

Do Benchmarks Measure the Same Thing?

Evaluation-Free Benchmark Redundancy Analysis with SAE

Yaoning Wang^{1†}, Jiahao Ying¹, Yubo Ma^{2*}, Yu Wan²,
Dayiheng Liu², Yixin Cao^{1,3*}, Yugang Jiang¹

¹ Fudan University, ² Alibaba Group, ³ Shanghai Innovation Institute
myb477606@alibaba-inc.com, yxcao@fudan.edu.cn

Abstract

The rapid growth of LLM evaluation resources makes exhaustive benchmark curation increasingly difficult. Existing efficient benchmarking methods reduce evaluation cost by selecting representative subsets or predicting full-benchmark performance, but typically require evaluation results from many models and additional training. In this paper, we ask a more fundamental question: to what extent do benchmarks actually measure different things? We study this question at two levels: intra-benchmark redundancy, which supports benchmark compression, and inter-benchmark similarity, which supports evaluation-suite design. We first formalize both notions through a functionalist behavioral criterion, and then propose an evaluation-free proxy based on Sparse Autoencoder (SAE) features for redundancy diagnosis. Our central hypothesis is that behavioral overlap can be approximated by feature overlap. This hypothesis yields level-specific metrics based on feature-coverage saturation within benchmarks and feature-footprint overlap across benchmarks. Experiments show that SAE feature-based redundancy strongly correlates with performance-based redundancy across 17 benchmarks, while SAE-guided subset selection produces more stable and representative subsets than random sampling. SAE feature overlap also predicts inter-benchmark performance similarity across 8 benchmarks. These results show that SAE features can turn benchmark analysis from a costly performance-matrix problem into a feature-footprint problem, enabling evaluation-free benchmark compression and suite curation.

1 Introduction

The evaluation ecosystem for large language models is expanding at a pace that makes manual

benchmark curation increasingly untenable. Public dataset repositories such as Hugging Face now host hundreds of thousands to over a million datasets, with new evaluation resources continuing to appear every day. As a result, model developers increasingly face large and growing evaluation suites: each new checkpoint, training recipe, or alignment method must be assessed across many benchmarks, incurring substantial costs in data construction, benchmark maintenance, and model inference. A growing line of work on efficient benchmarking has therefore sought to reduce this burden by selecting representative subsets or predicting full-benchmark performance from partial evaluations (Polo et al., 2024; Kipnis et al., 2024; Zhang et al., 2026).

In this paper, instead of asking how to evaluate a given benchmark more cheaply, we study a more fundamental problem: to what extent do evaluation benchmarks actually measure different things? We analyze this question at two levels. At the intra-benchmark level, we ask whether many samples within a benchmark provide redundant evidence about model capability, and whether a smaller subset can preserve the behavior of the full benchmark. This level of analysis can directly support benchmark compression and efficient evaluation, since redundant samples may be removed without substantially changing model comparison results. At the inter-benchmark level, we ask whether different benchmarks induce overlapping model behavior, and therefore whether they contribute distinct dimensions to an evaluation suite. This level of analysis can guide evaluation-suite design: benchmarks that provide complementary diagnostic information should be retained, whereas benchmarks that largely duplicate others can be consolidated or deprioritized.

To do so, we formalize this question from a functionalist view. An evaluation sample is characterized not merely by its surface form, task label, or intended capability category, but by the func-

[†] This work is done during Yaoning’s internship at Qwen Team, Alibaba Group.

^{*} Corresponding authors.

tional role it plays in distinguishing models. Operationally, this leads to a behavioral criterion: two samples are redundant if they yield highly consistent correctness patterns across a panel of models, and two benchmarks are similar if they induce similar model-level performance patterns or rankings. Under this view, intra-benchmark redundancy can be measured by how well subsets preserve the full-benchmark ranking, while inter-benchmark similarity can be measured by the correlation between benchmark-level model performances. This provides a natural performance-based ground truth for redundancy and similarity. The challenge, however, is that such a definition is inherently circular for large-scale curation: estimating redundancy requires first evaluating many models on the very samples and benchmarks we hope to compress, compare, or filter.

We break this circularity using Sparse Autoencoder (SAE) features. Our central hypothesis is that behavioral overlap can be approximated by feature overlap: if two samples or benchmarks activate substantially overlapping SAE features in an indicator model, then they are likely to engage similar internal representations and thus provide overlapping diagnostic information. This single hypothesis gives rise to level-specific metrics. At the intra-benchmark level, redundancy is captured by how quickly feature coverage saturates as samples are added. At the inter-benchmark level, similarity is captured by the overlap between benchmark-level feature footprints. Thus, SAE features turn benchmark analysis from a performance-matrix problem into a feature-footprint problem. For example, analyzing 10 benchmarks with 5 models using a conventional behavioral approach requires 50 evaluation runs to construct a model-by-benchmark matrix, whereas our method uses one indicator model to extract feature footprints for the 10 benchmarks, requiring only one pass per benchmark. This design requires no additional training, avoids judge-induced noise for open-ended evaluation, and — validated by our experiments — generalizes from the indicator model’s feature structure to performance-based redundancy and similarity across evaluated models. In this sense, SAE features provide an evaluation-free proxy for behavioral equivalence: they do not replace performance-based definitions, but make them tractable before exhaustive evaluation takes place. We summarize our contributions as follows:

- We formalize benchmark redundancy at two levels: **intra-benchmark redundancy** and **inter-benchmark similarity** from a behavioral equivalence perspective.
- We introduce SAE feature coverage as an evaluation-free proxy for redundancy analysis, which is more efficient and convenient.
- Extensive experiments validate SAE features as effective proxies at both levels: feature redundancy achieves Spearman $\rho \approx 0.85$ with performance-based redundancy across 17 benchmarks and enables stable and representative subset selection, and feature overlap achieves 75.5% Pearson correlation with performance-based similarity across 8 benchmarks.

2 Related Work

Sparse Autoencoders for language models.

Sparse Autoencoders (SAEs) were proposed to disentangle superposed representations into sparse, monosemantic features (Cunningham et al., 2024; Bricken et al., 2023). Subsequent work has scaled SAEs to frontier models (Templeton et al., 2024) and systematically studied their scaling properties and evaluation metrics (Gao et al., 2024). Beyond interpretability, SAE features have been applied to model steering (Templeton et al., 2024) and safety-relevant feature identification. Our work extends SAEs in a new direction: rather than understanding model internals, we use them to characterize the structure of evaluation benchmarks.

Efficient benchmarking. Evaluating models on large benchmarks is resource-intensive. To mitigate this, many works select a representative subset and extrapolate full-dataset performance. For instance, (Vivek et al., 2023) use confidence scores to pick highly correlated samples; (Polo et al., 2024) apply IRT followed by K-Means; (Kipnis et al., 2024) use Fisher Information to filter less discriminative samples; SparseEval (Zhang et al., 2026) directly minimizes the performance discrepancy between subset and full benchmark by optimizing over the evaluation matrix with an MLP. However, these methods typically rely on evaluation data from a large number of models (typically hundreds to thousands), which we avoid using in this work.

3 Preliminaries

Before presenting our framework, we briefly review the key building blocks: Sparse Autoencoders

(SAEs) and the notation we adopt in the paper. Given a benchmark $\mathcal{D} = \{x_1, x_2, \dots, x_K\}$ with K evaluation samples. For a given language model $\mathcal{M}$ equipped with an SAE at a chosen layer, we define the active feature set of sample x_i as:

$$F(x_i) = \{j \in \{1, \dots, D\} : z_j(x_i) > 0\}, \quad (1)$$

where $z_j(x_i)$ is the j -th component of the SAE latent representation of x_i , extracted at the last token position. A positive $z_j(x_i)$ indicates that the j -th SAE feature is activated when the model processes sample x_i . Prior work has shown that SAE features correspond to interpretable, monosemantic concepts (Bricken et al., 2023; Cunningham et al., 2024), and that causally intervening on individual features can modulate the corresponding model behaviors (Templeton et al., 2024); each active feature thus serves as a signal that the model is engaging the corresponding internal representation when generating its response. Note that $z_j(x_i)$ implicitly incorporates the Top- k ReLU activation applied within the SAE encoder; we omit this detail from the notation for brevity. The feature footprint of the entire benchmark is:

$$F(\mathcal{D}) = \bigcup_{i=1}^K F(x_i). \quad (2)$$

In the main text, we consistently use Qwen3-30B-A3B-Base (Yang et al., 2025) and its corresponding SAE (Deng et al., 2026) for all experiments. For the selection of hyperparameters (such as SAE insertion layer and the choice of top- k), as well as ablations conducted with SAEs from other model families, please refer to Appendix A.3 and D, respectively.

