.

where n is sampled from one of four magnitude groups and d is a fixed absolute gap. The magnitude groups are

$$\mathcal{G}_1 = [10, 99], \quad \mathcal{G}_2 = [100, 999], \quad \mathcal{G}_3 = [1000, 9999], \quad \mathcal{G}_4 = [10000, 99999]. \tag{11}$$

We use two gap regimes:

$$\mathrm{SG} = \{1, 2, 3, 4, 5\}, \qquad \mathrm{MG} = \{6, 7, 8, 9, 10\}. \tag{12}$$

For each magnitude group and exact gap, we sample 32 starting values with a fixed seed. We include both candidate orderings, (a, b) and (b, a), and both query directions, "larger" and "smaller", so the benchmark is balanced against positional and lexical shortcuts.

C.2. Likelihood-Based Scoring

For a prompt q, the valid continuations are the two candidate numbers c_a and c_b. We score each candidate using length-normalized log-likelihood:

$$s(c \mid q) = \frac{1}{|c|} \sum_{t=1}^{|c|} \log p(c_t \mid q, c_{<t}), \tag{13}$$

where $|c|$ is the number of tokens in the candidate continuation. The model prediction is

$$\hat{c} = \arg \max_{c \in \{c_a, c_b\}} s(c \mid q). \tag{14}$$

Accuracy is then

$$\mathrm{Acc} = \frac{1}{M} \sum_{j=1}^{M} \mathbb{I}[\hat{c}_j = c_j^\star], \tag{15}$$

with $M = 15{,}360$ examples per model-shot condition. This evaluation avoids sampling variance and measures the model's relative preference between the two numerical candidates.

C.3. Few-Shot Results

The positive Spearman correlations in Table 4 show that the β–accuracy relationship is not specific to the zero-shot prompt. The relationship remains positive from 1-shot through 4-shot prompting, and the 2–4 shot panels in Figure 9 show the same qualitative pattern as the main 0–1 shot figure.

C.4. Interpretation of the Behavioral Probe

The number-comparison probe should be interpreted as a controlled preference test. Since both candidate answers are scored directly, the result is not affected by stochastic decoding or by whether the model chooses to emit explanatory text. However, it is still not a complete measure of mathematical reasoning: it tests pairwise numerical discrimination under a fixed prompt format. The consistent positive relationship between β and accuracy in Tables 4 and 5 supports the main claim that less compressed number-line geometry is associated with better numerical comparison behavior.

Shots	ρ_s	Pearson r	R^2	Mean Acc.
0	0.84	0.80	0.64	0.64
1	0.83	0.80	0.64	0.63
2	0.74	0.79	0.62	0.62
3	0.80	0.81	0.65	0.62
4	0.82	0.85	0.72	0.62

Table 4. Correlation between PCA scaling factor β and number-comparison accuracy across shot settings. Correlations are computed across the 14 evaluated base models.

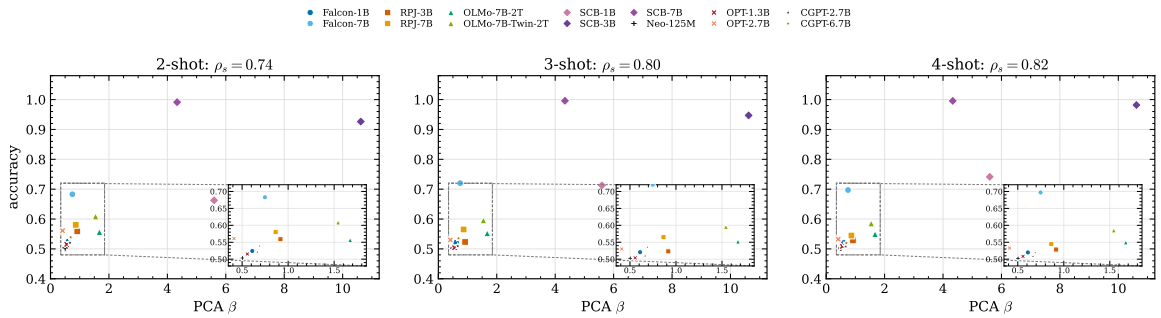

Figure 9. Number-comparison accuracy versus PCA scaling factor β for 2-, 3-, and 4-shot settings. Each point is one model; the inset zooms into the low-β region.

