# HIERARCHICAL COMPRESSION OF VISION-LANGUAGE MODEL BENCHMARKS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Thorough evaluation of vision–language models (VLMs) has become prohibitively expensive, as benchmarks span an ever-broader spectrum of capabilities and new models arrive at a relentless pace. Benchmark compression methods that preserve model rankings at a fraction of the cost are well studied for language models, but for VLMs the question remains under-explored. We present **PRIMEBench** (**P**runing **R**edundant **I**tems for **M**ultimodal **E**valuation), a vision-aware hierarchical benchmark compression framework that substantially reduces evaluation cost while preserving model rankings. This hierarchical framework operates in four stages: data cleaning to remove items answerable without the image and all-correct items, category representative selection to pick one benchmark per capability category, item pruning with Vision-Aware Variance (VAW), and category-count pruning. VAW combines inter-model variance with a vision-dependence score computed from multimodal embeddings alone, while encouraging coverage of diverse items within each benchmark. On models held out from item selection, it has the highest mean fidelity at the released $5\%$ retention. The hierarchical design lets practitioners stop at any stage to match their compute budget; the released suite removes over $97\%$ of items while preserving model rankings. Beyond compression, our analyses show how VLM evaluation behaves as model panels grow and evolve, providing guidance for designing future benchmarks that are more efficient, robust to model turnover, and explicit about the limits of evaluation-side pruning.

## 1 INTRODUCTION

A single full-suite evaluation of a modern vision–language model (VLM) now consumes hundreds of GPU-hours. The benchmark landscape has expanded far beyond traditional visual question answering (Goyal et al., 2017; Hudson & Manning, 2019) to include GUI and web understanding for vision agents (Cheng et al., 2024; Koh et al., 2024), with new categories emerging rapidly alongside advances in agentic and embodied AI. Consequently, every new model release must be evaluated against an ever-growing suite of benchmarks, causing evaluation costs to compound.

In this work, we study the problem of *compressing* VLM benchmarks for efficient evaluation. Such compression is possible because VLM benchmarks exhibit redundancy at two levels. First, *across benchmarks*, the model-benchmark score matrix is markedly low-rank, suggesting that many benchmarks measure overlapping capabilities. Second, *within benchmarks*, many items contribute little discriminative signal, because they are unimportant or redundant with other items. Although several prior works have explored benchmark compression using these redundancies, they focus exclusively on within-benchmark redundancy (Polo et al., 2024; Kipnis et al., 2025; Wang et al., 2026). Moreover, only a handful of studies consider compression for VLM benchmarks, and none validates benchmark-level compression at large scale (Uebayashi et al., 2026; Joshi et al., 2026).

To this end, we propose **PRIMEBench**, a hierarchical benchmark compression framework for VLMs, supported by large-scale evaluations. Our method exploits across- and within-benchmark redundancies. Across benchmarks, benchmarks within a capability category induce highly correlated model rankings (Liang et al., 2022; Polo et al., 2024); we replace each category with a single representative benchmark and exhaustively select a compact subset that preserves the full-suite ordering. Within benchmarks, items solved by all models or failed by all carry no separation signal, while items with high inter-model variance dominate ranking quality; the all-wrong tail



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

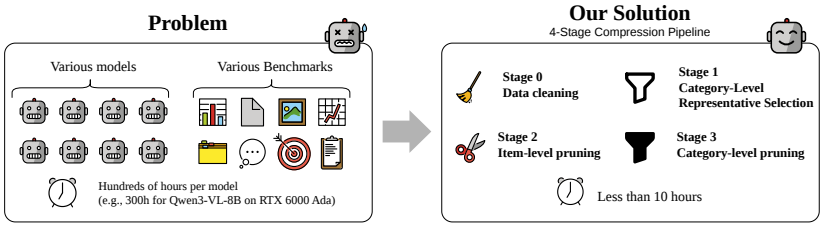



Figure 1: **Per-model VLM evaluation cost grows to hundreds of GPU-hours; PRIMEBench compresses the same suite to a compact subset that preserves model rankings.** *Left*: a full sweep over the benchmark suite consumes hundreds of GPU-hours per model and compounds with quarterly model releases. *Right*: the rankings induced by the compact **PRIMEBench** suite track the full-suite rankings closely at a small fraction of the original item budget.

has a diagnostic role for future VLM evaluation. We exploit these redundancies with one VLM-specific consideration: criteria for text-only large language models (LLMs) cannot tell apart items whose answer requires the image from items whose answer is text-driven, so we treat per-item vision-dependence as a design principle.