4 Measuring Benchmark Redundancy with SAE Features

In this section, we formalize intra-dataset redundancy from the perspective of behavioral equivalence (Section 4.1), and then introduce an efficient SAE feature-based proxy. Through extensive experiments, we demonstrate the reliability of this proxy (Section 4.2) and leverage it to enable more stable and efficient benchmark evaluation (Section 4.3).

4.1 Performance-Based Redundancy

We instantiate intra-dataset redundancy through behavioral equivalence. For each subset, we measure how closely its evaluation outcomes align with

those of the full dataset across the evaluated models. The more uniformly subsets agree with the full-set evaluation, the more redundant the benchmark – since any one of them already reflects what the whole conveys. Aggregating these per-sample consistency scores therefore yields a direct measure of the benchmark’s overall redundancy. Formally, fix a panel of n models $\mathcal{M}_1, \dots, \mathcal{M}_n$. Let $p = [p_1, \dots, p_n]$ denote the vector of average model accuracies on the full benchmark $\mathcal{D} = \{x_1, \dots, x_K\}$, and let $\hat{p}(\mathcal{S}) = [\hat{p}_1, \dots, \hat{p}_n]$ denote the corresponding vector on a subset $\mathcal{S} \subseteq \mathcal{D}$. The rank correlation between $\mathcal{S}$ and $\mathcal{D}$ is:

$$\tau(\mathcal{S}, \mathcal{D}) = \tau(p, \hat{p}(\mathcal{S})), \quad (3)$$

where $\tau(\cdot, \cdot)$ denotes Kendall’s τ . Kendall’s τ is preferred over Spearman’s ρ here because it has a direct combinatorial interpretation: $(\tau + 1)/2$ equals the fraction of model pairs whose relative ordering is preserved by the subset. To characterize the typical behavior on a given subset size, we take expectations:

$$\tau_k = \mathbb{E}_{\mathcal{S} \subseteq \mathcal{D}, |\mathcal{S}|=k} [\tau(\mathcal{S}, \mathcal{D})]. \quad (4)$$

The function $k \mapsto \tau_k$ traces out a rank-correlation curve: it starts near zero for very small k and approaches 1 as $k \rightarrow K$. The shape of this curve encodes how redundant the benchmark is. Intu-

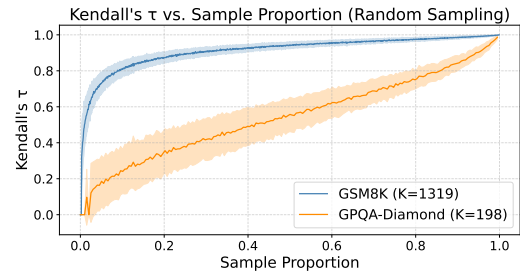


Figure 1: Rank-correlation curves τ_k for GPQA-Diamond ($K = 198$) and GSM8K ($K = 1319$). Horizontal axis: sample proportion k/K ; vertical axis: τ_k (Eq. (4)). At 50% of the data, GPQA-D achieves $\tau \approx 0.55$ while GSM8K already exceeds 0.90, indicating that GSM8K is substantially more redundant.

itively, if the curve rises steeply and saturates early, the benchmark is highly redundant: a small fraction of samples already determines the model ranking; if the curve rises gradually and remains concave throughout, the benchmark has high effective information: each additional sample contributes meaningfully to discriminating models. As an example,

Figure 1 shows the rank-correlation curves for two benchmarks, GPQA-Diamond (Rein et al., 2023) and GSM8K (Cobbe et al., 2021), evaluated using 26 Qwen-3.5-35B-A3B checkpoints, trained on data mixtures with different domain-specific weighting ratios. The contrast is stark: at 50% of the data, GSM8K already achieves $\tau_k > 0.90$, meaning that the vast majority of model pairs are correctly ranked by a random half of the benchmark. In contrast, GPQA-Diamond at 50% reaches only $\tau_k \approx 0.60$, and the curve continues to rise meaningfully as more data is added. This is also consistent with our prior expectations about these benchmarks: GSM8K primarily targets elementary-level math problems, while GPQA-D focuses on high-difficulty STEM reasoning. For elementary math, a small proportion of problems suffices to assess a model’s mastery; whereas complex STEM reasoning typically requires more samples to evaluate a model’s comprehensive capabilities.¹ To more conveniently quantify and compare the degree of such redundancy across datasets, we can aggregate the entire curve into a single scalar:

$$\mathcal{R}(\mathcal{D}) = \text{AUC}(\tau_k) = \frac{1}{K} \sum_{k=1}^K \tau_k. \quad (5)$$

Intuitively, $\mathcal{R}(\mathcal{D})$ corresponds to the area under the rank-correlation curve, with higher values indicating greater redundancy. For instance, $\mathcal{R}(\text{GPQA-D}) = 0.54$ is substantially lower than $\mathcal{R}(\text{GSM8K}) = 0.91$, suggesting that GSM8K admits substantial compression while GPQA-D does not. While $\mathcal{R}(\mathcal{D})$ provides a straightforward measure of redundancy, computing it is prohibitively expensive: computing τ_k for a single value of k already requires sampling many random subsets and running model evaluations on each. Can we estimate redundancy before evaluation takes place?

4.2 SAE-Based Redundancy Analysis

Since performance-based approaches are costly, we therefore seek a measure of behavioral equivalence computable from the inputs alone. SAE features offer a principled basis for this: the feature activations elicited by a sample reflect the internal functionalities the model engages when processing it. Using these activations as a functional map of

¹This gap is not an artifact of GPQA-Diamond’s smaller size. Subsampling GSM8K to GPQA-D’s size (198 examples, 10 random seeds) still yields $\mathcal{R} = 0.794 \pm 0.010$, far above GPQA-D’s 0.542, so the difference reflects intrinsic redundancy rather than sample count (See Appendix F).

the target model, we can measure functional similarity between samples directly in feature space. Analogously to the performance-based redundancy analysis (Section 4.1), we compute the proportion of SAE features covered by each subset relative to those activated by the full dataset. Aggregating the results across subsets then yields an SAE feature-based redundancy. We formalize this notion using SAE features: First, we define the expected coverage at size k , which corresponds to Eq. (4):

$$c_k = \mathbb{E}_{\mathcal{S} \subseteq \mathcal{D}, |\mathcal{S}|=k} \left[\frac{|F(\mathcal{S})|}{|F(\mathcal{D})|} \right]. \quad (6)$$

The curve $k \mapsto c_k$ is the **feature-coverage curve**. Its shape parallels the rank-correlation curve $k \mapsto \tau_k$: a benchmark whose samples share many features will see c_k saturate quickly (high redundancy), while a benchmark with diverse feature activation patterns will show slow, steady growth. Similar to Eq. (5), we aggregate the feature-coverage curve into a scalar metric:

$$\text{AUC}(c_k) = \frac{1}{K} \sum_{k=1}^K c_k. \quad (7)$$

However, the raw coverage AUC alone does not capture absolute feature diversity. Consider two benchmarks of the same size K whose coverage both grow linearly ($c_k = k/K$), yielding $\text{AUC} = 0.5$ in both cases. Suppose the first activates $|F(\mathcal{D})| = 1,000$ distinct features in total while the second activates 2,000. Both have the same AUC, yet the second benchmark clearly probes a broader range of capabilities; it should be considered less redundant. The coverage curve, being normalized to $[0, 1]$, erases this difference in absolute scale. To restore it, we multiply the AUC by a growth-rate correction $K/|F(\mathcal{D})|$. Intuitively, $|F(\mathcal{D})|/K$ measures how many new features each sample contributes on average: a benchmark that activates 2,000 features over K samples has twice the per-sample growth rate of one that activates 1,000, and should therefore receive a lower redundancy score. Multiplying by the reciprocal $K/|F(\mathcal{D})|$ achieves exactly this, yielding the **feature redundancy**:

$$\hat{\mathcal{R}}(\mathcal{D}) = \text{AUC}(c_k) \cdot \frac{K}{|F(\mathcal{D})|} = \frac{\sum_{k=1}^K c_k}{|F(\mathcal{D})|}. \quad (8)$$

To verify whether feature redundancy serves as an effective proxy for performance redundancy as defined in Section 4.1, we select 26 pre-trained