Model	Family	β	Type	0-shot	1-shot	2-shot	3-shot	4-shot
Falcon-1B	Falcon	0.61	baseline	0.519	0.530	0.524	0.521	0.520
Falcon-7B	Falcon	0.75	baseline	0.633	0.647	0.683	0.720	0.697
RPJ-3B	RedPajama	0.92	baseline	0.556	0.566	0.559	0.523	0.529
RPJ-7B	RedPajama	0.86	baseline	0.528	0.599	0.580	0.565	0.545
OLMo-7B-2T	OLMo	1.67	baseline	0.686	0.582	0.557	0.552	0.550
OLMo-7B-Twin-2T	OLMo	1.54	baseline	0.694	0.614	0.609	0.596	0.585
SCB-1B	StarCoderBase	5.60	baseline	0.737	0.676	0.663	0.713	0.741
SCB-3B	StarCoderBase	10.62	baseline	0.897	0.922	0.926	0.947	0.982
SCB-7B	StarCoderBase	4.33	baseline	0.974	0.991	0.991	0.996	0.996
CGPT-2.7B	Cerebras-GPT	0.67	candidate	0.568	0.537	0.521	0.509	0.508
CGPT-6.7B	Cerebras-GPT	0.69	candidate	0.599	0.542	0.539	0.535	0.519
Neo-125M	GPT-Neo	0.50	candidate	0.500	0.502	0.503	0.503	0.503
OPT-1.3B	OPT	0.56	candidate	0.541	0.554	0.515	0.504	0.508
OPT-2.7B	OPT	0.41	candidate	0.534	0.566	0.561	0.530	0.533

Table 5. Per-model number-comparison accuracy across shot settings. Candidate models are included only in the behavioral comparison because their exact pretraining corpora are not matched in the corpus-frequency analysis.

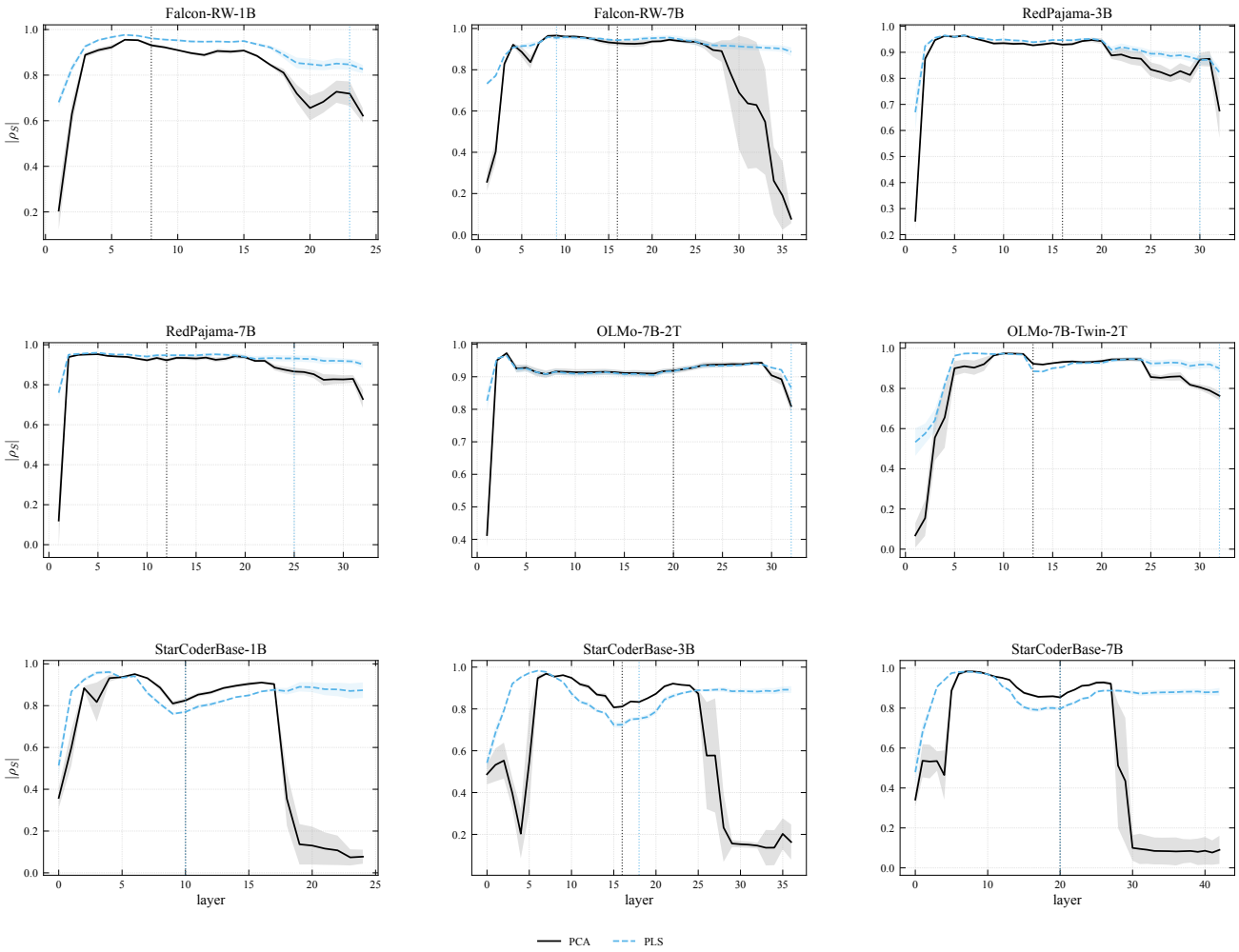

Figure 10. Layer-wise monotonicity of numerical representations using PCA and PLS. The curves show the absolute Spearman correlation $|\rho_s|$ across layers, and the dotted vertical lines indicate the layer with the highest explained variance selected for extracting the final representation.