Our four-stage pipeline processes the score matrix end to end. *Data Cleaning* (Stage 0) removes items solvable without visual input and items answered correctly by every model. *Category Representative Selection* (Stage 1) picks one representative benchmark per category by correlation against the category mean, the average standardised score over the category's benchmarks. *Item Pruning with Vision-Aware Variance* (Stage 2) first divides the category budget, the number of items the category keeps, among its benchmarks in proportion to their sizes, which gives each benchmark its share. Within each benchmark it groups the items into *coverage cells* (clusters of items with similar vision-text embeddings, one per item to keep) and keeps the highest-scoring item of each cell. The score is the inter-model variance plus a label-free vision-dependence prior, and the prior's effective weight is annealed with the number of evaluated models so that it fades as variance estimates become reliable. *Category-Count Pruning* (Stage 3) selects a compact subset of category pools, yielding the released suite. We validate the pipeline on a panel of 30 recent VLMs and its benchmark selection externally on a large public leaderboard, and complement it with diagnostic analyses that go beyond ranking and aim to inspire the design of next-generation VLM benchmarks.

Our contributions are threefold.

1. **Open four-stage VLM compression pipeline.** We release PRIMEBench, to our knowledge the first VLM compression suite that compresses both benchmarks and items, with an end-to-end evaluation of ranking preservation, including on models held out from selection, and a benchmark-level check on an external leaderboard.
2. **Vision-Aware Variance (VAW), a variance-based item criterion with a label-free vision-dependence prior and coverage cells.** VAW scores each item by its inter-model variance plus a vision-dependence prior computed from embeddings alone, whose effective weight shrinks as the number of models $M$ grows, and keeps one item per cell within each benchmark's share.
3. **Diagnostic analyses that guide future VLM evaluation.** Beyond compression, our analyses explain where the gain of VAW comes from, test whether a compressed selection stays valid as new models arrive, and mark where compressed evaluation helps and where it stops, pointing to how evaluation suites can stay efficient as the model landscape changes.

## 2 Related Work

**VLM benchmarks and evaluation.** VLMs are now evaluated across a broad capability spectrum: general visual question answering (Goyal et al., 2017; Hudson & Manning, 2019), document and chart understanding (Mathew et al., 2021; Masry et al., 2022), and multi-discipline reasoning (Yue et al., 2024; Liu et al., 2024b; Fu et al., 2025). Evaluating a single model now routinely spans 20+ benchmarks, with public leaderboards such as the OpenVLM leaderboard (Duan et al., 2024) aggregating community results at increasing cost. Discriminative signal diminishes as models improve, since competitive models increasingly solve the same items (Kiela et al., 2021).

Table 1: **Selected benchmark compression methods, organised by modality, compression level, and validation panel size.** *Mod.* denotes the demonstration setting in the original work (T=text-only LLMs, V=vision–language). *Level*: B=benchmark, I=item. *Scale*: largest validation panel. **PRIMEBench** combines benchmark- and item-level compression, releases a compact suite, and validates item selection on 30 VLMs and benchmark-level ranking preservation against 231 VLMs.

| Method | Mod. | Level | Scale |
|---|---|---|---|
| tinyBenchmarks (Polo et al., 2024) | T | I | ∼300 LLMs |
| metabench (Kipnis et al., 2025) | T | I | ∼5000 LLMs |
| DISCO (Rubinstein et al., 2026) | T | I | — |
| EssenceBench (Wang et al., 2026) | T | I | — |
| SparseEval (Zhang et al., 2026) | T | I | — |
| SubLIME (Saranathan et al., 2025) | T | I | — |
| M3IRT (Uebayashi et al., 2026) | V | I | 24 VLMs |
| DatBench (Joshi et al., 2026) | V | I | 27 VLMs |
| **PRIMEBench (ours)** | **V** | **B+I** | **30 / 231 VLMs** |