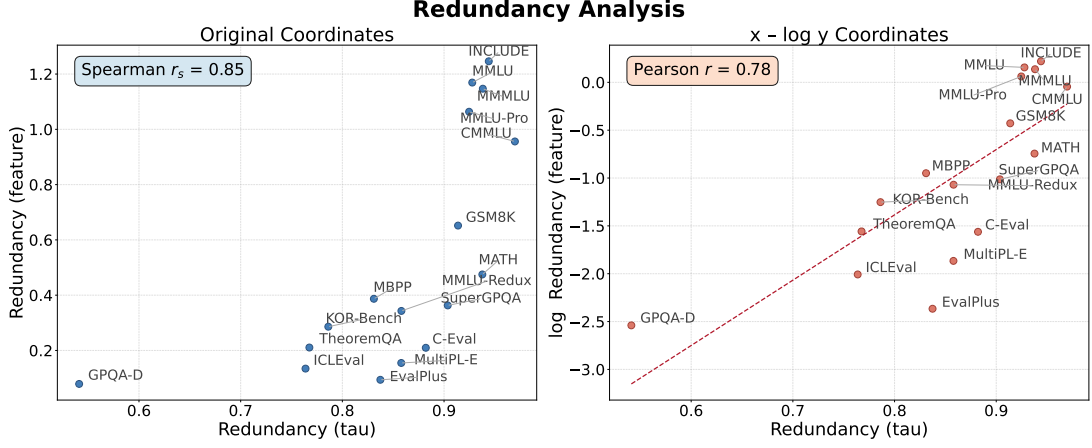


Figure 2: Spearman rank correlation between performance-based redundancy $\mathcal{R}(\mathcal{D})$ and feature redundancy $\hat{\mathcal{R}}(\mathcal{D})$ across 17 benchmarks (Spearman $\rho \approx 0.85$), suggesting that $\hat{\mathcal{R}}(\mathcal{D})$ serves as a reliable proxy for $\mathcal{R}(\mathcal{D})$.

checkpoints with varying training steps and data mixture ratios, and evaluate the correlation between $\mathcal{R}(\mathcal{D})$ and $\hat{\mathcal{R}}(\mathcal{D})$ across 17 widely-used benchmarks spanning general knowledge, mathematics, coding, multilingual understanding, and in-context reasoning. For the full list of the used benchmarks, please refer to Appendix A.2. Key observations from the analysis (Figure 2): **1)** The Spearman rank correlation between $\mathcal{R}(\mathcal{D})$ and $\hat{\mathcal{R}}(\mathcal{D})$ across 17 benchmarks is $\rho \approx 0.85$, suggesting that **feature redundancy may serve as a reasonable evaluation-free proxy for performance-based redundancy.** **2)** A smaller sample count does not necessarily imply lower redundancy. For example, GSM8K (1,319 samples) exhibits substantially higher redundancy (both performance-based and feature-based) than MMLU-Redux (3,000 samples). This further demonstrates that our metric is able to mitigate the influence of the dataset’s absolute size. **3)** Early general-purpose benchmarks often suffer from considerable redundancy. For instance, MMLU, MMLU-Pro, and CMMLU are all located in the upper-right region of the figure, indicating relatively high redundancy. In contrast, the more recent SuperGPQA was explicitly designed to cover a broader range of domains. Although it contains more samples (26K) than the aforementioned benchmarks (10-15K), it achieves significantly lower redundancy. This suggests that deliberate diversity in benchmark construction translates into measurably lower redundancy. Together, these observations validate $\hat{\mathcal{R}}(\mathcal{D})$ as a reliable, evaluation-free measure of benchmark redundancy. Beyond serving as a design-time diagnostic, our analysis also offers practical guidance for choosing among existing ones — favoring benchmarks with

lower redundancy when evaluation budget is limited or when broader capability coverage is desired.

4.3 Feature-guided Subset Sampling

Section 4.2 established that feature-based redundancy is a reliable, evaluation-free diagnostic for benchmark design. We now ask whether the same signal can be put to active use: can we exploit it to select a compact, low-redundancy subset that supports efficient evaluation? Since each feature corresponds to a distinct functionality engaged by the benchmark, a subset that preserves the feature distribution of the full dataset should preserve its evaluation signal as well. This motivates a natural objective: select the smallest subset whose feature distribution approximates that of the full benchmark. To characterize feature distribution, we define feature frequency vector $f(\mathcal{D}) \in \mathbb{R}^D$ as

$$[f(\mathcal{D})]_i = \frac{1}{|\mathcal{D}|} \sum_{x \in \mathcal{D}} \mathbb{1}(i \in F(x)), \quad (9)$$

where i is the feature index, $F(x)$ is the feature set activated by sample $x \in \mathcal{D}$, and $\mathbb{1}(\cdot)$ is the indicator function. The optimization objective can be formalized as

$$\hat{\mathcal{S}} = \arg \min_{\mathcal{S}} \mathcal{L}(f(\mathcal{S}), f(\mathcal{D})), \quad (10)$$

where we adopt the L2 loss. Since obtaining an exact solution to this combinatorial optimization problem is intractable, we employ a greedy algorithm to approximate the solution, where at each step, the sample that minimizes the loss is selected and added to the subset.

Experimental setup. We run SAE-guided selection on the redundant benchmarks identified in Figure 2 (i.e., the 7 benchmarks in the upper-right

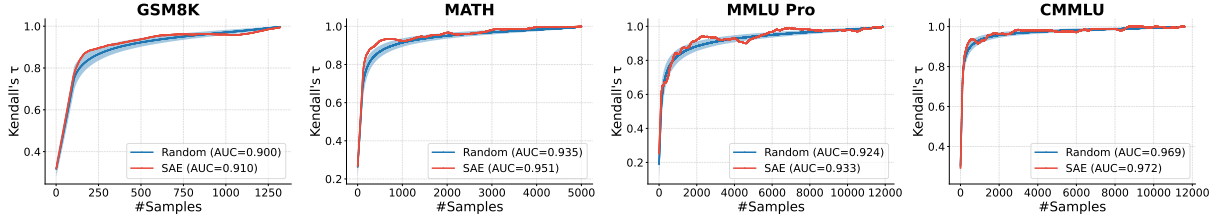


Figure 3: Random vs. SAE-Guided sampling across 4 benchmarks. For results on other benchmarks, please refer to Figure 7 in the Appendix. Blue line: Kendall’s τ under random sampling ($\pm$ std in shaded region). Red line: Kendall’s τ under SAE-guided sampling ($\pm$ std in shaded region). SAE-guided subset consistently achieves higher mean τ and lower variance, with the advantage most pronounced at small subset sizes.

quadrant: see Appendix Figure 7 for the full list) and compare against random subset selection as a baseline. For more details on the algorithm settings and analysis, please refer to Appendix B. The results (Figure 3) reveal several consistent patterns:

1) SAE subsets yield higher correlation, especially at small k . Overall, the expected Kendall’s τ under SAE-guided sampling equals or exceeds that of random sampling across subset sizes: $\mathbb{E}[\tau_k^{\text{SAE}}] \geq \mathbb{E}[\tau_k^{\text{random}}]$. This advantage is most pronounced when k/K is small (roughly $k/K < 0.3$), where SAE-guided sampling yields up to 5-10% higher Kendall’s τ than random sampling. This suggests that by better approximating the feature distribution of the full benchmark, our method achieves lower redundancy with the same number of samples, thereby attaining higher rank correlation. Beyond 30% of the benchmark, both methods converge toward the full-set evaluation result as more samples are included. This concentration of gains in the small-subset regime is consistent with recent findings that random sampling is competitive on average while structured selection helps most when the subset is small (Zhang et al., 2025; Yauney et al., 2026). **2) SAE subsets are more stable than random subsets, with near-zero variance.** The standard deviation of τ_k under SAE subsets is nearly zero, remaining consistently lower than that observed under random sampling. This is visible in Figure 3 as the narrower shaded region around the SAE curves (red) compared to the random curves (blue). In contrast, random sampling at small subset ratios can exhibit substantial variance, for example, on MATH ($K = 5,000$) with $k = 200$, the estimated Kendall’s tau under random sampling is 0.79 ± 0.05 , requiring approximately 150 independent draws to achieve a stable mean estimate. The high variance prevents random sampling from reliably preserving the model ranking between the selected subset and the full benchmark. In contrast,

SAE feature-guided sampling produces a stable subset that better maintains behavioral consistency with the original dataset. Taken together, our sampling method **provides an evaluation-free and stable approach to subset selection that effectively reduces dataset redundancy.**

5 Inter-Benchmark Similarity Analysis

Section 4 focused on the internal structure of a single benchmark: how redundant it is, and how to select maximally informative subsets. In this section, we extend the analysis to the relationships across multiple benchmarks. In Section 5.1, we first define inter-dataset similarity from the perspective of behavioral equivalence. We then propose a proxy based on SAE features in Section 5.2, and subsequently validate the effectiveness of SAE features as a proxy in Section 5.3.