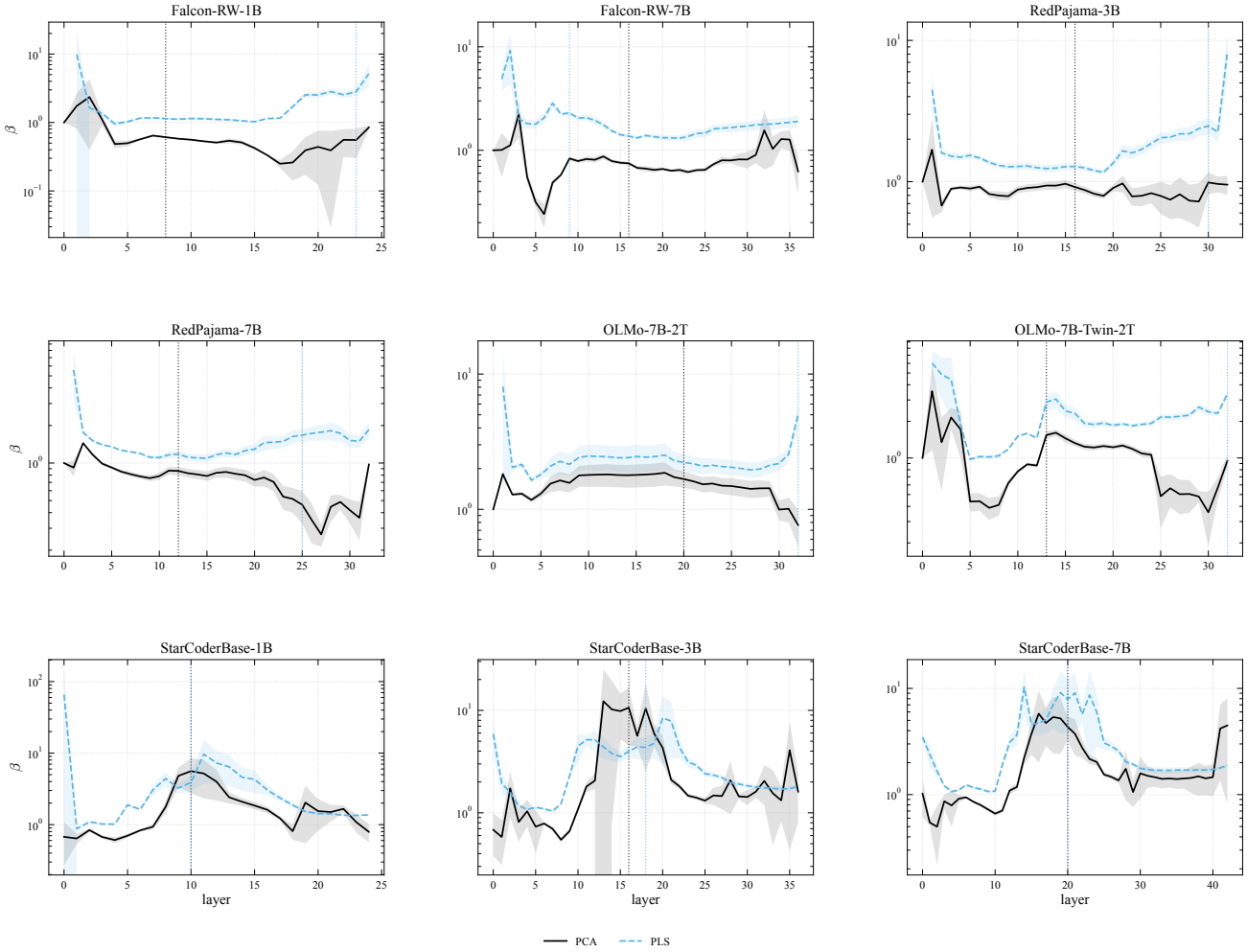

Figure 11. Layer-wise scaling factor β for PCA and PLS numerical projections. The curves show how the estimated number-line compression factor varies across layers, and the dotted vertical lines mark the highest-explained-variance layer used to extract the final representation.