**Benchmark compression.** A growing body of work reduces evaluation cost by selecting item subsets, but almost exclusively for text-only LLMs. Item-level criteria fall into families: item response theory (IRT) anchor selection (tinyBenchmarks (Polo et al., 2024), metabench (Kipnis et al., 2025)), inter-model-disagreement ranking (DISCO (Rubinstein et al., 2026)), genetic-algorithm search guided by sample attribution (EssenceBench (Wang et al., 2026)), sparse-anchor optimisation (SparseEval (Zhang et al., 2026)), rank-correlation prediction (SubLIME (Saranathan et al., 2025)), and embedding coverage (LMMs-Eval Lite (Zhang et al., 2025b)). Like tinyBenchmarks and DISCO, which evaluate a subset on models outside the set it was fitted on, we score every VLM selector on models unseen during selection. For multimodal evaluation, prior work has proposed multimodal IRT that decomposes item difficulty into image-only, text-only, and cross-modal factors and uses it to select subsets (M3IRT (Uebayashi et al., 2026)), and applied point-biserial item discrimination to VLM benchmarks (DatBench (Joshi et al., 2026)); none releases a compressed suite that jointly compresses the benchmark axis (which benchmarks to keep) and the item axis (which items within each benchmark to retain), and DatBench's correctness-only criterion remains vision-blind, ignoring whether an item's answer requires the visual input. **PRIMEBench** unifies the two axes for multimodal evaluation and validates benchmark-level ranking preservation on a large external panel.

## 3 Problem Statement

We formalise the task of benchmark compression as follows. Let $\mathcal{M}$ be a set of $M$ models, and let $\mathcal{B}$ be a catalogue of $B$ benchmarks. Each benchmark $b \in \mathcal{B}$ is a set of $N_b$ items.

The *item score* $x_{m,b,i}$ denotes the prediction quality of the model $m \in \mathcal{M}$ on the $i$-th item of the benchmark $b \in \mathcal{B}$. For closed-option problems, we have $x_{m,b,i} \in \{0, 1\}$, indicating whether the model has been correct on the item or not. The item score can be continuous, e.g., when using continuous metrics or LLM judge outputs. The *score* $s_{m,b} = \frac{1}{N_b} \sum_{i=1}^{N_b} x_{m,b,i}$ is the average item score of the model $m$ on benchmark $b$, where $N_b$ is the number of items in $b$. The *mean score* $\bar{s}_m = \frac{1}{B} \sum_b s_{m,b}$ is the average score of a model $m$ over all $B$ benchmarks.

The *full ranking* $\boldsymbol{\pi}$ is an ordering of $M$ models according to the mean score, i.e.,

$$\boldsymbol{\pi} = (m_{(1)}, m_{(2)}, \ldots, m_{(M)}), \qquad \text{such that} \quad \bar{s}_{m_{(1)}} \geq \bar{s}_{m_{(2)}} \geq \cdots \geq \bar{s}_{m_{(M)}} \tag{1}$$

We consider compressing this catalogue along two axes: across-benchmark, and within-benchmark. In other words, our goal is to select the *benchmark subset* $\mathcal{S} \subseteq \mathcal{B}$ and the *item subsets* $\mathcal{I}'_b \subseteq \mathcal{I}_b := \{1, \ldots, N_b\}$ for each $b \in \mathcal{S}$ in a way that the ranking evaluated on this set is close to the full ranking $\boldsymbol{\pi}$. Precisely, let $\bar{s}'_m$ be the average score of the model $m$ over the compressed suite, and $\boldsymbol{\pi}'$ be the corresponding *compressed ranking*. Then, our goal is to solve:

$$\max_{\mathcal{S}, \{\mathcal{I}'_b\}} \rho(\boldsymbol{\pi}, \boldsymbol{\pi}'), \qquad \text{subject to} \quad |\mathcal{S}| \leq k, \; |\mathcal{I}'_b| \leq r N_b, \tag{2}$$

where $k$ is the number of benchmarks (category pools in Stage 3) kept, $r$ is the fraction of each benchmark's items kept (the *retention*; Stage 2 adds a 100-item floor per category pool), and $\rho(\cdot, \cdot)$ is the Spearman rank correlation.

# 4 METHOD

Compression of a multi-benchmark VLM evaluation suite has two natural axes: *which benchmarks to keep* and *which items within them to retain*. Before either decision can be made, the item pool itself needs to be cleaned of items that no longer separate models or that admit a non-visual shortcut. Our method therefore proceeds in four stages—data cleaning (Section 4.1), category representative selection (Section 4.2), item pruning with VAW (Section 4.3), and category-count pruning (Section 4.4)—each motivated by a distinct property of VLM evaluation.