5.1 Performance-Based Similarity

We begin by establishing a performance-based ground truth. Fix a panel of models $\mathcal{M}_1, \dots, \mathcal{M}_n$; for two benchmarks $\mathcal{D}_1$ and $\mathcal{D}_2$, we compute the average performance of each model on both benchmarks respectively, and quantify the similarity between the two datasets via the consistency of their average performance (i.e., performance correlation), which we formalize as follows. Let $p = [p_1, \dots, p_n]$ and $q = [q_1, \dots, q_n]$ be the vectors of average model accuracies on $\mathcal{D}_1$ and $\mathcal{D}_2$, respectively. The **performance-based similarity** between $\mathcal{D}_1$ and $\mathcal{D}_2$ is:

$$\rho(\mathcal{D}_1, \mathcal{D}_2) = \text{corr}(p, q), \quad (11)$$

where $\text{corr}(\cdot, \cdot)$ can be Pearson, Spearman, or Kendall correlation. When $\rho(\mathcal{D}_1, \mathcal{D}_2)$ is high, models that perform well on $\mathcal{D}_1$ also tend to perform well on $\mathcal{D}_2$ — they rank models similarly. When ρ is low, the benchmarks induce different rankings, suggesting that they probe different capability distributions: a model’s performance on $\mathcal{D}_1$ is not

predictive of its performance on $\mathcal{D}_2$. In this work, we adopt the Pearson correlation coefficient as the similarity metric. This choice is motivated by its direct relationship with the coefficient of determination (R^2) in linear regression, making it a natural measure of how well performance on one benchmark predicts performance on another.

While Pearson correlation measures the extent to which performance on one benchmark predicts performance on another, the resulting similarity may conflate multiple factors. In particular, two benchmarks can appear highly correlated simply because both are strongly associated with overall model quality. As model capability improves, performance tends to increase across many benchmarks simultaneously, even when those benchmarks evaluate substantially different skills. Consequently, raw benchmark correlation may reflect not only benchmark-specific capability overlap, but also a shared dependence on general model ability. To disentangle these two effects, we employ partial correlation analysis. Specifically, we use a general and established benchmark as a proxy for overall model quality, and measure the residual correlation between benchmarks after removing the component explained by this shared factor. In our experiments, we use MMLU as the control variable due to its broad coverage and strong correlation with general model performance. Given two benchmarks $\mathcal{D}_1$ and $\mathcal{D}_2$, we define the **corrected performance similarity** as their partial correlation conditioned on MMLU, denoted by $\rho(\mathcal{D}_1, \mathcal{D}_2 \mid \text{MMLU})$. The full definition and an illustrative visualization are provided in Appendix C. Intuitively, this metric measures the correlation between the residual performance patterns of $\mathcal{D}_1$ and $\mathcal{D}_2$ after the variation attributable to overall model quality has been regressed out. In the following analysis, we report both the raw similarity and the corrected similarity. We defer the empirical results to Section 5.3, where we compare performance-based similarity against the feature-based proxy introduced next.

5.2 Feature Overlap Between Benchmarks

As with intra-benchmark redundancy, we seek an evaluation-free proxy for performance-based similarity. The natural choice is **feature overlap**: two benchmarks are similar if they activate largely the same SAE features, and dissimilar if their feature footprints have little intersection. To measure the potential inclusion relationship between benchmarks, given two benchmarks $\mathcal{D}_1$ and $\mathcal{D}_2$, we

define an asymmetric feature overlap metric that quantifies the extent to which the features of $\mathcal{D}_1$ are covered by $\mathcal{D}_2$:

$$\text{overlap}(\mathcal{D}_1, \mathcal{D}_2) = \frac{|F(\mathcal{D}_1) \cap F(\mathcal{D}_2)|}{|F(\mathcal{D}_1)|}. \quad (12)$$

The asymmetry is informative: For instance, we expect GSM8K (elementary math) to have a large fraction of its features covered by MATH (more advanced math benchmark), while the reverse should be less true, i.e. $\text{overlap}(\text{GSM8K}, \text{MATH}) > \text{overlap}(\text{MATH}, \text{GSM8K})$. To verify that feature overlap captures semantically meaningful patterns, we compute the pairwise overlap matrix of eight representative benchmarks (Figure 4 (a)). Several well-known relationships are clearly recovered:

- **Mathematical hierarchy.** GSM8K’s features are largely covered by MATH, but not vice versa—consistent with the expectation that competition-level math subsumes elementary arithmetic while also requiring deeper capabilities such as proof techniques and algebraic manipulation.
- **Code cluster.** Code benchmarks (EvalPlus, MBPP, MultiPL-E) form a tight cluster with high pairwise overlaps, reflecting the shared code-generation capabilities they evaluate despite differences in language coverage and difficulty.
- **Knowledge subsumption.** Broad knowledge benchmarks (MMLU-Pro, SuperGPQA) cover 56–68% of the features activated by specialized benchmarks such as TheoremQA, consistent with the intuition that wide-coverage evaluations encompass many of the domain-specific features probed by focused ones.

These patterns confirm that SAE feature overlap captures semantically meaningful relationships between benchmarks, motivating its use as a proxy for performance-based similarity in the next section.

5.3 Connecting Overlap to Performance Similarity

The asymmetric overlap defined above is well-suited for detecting containment relationships, but performance-based similarity $\rho(\mathcal{D}_1, \mathcal{D}_2)$ is inherently symmetric. To enable a direct quantitative comparison between the two, we introduce a symmetric variant of the overlap metric. We adopt a min-based normalization: Given two benchmarks

Feature Overlap Analysis

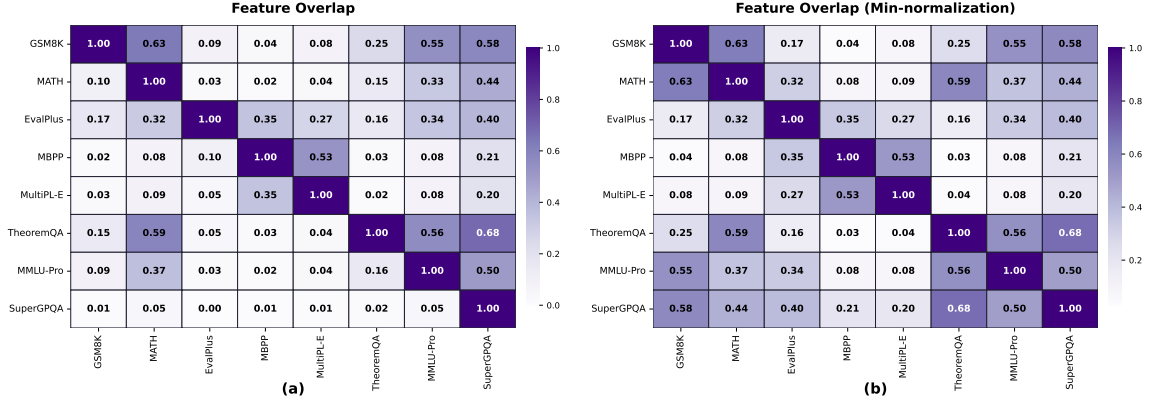


Figure 4: Feature overlap matrix for eight benchmarks. Entry (i, j) gives $\text{overlap}(\mathcal{D}_i, \mathcal{D}_j)$ (a) and $\text{overlap}_{\min}(\mathcal{D}_i, \mathcal{D}_j)$ (b). The matrix reveals intuitive containment relationships: GSM8K is largely covered by MATH, code benchmarks form a tight cluster, and broad knowledge benchmarks subsume specialized ones.

$\mathcal{D}_1$ and $\mathcal{D}_2$, their symmetric feature overlap is:

$$\text{overlap}_{\min}(\mathcal{D}_1, \mathcal{D}_2) = \frac{|F(\mathcal{D}_1) \cap F(\mathcal{D}_2)|}{\min(|F(\mathcal{D}_1)|, |F(\mathcal{D}_2)|)}. \quad (13)$$

The min-denominator ensures that the metric is high when the smaller benchmark’s features are largely contained in the larger one, capturing the intuition of capability subsumption. Figure 4 (b) shows the symmetrized overlap matrix. Compared to the asymmetric version, the diagonal blocks (math, code, knowledge) remain clearly visible, but the cross-block values are adjusted to reflect the bidirectional relationship. We now test the central hypothesis: does feature overlap predict performance-based similarity? Crucially, we evaluate this prediction both for the raw $\rho(\mathcal{D}_1, \mathcal{D}_2)$ and for the corrected $\rho(\mathcal{D}_1, \mathcal{D}_2 \mid \text{MMLU})$ defined in Section 5.1. We compute pairwise performance-based similarities for all $\binom{8}{2} = 28$ benchmark pairs using the 26 checkpoints, and compare these against the corresponding $\text{overlap}_{\min}$ values. As shown in Table 1, the correlations are positive and significant (Pearson 68.4%, Spearman 60.7%), but not overwhelmingly strong, likely due to the general ability confound discussed before.

Similarity	$\rho(\mathcal{D}_1, \mathcal{D}_2)$	$\rho(\mathcal{D}_1, \mathcal{D}_2 \mid \text{MMLU})$
Pearson	68.4	75.5
Spearman	60.7	71.3

Table 1: Correlation between symmetric feature overlap and performance-based similarity ρ , before and after controlling for general ability with MMLU.

To isolate the relationship between feature overlap and benchmark-specific capability patterns, we

replace the raw similarity with the corrected similarity (taking MMLU as the control variable). After partialing out general ability, the correlation between feature overlap and performance similarity increases substantially in both metrics, as detailed in Table 1: the Pearson correlation rises to **75.5%**, and the Spearman correlation to **71.3%**. This notable improvement provides strong evidence that **feature overlap captures benchmark-specific capability similarity beyond the trivial effect of overall model quality**.

Implications for evaluation suite design. This result has a direct practical implication: feature overlap can guide evaluation suite design without any model evaluation. Benchmarks with low overlap probe distinct capabilities and should both be retained; benchmarks with high overlap are candidates for consolidation. For example, the asymmetric overlap analysis shows that 63% of GSM8K’s features are already covered by MATH, suggesting that an evaluation suite containing MATH can safely drop GSM8K with little loss of discriminative information. Conversely, a benchmark whose feature footprint has low overlap with all current suite members may probe capabilities not yet covered, making it a natural candidate for inclusion.

6 Conclusion

In this paper, we introduce a formal framework for quantifying benchmark redundancy from the perspective of behavioral equivalence, operating at two levels: the **intra-benchmark** level and the **inter-benchmark** level. Building on these performance-based metrics, we propose SAE feature-based met-

rics as proxies and validate their effectiveness at both levels. We further show that SAE-guided subset selection yields more stable and representative subsets than random sampling. We hope our work provides a new perspective for measuring benchmark redundancy, enabling more efficient evaluation as the benchmark landscape continues to grow.

7 Limitations

Benchmark Quality as Assumed Ground Truth.

Throughout the analysis in Section 4, we treat the full benchmark as ground truth and do not question whether its original distribution is “correct” or reasonable. The original benchmark, with all its potential flaws, defines the population of interest. Redundancy arises from factors such as lowering evaluation variance, convenience in sample collection, or accidental duplication — not from a mismatch with some external ideal. More broadly, our framework addresses redundancy within and across benchmarks, but does not perform *meta-evaluation* of individual benchmark quality — whether its items are well-constructed, its difficulty distribution is appropriate, or its coverage of the target domain is adequate.

Dependence on the model panel. The performance-based metrics (e.g., τ_k) are functions of both the benchmark distribution and the model capability distribution: $\tau = \tau(\mathcal{D}, \{\mathcal{M}_i\})$, while we wish to measure the relationship between datasets themselves, minimizing the bias introduced by model selection. In practice, we mitigate this by using a large and diverse model panel. Because our main panel comprises checkpoints from a single pre-training series, we additionally validate robustness on an expanded panel of 34 models spanning four families (Gemma3, Llama-3.1, Qwen2.5, Qwen3) and scales (4B–14B). The intra-benchmark Spearman correlation is unchanged ($\rho = 0.85$) and the inter-benchmark correlation is essentially stable ($75.5\% \rightarrow 75.2\%$), indicating that the *relative* benchmark rankings our metrics rely on generalize beyond a single model family (full results in Appendix E). The absolute values of the performance-based metric can still shift with panel composition (e.g., the Pearson correlation drops from 0.82 to 0.72), so we continue to recommend a large and diverse panel when absolute scores matter.

Marginal gains of SAE-guided sampling. SAE-guided sampling consistently outperforms random sampling in mean rank correlation, but the improvement concentrates at small subset sizes and diminishes as the subset ratio grows. The primary practical value of SAE-guided sampling lies in **variance reduction**: while random sampling can match the mean performance given enough independent draws, a single SAE-guided subset achieves near-zero variance, eliminating the need for repeated sampling and providing a reliable, one-shot compression of the benchmark.

Acknowledgements

This work is supported by New Generation Artificial Intelligence-National Science and Technology Major Project No. 2025ZD0124102.

References

- Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. 2021. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*.
- Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermy, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, et al. 2023. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*.
- Federico Cassano, John Gouwar, Daniel Nguyen, Sydney Nguyen, Luna Phipps-Costin, Donald Pinckney, Ming-Ho Yee, Yangtian Zi, Carolyn Jane Anderson, Molly Q Feldman, et al. 2022. Multipl-e: A scalable and extensible approach to benchmarking neural code generation. *arXiv preprint arXiv:2208.08227*.
- Wenhu Chen, Ming Yin, Max Ku, Pan Lu, Yixin Wan, Xueguang Ma, Jianyu Xu, Xinyi Wang, and Tony Xia. 2023. Theoremqa: A theorem-driven question answering dataset. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7889–7901.
- Wentong Chen, Yankai Lin, ZhenHao Zhou, HongYun Huang, Yantao Jia, Zhao Cao, and Ji-Rong Wen. 2025. Icleval: evaluating in-context learning ability of large language models. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 10398–10422.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

- Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. 2024. Sparse autoencoders find highly interpretable features in language models. *Proceedings of the International Conference on Learning Representations (ICLR)*.
- Boyi Deng, Xu Wang, Yaoning Wang, Yu Wan, Yubo Ma, Baosong Yang, Haoran Wei, Jialong Tang, Huan Lin, Ruize Gao, Tianhao Li, Qian Cao, Xuancheng Ren, Xiaodong Deng, An Yang, Fei Huang, Dayiheng Liu, and Jingren Zhou. 2026. [Qwen-Scope: Turning sparse features into development tools for large language models](#). *Preprint*, arXiv:2605.11887.
- Xinrun Du, Yifan Yao, Kaijing Ma, Bingli Wang, Tianyu Zheng, King Zhu, Minghao Liu, Yiming Liang, Xiaolong Jin, Zhenlin Wei, et al. 2025. Supergqa: Scaling llm evaluation across 285 graduate disciplines. *arXiv preprint arXiv:2502.14739*.
- Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeff Wu. 2024. Scaling and evaluating sparse autoencoders. *arXiv preprint arXiv:2406.04093*.
- Aryo Pradipta Gema, Joshua Ong Jun Leang, Giwon Hong, Alessio Devoto, Alberto Carlo Maria Mancino, Rohit Saxena, Xuanli He, Yu Zhao, Xiaotang Du, Mohammad Reza Ghasemi Madani, et al. 2025. Are we done with mmlu? In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5069–5096.
- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.
- Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *NeurIPS*.
- Yuzhen Huang, Yuzhuo Bai, Zhihao Zhu, Junlei Zhang, Jinghan Zhang, Tangjun Su, Junteng Liu, Chuancheng Lv, Yikai Zhang, Jiayi Lei, Yao Fu, Maosong Sun, and Junxian He. 2023. C-eval: A multi-level multi-discipline chinese evaluation suite for foundation models. In *Advances in Neural Information Processing Systems*.
- Alex Kipnis, Konstantinos Voudouris, Luca M Schulze Buschoff, and Eric Schulz. 2024. metabench – A Sparse Benchmark to Measure General Ability in Large Language Models. *arXiv preprint arXiv:2407.12844*.
- Haonan Li, Yixuan Zhang, Fajri Koto, Yifei Yang, Hai Zhao, Yeyun Gong, Nan Duan, and Timothy Baldwin. 2023. [Cmmlu: Measuring massive multitask language understanding in chinese](#). *Preprint*, arXiv:2306.09212.
- Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. 2023. [Is your code generated by chat-GPT really correct? rigorous evaluation of large language models for code generation](#). In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Kaijing Ma, Xinrun Du, Yunran Wang, Haoran Zhang, Zhoufutu Wen, Xingwei Qu, Jian Yang, Jiaheng Liu, Minghao Liu, Xiang Yue, et al. 2024. Kor-bench: Benchmarking language models on knowledge-orthogonal reasoning tasks. *arXiv preprint arXiv:2410.06526*.
- OpenAI. 2024. [Multilingual massive multitask language understanding](#).
- Lorenzo Pacchiardi, Lucy G Cheke, and José Hernández-Orallo. 2024. 100 instances is all you need: predicting the success of a new llm on unseen data by testing on a few instances. *arXiv preprint arXiv:2409.03563*.
- Felipe Maia Polo, Lucas Weber, Leshem Choshen, Yuekai Sun, Gongjun Xu, and Mikhail Yurochkin. 2024. tinybenchmarks: evaluating llms with fewer examples. *arXiv preprint arXiv:2402.14992*.
- David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. 2023. Gpqa: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022*.
- Angelika Romanou, Negar Foroutan, Anna Sotnikova, Zeming Chen, Sree Harsha Nelaturu, Shivalika Singh, Rishabh Maheshwary, Micol Altomare, Mohamed A Haggag, Alfonso Amayuelas, et al. 2024. Include: Evaluating multilingual language understanding with regional knowledge. *arXiv preprint arXiv:2411.19799*.
- Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, et al. 2024. Scaling monosemanticity: Extracting interpretable features from Claude 3 Sonnet. *Transformer Circuits Thread*.
- Rajan Vivek, Kawin Ethayarajh, Diyi Yang, and Douwe Kiela. 2023. Anchor points: Benchmarking models with much fewer examples. *arXiv preprint arXiv:2309.08638*.
- Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyang Jiang, et al. 2024. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *Advances in Neural Information Processing Systems*, 37:95266–95290.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge,

Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Gregory Yauney, Shahzaib Warraich, and Swabha Swayamdipta. 2026. [How reliable is language model micro-benchmarking?](#) In *International Conference on Learning Representations*, volume 2026, pages 27389–27412.