## 4.1 STAGE 0: DATA CLEANING

Stage 0 filters out benchmark items that do not carry useful evaluation signals. For example, the items that all models in $\mathcal{M}$ can answer correctly may carry no useful information for ranking. Also, the items that can be answered correctly from the text alone may fall outside the scope of VLM evaluation—their solvability reflects language priors that are measured by LLM-only benchmarks already, not vision–language capability. This stage removes both populations to obtain a clean pool that measures VLM-specific capability. We use this pool as the reference for end-to-end evaluation.

Concretely, for benchmarks with binary correctness $x_{m,b,i} \in \{0, 1\}$, Stage 0 applies two filters, the second where enough image-free scores exist.

- The *all-correct* filter removes items that all evaluated models in $\mathcal{M}$ solve correctly, i.e., $x_{m,b,i} = 1$ for all $m \in \mathcal{M}$. For items that all models predict incorrectly, we keep those items, since later generations of models may solve them.
- The *blind-solvability* filter computes $\beta_i$—the fraction of the models run without images that answer item $i$ correctly from the text prompt alone—and removes items with $\beta_i \geq \tau$ for a fixed $\tau$=2/3, a text-only filtering strategy motivated by MMStar (Chen et al., 2024a).

On benchmarks with continuous or judge scores both filters use the scores scaled to $[0, 1]$, so an item is blind-solvable when the image-free models reach a mean score of at least $2/3$ (Section B).

## 4.2 STAGE 1: CATEGORY REPRESENTATIVE SELECTION

Stage 1 reduces the number of benchmarks by removing the benchmarks that measure overlapping VLM capabilities. We organise the benchmarks into 18 capability categories—constructed by aggregating the evaluation taxonomies adopted by recent frontier VLM technical reports (Bai et al., 2025a; Wang et al., 2025; Hong et al., 2025)—and select one representative benchmark for each category. We provide additional details on the category construction in Section C.1.

More formally, define the score vector $\mathbf{s}_b = (s_{1,b}, \ldots, s_{M,b})$ for each benchmark $b$. For each category $C$, we define its *category mean* as an average of the $z$-score-normalised score vectors of the benchmarks belonging to this category, i.e., $\mathbf{s}_C = \frac{1}{|C|} \sum_{b \in C} z(\mathbf{s}_b)$, where $z(\cdot)$ denotes the $z$-score normalisation. As the category representative, we select the benchmark whose score vector is best correlated with the category mean:

$$b^*_C = \arg\max_{b \in C} \rho(\mathbf{s}_b, \mathbf{s}_C). \tag{3}$$

For categories whose representative correlates weakly with the category mean, we supplement it with items from other benchmarks in the category, using a variance-ranked prefix of those candidate items, to preserve the category-level ranking; details are in Section C.2.

## 4.3 STAGE 2: ITEM PRUNING WITH VISION-AWARE VARIANCE

Stage 2 reduces the number of items in category pools, by keeping only the items that play a critical role in ranking the vision–language capabilities of VLMs. Our decision criterion is similar to that in Stage 0. In particular, we ask two questions: (1) Does this item provide meaningful discriminative signal? (2) Does this item use visual input in a meaningful way?

Precisely, we define the following **VAW** (Vision-AWare Variance) score

$$q_i^{\mathrm{VAW}} = \widehat{\sigma}_i^2 + \alpha \cdot \delta_i^{\mathrm{VL}}, \tag{4}$$

which is a mixture of two scores: $\widehat{\sigma}_i^2$ is the *variance score*, measuring the strength of the discriminative signal provided by the item $i$; $\delta_i^{\mathrm{VL}}$ is the *vision-dependence score*, a proxy score for measuring how much the item relies on visual input; the weight $\alpha \geq 0$ sets their balance.

**Variance score.** The variance score is a $\sqrt{M}$-scaled variance of item scores across models, i.e.,

$$\widehat{\sigma}_i^2 := \frac{1}{\sqrt{M}} \sum_{m=1}^{M} (x_{m,i} - \bar{x}_i)^2, \tag{5}$$

where $\bar{x}_i = \frac{1}{M} \sum_m x_{m,i}$ is the average item score of the models on item $i$, and item scores are scaled from the range of their metric to $[0, 1]$ so that binary and continuous scores share one scale.[1] Whenever the item score is binary, the variance score admits a simpler form $\widehat{\sigma}_i^2 = \sqrt{M}\, \bar{x}_i(1 - \bar{x}_i)$.[2] LLM benchmark compression methods also use similar variance scores (Rubinstein et al., 2026).