Guanhua Zhang, Florian E. Dörner, and Moritz Hardt. 2025. [How benchmark prediction from fewer data misses the mark.](#) In *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pages 135638–135678. Curran Associates, Inc.

Taolin Zhang, Hang Guo, Wang Lu, Tao Dai, Shu-Tao Xia, and Jindong Wang. 2026. Sparseeval: Efficient evaluation of large language models by sparse optimization. *arXiv preprint arXiv:2602.07909*.

A Experimental Details

This appendix describes the experimental setup underlying all results presented in the main text.

A.1 Model Panel

We use a panel of 26 models from the Qwen-3.5-35B-A3B pre-training series. These models share the same architecture but differ in:

- **Pre-training step.** Models are sampled at different stages of pre-training, capturing the trajectory from the early stage of training to convergence.
- **Data mixture ratio.** Different checkpoints correspond to different proportions of code, math, multilingual, and general text data.

By using a single model family with varying training configurations, we obtain a panel with naturally varying capability profiles: early checkpoints are weak across the board, while later checkpoints develop differentiated strengths (e.g., strong in STEM but weaker in entertainment-domain knowledge). This diversity is essential for meaningful rank-correlation analysis.

A.2 Benchmarks

To validate the effectiveness of our SAE-based analysis method, we conduct experiments across 17 widely-used benchmarks spanning general knowledge, mathematics, coding, multilingual understanding, and in-context reasoning, as listed below.

- **General Tasks:** MMLU (Hendrycks et al., 2020), MMLU-Redux (Gema et al., 2025), MMLU-Pro (Wang et al., 2024), SuperGPQA (Du et al., 2025), C-Eval (Huang et al., 2023), CMMLU (Li et al., 2023).
- **STEM & Math Tasks:** GSM8K (Cobbe et al., 2021), MATH (Hendrycks et al., 2021), GPQA-Diamond (Rein et al., 2023), TheoremQA (Chen et al., 2023).
- **Code Tasks:** MBPP (Austin et al., 2021), EvalPlus (Liu et al., 2023), MultiPL-E (Casano et al., 2022).
- **Multilingual Tasks:** MMMLU (OpenAI, 2024), INCLUDE (Romanou et al., 2024).
- **In-Context Reasoning Tasks:** KOR-Bench (Ma et al., 2024), ICLEval (Chen et al., 2025).

A.3 SAE Configuration

Base Model & SAE. In the main text, we use Qwen3-30B-A3B-Base (Yang et al., 2025) and its corresponding SAE (Deng et al., 2026) for all experiments.

Hyperparameter Selection. We ablate two factors related to SAE feature extraction: 1) the layer at which the SAE is inserted; and 2) the value of k in the Top- k ReLU activation of the SAE. We propose a simple proxy task to select the optimal hyperparameters: our goal is for the SAE features to be sensitive to the type of capability being examined in the data, i.e., when controlling for the number of samples, a larger number of categories should activate a more diverse set of features. To this end, we fix the total number of samples to $n = 1,500$ on MMLU while varying the number of categories from 1 to 57, and select the hyperparameters that maximize the Spearman rank correlation with the number of categories. The ablation results are shown in Figure 5 and 6. For the selection of SAE insertion layer, the results in Figure 5 suggest that layer 12 and layer 24 yield higher correlation coefficients than layer 36; however, we find in practice that earlier layers (i.e., layer 12 and 24) tend to assign disproportionately large activations to certain instructions (e.g., *Let's think step by step.*). Taking both factors into consideration, we adopt layer 36 as the SAE insertion layer.

For the selection of top- k (Figure 6), we find that a larger top- k tends to provide more information, as indicated by a higher Spearman rank correlation. Therefore, we set $k = 100$ for the top- k activation.

B SAE-Guided Subset Sampling

Random Baseline. The random baseline is already a strong baseline in terms of the mean, because it naturally preserves the data distribution, but it suffers from very large variance. Therefore, in Section 4.3, we use the Monte Carlo method to estimate the mean and standard deviation of the random approach. We pre-compute the evaluation results for all samples in the dataset. For τ_k , we repeatedly sample subsets $\mathcal{S}_t$ of size k ($|\mathcal{S}_t| = k$) until the sample mean

$$\hat{\tau}_k = \frac{1}{T} \sum_{t=1}^T \tau(\mathcal{S}_t, \mathcal{D})$$

fluctuates by less than 0.001, so as to obtain a stable estimate of the mean and standard deviation.

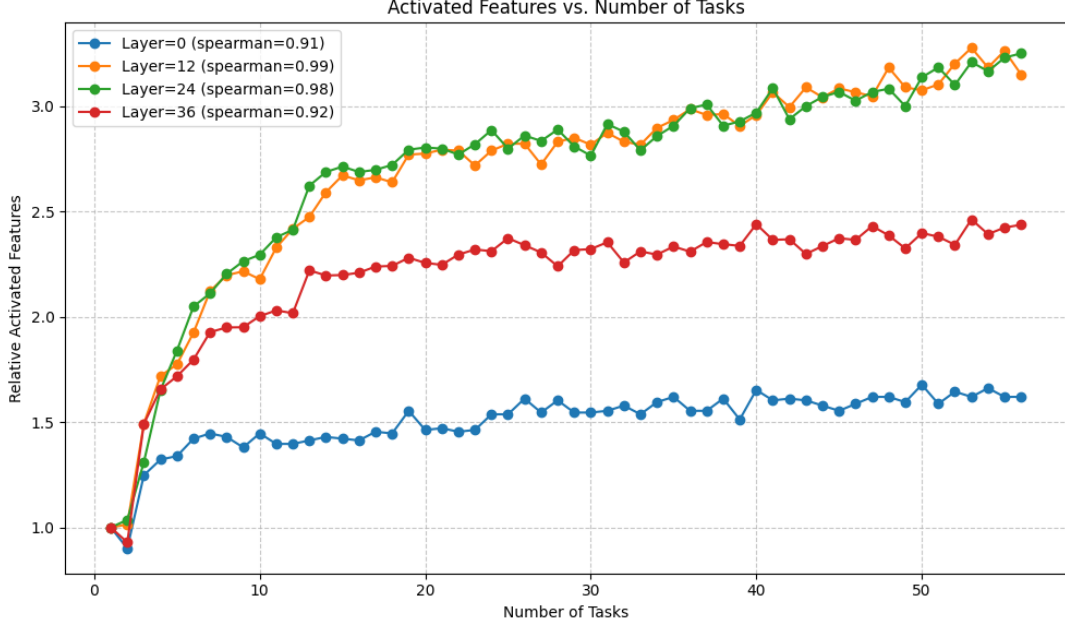


Figure 5: Ablation study on the SAE insertion layer, with k fixed at 100. The Spearman rank correlation between the number of active features and the number of categories on MMLU is reported for each layer.

SAE-Guided Sampling. Since we solve the optimization problem in Eq. (10) using a greedy algorithm, and due to the sparsity of SAE features, ties between two samples may occur during the solution iterations. To eliminate the influence of random factors such as the ordering of samples, we randomly shuffle the samples each time and repeat the greedy algorithm 10 times. We observe that this solution process is essentially deterministic (standard deviation almost 0, Figure 3 and Figure 7), thus providing a more stable alternative to random sampling.

Computational efficiency. A crucial advantage of the greedy approach is that it produces a nested sequence of subsets: $\mathcal{S}_1 \subset \mathcal{S}_2 \subset \dots \subset \mathcal{S}_K = \mathcal{D}$. This means a single pass ($\mathcal{O}(K)$) of the algorithm yields subsets at all sizes simultaneously — unlike KMeans-based methods (Polo et al., 2024; Pacchiardi et al., 2024), where each target subset size k requires an independent clustering run. The total time complexity is $\mathcal{O}(K^2)$ when pre-computing feature sets, which is tractable for benchmarks with up to tens of thousands of samples.

C Partial Correlation Correction for General Model Ability

Motivation. When evaluating whether two benchmarks $\mathcal{D}_1$ and $\mathcal{D}_2$ probe similar capabilities, a natural starting point is to measure the correlation

between model performances on them. However, model performances across diverse benchmarks are invariably influenced by a common factor: *general model ability* (often denoted g in the psychometric literature). A model that is stronger overall—by virtue of scale, training data, or algorithmic improvements—tends to outperform a weaker model on virtually any reasonably constructed evaluation. Consequently, even two benchmarks that test completely disjoint capabilities (e.g., pure mathematics and literary translation) can exhibit a high positive performance correlation, simply because all model scores co-vary with this dominant general-ability dimension.

If the goal is to measure whether $\mathcal{D}_1$ and $\mathcal{D}_2$ probe the same specific capabilities (beyond overall quality), the raw performance correlation is misleading. The confound can be removed by statistically controlling for a suitable proxy of general ability.

Method: partial correlation. Let $\mathcal{G}$ be a benchmark chosen to represent general model ability (in our work we use MMLU, a wide-coverage knowledge benchmark that strongly correlates with overall model quality). For a panel of n models, we have performance vectors p , q , and g on $\mathcal{D}_1$, $\mathcal{D}_2$, and $\mathcal{G}$, respectively. The linear effect of general ability can be removed by first regressing p and q

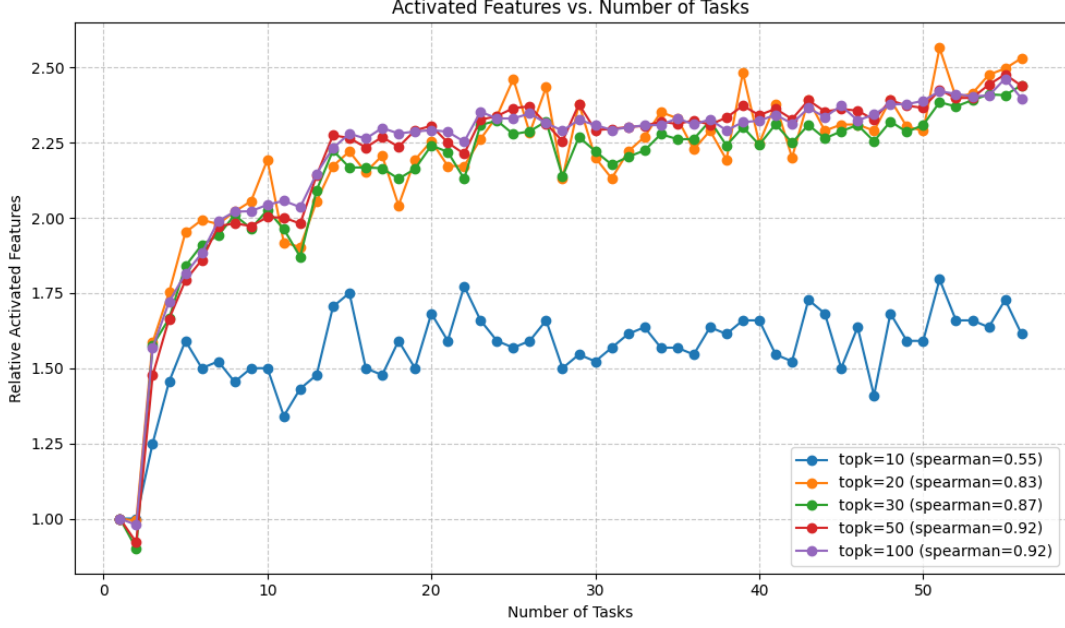


Figure 6: Ablation study on the Top- k activation parameter of the SAE, with the insertion layer fixed at 32. The Spearman rank correlation between the number of active features and the number of categories on MMLU is reported for each value of k .

on g and keeping the residuals:

$$e_1 = p - \hat{\alpha}_1 - \hat{\beta}_1 g, \quad e_2 = q - \hat{\alpha}_2 - \hat{\beta}_2 g.$$

The *partial correlation* between $\mathcal{D}_1$ and $\mathcal{D}_2$ given $\mathcal{G}$ is then simply the ordinary correlation of these residuals:

$$\rho_{\text{partial}}(\mathcal{D}_1, \mathcal{D}_2 \mid \mathcal{G}) = \text{corr}(e_1, e_2).$$

Equivalently, it can be expressed directly in terms of the zero-order correlations $\rho_{12} = \rho(\mathcal{D}_1, \mathcal{D}_2)$, $\rho_{1g} = \rho(\mathcal{D}_1, \mathcal{G})$, and $\rho_{2g} = \rho(\mathcal{D}_2, \mathcal{G})$ without explicitly computing residuals:

$$\rho_{\text{partial}}(\mathcal{D}_1, \mathcal{D}_2 \mid \mathcal{G}) = \frac{\rho_{12} - \rho_{1g} \rho_{2g}}{\sqrt{1 - \rho_{1g}^2} \sqrt{1 - \rho_{2g}^2}}. \quad (14)$$

Empirical comparison. To illustrate the effect of partialing out general ability, we compute both the direct performance correlation $\rho(\mathcal{D}_i, \mathcal{D}_j)$ and the partial correlation $\rho(\mathcal{D}_i, \mathcal{D}_j \mid \text{MMLU})$ for all pairs of the eight benchmarks used in Section 5.3, and display the two matrices side by side in Figure 8. In the direct correlation matrix, most entries are uniformly high, reflecting the dominant influence of general model ability: stronger models tend to outperform weaker ones on nearly all benchmarks, inflating pairwise correlations regardless of content.

After controlling for MMLU, the partial correlation matrix reveals a markedly different structure. Cross-domain pairs (e.g., code vs. math benchmarks) exhibit substantially reduced correlations, while within-domain pairs (e.g., GSM8K–MATH, EvalPlus–MBPP) retain high values. This contrast confirms that partial correlation effectively isolates benchmark-specific capability overlap from the shared general-ability component, providing a more faithful basis for comparing with SAE feature overlap in Section 5.3.

D Ablation on Other SAE Models

A key concern of SAE-based analysis is that the results may be highly dependent on the choice of SAE. Different base models and their corresponding trained SAEs (as well as different feature dimensions) may learn features with different representations, potentially leading to different conclusions. To verify this, we re-examine the two key findings in the main text using another open-source model family, Gemma-3.

SAE Configuration. We use Gemma3-12B-it as the base model, follow the method described in Section A.3 for the selection of top- k and SAE insertion layer, and replicate the experiments in the main text.

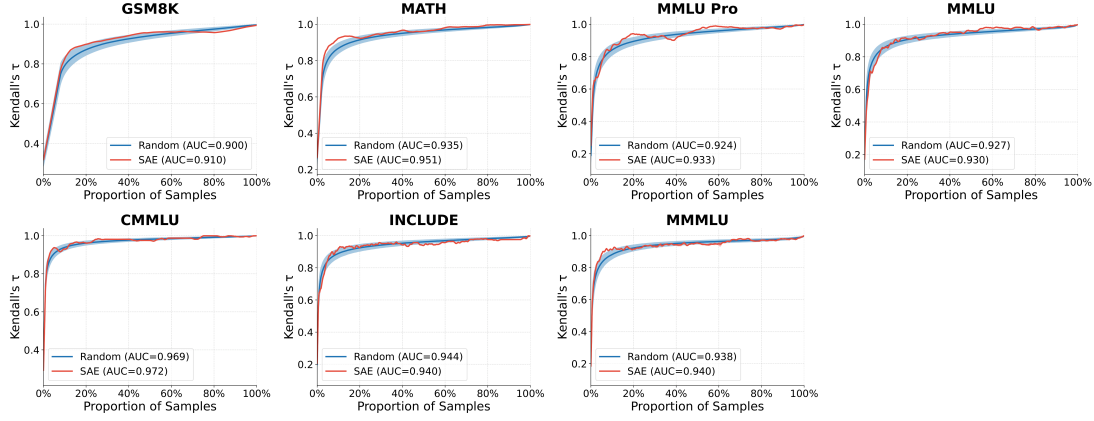


Figure 7: Random vs. SAE-Guided sampling across 7 benchmarks. Blue line: Kendall’s τ under random sampling ($\pm$ std in shaded region). Red line: Kendall’s τ under SAE-guided sampling ($\pm$ std in shaded region). SAE-guided subset achieves higher mean τ and lower variance, with the advantage most pronounced at small subset sizes.