We use $1/\sqrt{M}$ as the scaling factor instead of the standard $1/M$. This makes the effective weight of the vision-dependence score decrease with the number of evaluated models. Dividing (4) by $\sqrt{M}$ gives $\widehat{v}_i + (\alpha/\sqrt{M})\, \delta_i^{\mathrm{VL}}$, where $\widehat{v}_i = \widehat{\sigma}_i^2/\sqrt{M}$ is the plug-in variance of the item's scores across models and $\alpha_M = \alpha/\sqrt{M}$ is the *tilt*. Under i.i.d. model responses, the estimation error of $\widehat{v}_i$ concentrates at the generic $O(M^{-1/2})$ rate (Section F.2), while the tilt shrinks at the same rate. Thus, the vision-dependence term has greater relative weight for smaller model panels and fades as the variance estimate becomes more stable. For a common $M$, this rescaling does not change the item ranking; when the number of valid model scores differs across items, we rank using the corresponding divided score. We fix $\alpha = 0.187$ for all experiments, without tuning it on the held-out splits; $\alpha = 0$ gives the variance score alone.

**Vision-dependence score.** The vision-dependence score measures how adding visual inputs can steer the direction of the text-only embedding. Precisely, the score is defined as the cosine distance

$$\delta_i^{\mathrm{VL}} = 1 - \cos(E_i^{\mathrm{L}}, E_i^{\mathrm{VL}}), \tag{6}$$

where $E_i^{\mathrm{L}}$ denotes the text-only embedding of the item prompt, and $E_i^{\mathrm{VL}}$ denotes the vision-text embedding, both extracted by one VLM encoder, Qwen3-VL-Embedding-8B (Li et al., 2026).

The vision-dependence score has two advantages. First, the score is easy to compute, as we do not need to generate answers in multiple decoding steps. Second, the score is label-free: it does not use the models' item scores $x_{m,b,i}$.

**Selection with coverage cells.** At retention $r$, a Stage 1 *category pool* of $n_C$ items keeps $K_C = \max(\lfloor r n_C \rfloor, \min(n_C, 100))$ items, which we call the *category budget*. We divide this budget among benchmarks in proportion to their sizes, so benchmark $b$ receives $K_b$ items as its *benchmark share*. Within each benchmark, we group the items into $K_b$ *coverage cells* by $k$-means on the vision-text embeddings $E_i^{\mathrm{VL}}$ and keep the highest-scoring item from each cell. This fills the benchmark share while discouraging near-duplicate items from consuming the budget. Ties are broken in a seeded random order, and the embeddings are reduced to 64 dimensions by a fixed random projection before $k$-means. As $M$ grows, VAW recovers the variance-only choice within each cell when the competing items are sufficiently separated in variance (Section F.2).

### 4.4 Stage 3: Category-count pruning

Stage 3 further compresses the suite by pruning out redundant categories. The key motivation comes from the observation that the model×category score matrix is markedly low-rank. On the 18-category score matrix, the cumulative spectral energy reaches 95% within the top-5 singular values, where energy is the sum of squared singular values (Figure 2).

We therefore select a subset of $k$ categories from $\tilde{\mathcal{B}}$—the set of category pools constructed by Stage 1. Let $\tilde{\boldsymbol{\pi}}$ denote the model ranking induced by the mean score over all Stage 2-pruned category pools in $\tilde{\mathcal{B}}$, and let $\tilde{\mathbf{s}}_C$ denote the per-model score vector on category pool $C$ *after Stage 2 item pruning at*

[1] We drop the benchmark index $b$ of $x_{m,b,i}$ to lighten notation.

[2] With binary $x_{m,i} \in \{0, 1\}$, $\sum_m (x_{m,i} - \bar{x}_i)^2 = M\, \bar{x}_i(1 - \bar{x}_i)$, so dividing by $\sqrt{M}$ gives $\sqrt{M}\, \bar{x}_i(1 - \bar{x}_i)$; equivalently $\widehat{\sigma}_i^2 = \sqrt{M}\, \widehat{v}_i$ where $\widehat{v}_i = \bar{x}_i(1 - \bar{x}_i)$ is the standard plug-in variance.