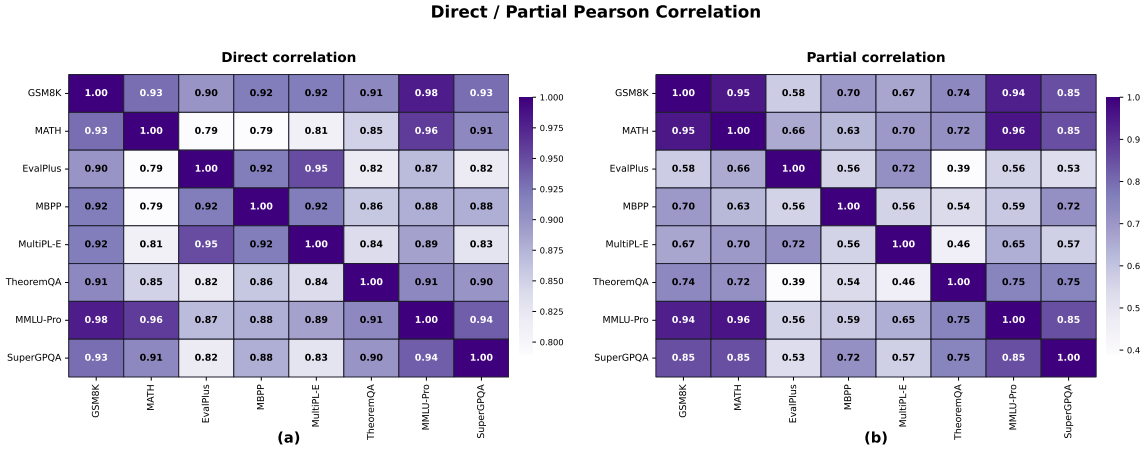


Figure 8: Performance correlation matrices for eight benchmarks. **(a)** Direct Pearson correlation $\rho(\mathcal{D}_i, \mathcal{D}_j)$: entries are uniformly high due to the dominant effect of general model ability. **(b)** Partial correlation $\rho(\mathcal{D}_i, \mathcal{D}_j \mid \text{MMLU})$: after removing the shared dependence on general ability, cross-domain correlations decrease substantially while within-domain correlations are preserved, revealing benchmark-specific capability overlap.

Benchmark Redundancy Analysis (Section 4).

For each benchmark, we recompute the feature redundancy (Eq. (8)) using the Gemma-based SAE, and calculate the correlation between performance-based redundancy and feature redundancy, as shown in Figure 9. The results demonstrate that the feature redundancy defined using the Gemma model family can still serve as a good proxy for redundancy, as indicated by the high correlation between the two.

Benchmark Similarity Analysis (Section 5).

For each pair of benchmarks, we recompute $\text{overlap}_{\min}(\mathcal{D}_i, \mathcal{D}_j)$ (Eq. (13)) using Gemma-based SAE, and then calculate its correlation with performance-based benchmark similarity. The results are shown in Table 2. The results obtained with the Gemma SAE are consistent with our pre-

vious findings: the SAE feature-based overlap exhibits a high correlation with performance-based benchmark similarity, and this correlation is further strengthened after removing the influence of general capabilities.

Summary. Through ablations on SAEs from other model families, we can verify that features learned by different SAEs may exhibit different representations (e.g., different numbers of activated features and different absolute overlap values), but in the context of benchmark analysis, such representations can be considered robust, yielding consistent conclusions in both redundancy analysis and similarity analysis.

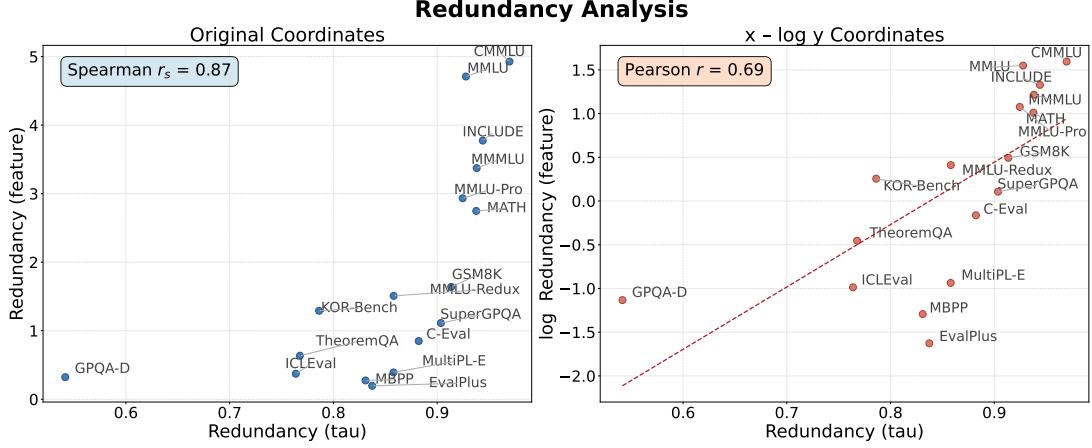


Figure 9: Spearman rank correlation between performance-based redundancy $\mathcal{R}(\mathcal{D})$ and feature redundancy $\hat{\mathcal{R}}(\mathcal{D})$ (computed using Gemma3-12B-it with its corresponding SAE) across 17 benchmarks (Spearman $\rho \approx 0.87$), suggesting that feature redundancy serves as a reasonable evaluation-free proxy for $\mathcal{R}(\mathcal{D})$.

Similarity	$\rho(\mathcal{D}_1, \mathcal{D}_2)$	$\rho(\mathcal{D}_1, \mathcal{D}_2 \mid \text{MMLU})$
Pearson	69.7	75.9
Spearman	66.6	72.2

Table 2: Spearman rank correlation between symmetric feature overlap (computed using Gemma3-12B-it with its corresponding SAE) and performance-based similarity, before and after controlling for general ability (MMLU).

E Robustness to a Heterogeneous Panel

To test whether our conclusions depend on using a single pre-training series, we augment the 26 Qwen checkpoints with 8 independently trained models spanning four families and multiple scales: Gemma3-12B-pt, Gemma3-4B-pt, Llama-3.1-8B, Qwen2.5-14B, Qwen2.5-7B, Qwen3-14B-Base, Qwen3-8B-Base, and Qwen3-4B-Base. The resulting 34-model panel covers diverse architectures, tokenizers, training corpora, and parameter scales (4B–14B). We re-run the analyses of Sections 4 and 5 on this expanded panel under an otherwise identical setup.

As shown in Table 3, the Spearman correlation between feature-based redundancy $\hat{\mathcal{R}}(\mathcal{D})$ and performance-based redundancy $\mathcal{R}(\mathcal{D})$ remains unchanged at $\rho = 0.85$, confirming that the *relative* ranking of benchmarks by redundancy is robust to panel composition. The Pearson correlation decreases from 0.82 to 0.72, which we attribute to shifts in the absolute values of $\mathcal{R}(\mathcal{D})$ induced by architecturally diverse models rather than to a change in rank order. For inter-benchmark

similarity (Section 5), the correlation between symmetric feature overlap and performance-based partial correlation remains essentially unchanged (75.5% $\rightarrow$ 75.2%). Because benchmark analysis is fundamentally about relative comparisons between benchmarks, the stability of the Spearman and inter-benchmark correlations provides strong evidence that our SAE-based metrics generalize beyond a single model family.

Model Panel	Spearman ρ	Pearson
Original (26 Qwen checkpoints)	0.85	0.82
Expanded (26 + 8 heterogeneous)	0.85	0.72

Table 3: Intra-benchmark redundancy: correlation between $\hat{\mathcal{R}}(\mathcal{D})$ and $\mathcal{R}(\mathcal{D})$ under the original homogeneous panel and the expanded heterogeneous panel. The Spearman (rank) correlation is unchanged; the Pearson correlation shifts only because the absolute redundancy scale changes with a more diverse panel.

F Size-Matched Redundancy Control

Figure 1 contrasts the rank-correlation curves of GPQA-Diamond ($K = 198$) and GSM8K ($K = 1,319$). To verify that the gap is not merely an effect of GPQA-Diamond’s smaller size, we subsample GSM8K to 198 examples (matching GPQA-Diamond) and recompute its redundancy, repeating the subsampling 10 times with different random seeds. Even at the same sample count, GSM8K’s redundancy ($\mathcal{R} = 0.794 \pm 0.010$) remains substantially higher than GPQA-Diamond’s ($\mathcal{R} = 0.542$). The gap therefore reflects intrinsic differences in benchmark redundancy rather than a sample-count

artifact.

G Use Of AI Assistants

During the preparation of this manuscript, we used large language models (LLMs) solely for assistance with grammar correction and phrasing improvements. All scientific content, experimental analyses, methodological decisions, and citations were conceived, critically reviewed, and validated entirely by the authors, who retain full responsibility for the integrity and accuracy of the work.