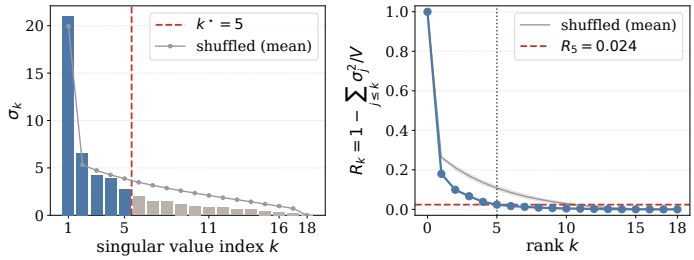



Figure 2: **The VLM score matrix is markedly low-rank.** Singular-value spectrum (*left*) and cumulative residual energy (*right*) of the per-model $z$-normalised category score matrix. The top-5 singular values capture 97.6% of the spectral energy, matching the released number of categories $k^\star$=5.

$r$=5% (*i.e.*, Stage 3 is fitted on the Stage 2 output, making the four-stage pipeline strictly sequential). Because the search space $\binom{|\tilde{\mathcal{B}}|}{k}$ is small for the numbers of categories of interest, we exhaustively enumerate every $k$-subset and pick the one that maximises Spearman correlation with $\tilde{\boldsymbol{\pi}}$:

$$\mathcal{S}_k^\star = \arg \max_{\mathcal{S} \subseteq \tilde{\mathcal{B}},\, |\mathcal{S}|=k} \rho\left(\tilde{\boldsymbol{\pi}},\ \mathrm{rank}\left(\frac{1}{|\mathcal{S}|} \sum_{C \in \mathcal{S}} \tilde{\mathbf{s}}_C\right)\right). \tag{7}$$

The $k$-sweep is in Section E.

Concretely, the released suite reports only the $k$ selected category pools, each pruned by Stage 2, yielding $\sum_{C \in \mathcal{S}_k^\star} K_C$ items in total.

## 5 Experimental Setup

**Models.** We evaluate the proposed compression pipeline on 30 vision–language and omni models from multiple families, including Qwen, InternVL, Gemma, LLaVA-OneVision, MiniCPM-V, and Kimi-VL; per-model details are in Section A.1. Vision-text embeddings from Qwen3-VL-Embedding-8B (Li et al., 2026), which is not among the evaluated models, are used for Stage 2; Section D.4 compares alternative embedding encoders.

**Benchmarks.** The catalogue contains 43 benchmarks organised into 18 capability categories spanning visual question answering, reasoning, document understanding, and GUI/web understanding; scoring types and item counts are given in Section A.2.

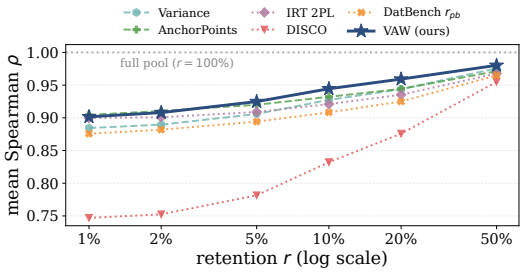
**External validation.** We additionally evaluate on the OpenVLM leaderboard (Duan et al., 2024), which contains up to 265 VLMs; details are in Section H.

**Baselines.** Method comparisons focus on Stage 2, where the item selectors differ. We compare VAW with DISCO (Rubinstein et al., 2026), DatBench $r_{pb}$ (Joshi et al., 2026), IRT 2PL (Kipnis et al., 2025), AnchorPoints (Vivek et al., 2024), and Variance at retentions $r \in \{1, 2, 5, 10, 20, 50\}$%. Variance selects the category-wide top-$K$ items ($K$=$K_C$) by Equation (5), without benchmark shares or coverage cells. DISCO selects items from prediction disagreement on the logged model answers; for a consistent comparison, its selected items are scored by their mean rather than by DISCO's learned performance predictor.

**Metrics and evaluation.** Our primary metric is the Spearman rank correlation (Spearman, 1904) between compressed- and full-pool model rankings. Each split partitions the 30 models into build and held-out sets, with Stage 2 selection performed only on the build models. For the Stage 2 comparison, all selectors operate on common category pools, built once on the full panel, and fidelity is measured on held-out models within each category. For end-to-end evaluation, Stages 1–3 are re-fitted on the build models and evaluated against the ranking from all Stage 0-cleaned benchmarks. Category pools retain at least 100 items, with smaller pools kept in full; the same floor applies to every selector. We report results over 20 balanced random 15/15 splits and a chronological split using the 15 oldest models for selection. Stage 0 cleaning and the released suite both use the full 30-model panel. Additional protocol details are provided in Section D.3.

Table 2: **Stage 0–3 pipeline at a glance: the released suite preserves model rankings with a fraction of the original items.** Change is relative to each row's input, except for the released suite, which is relative to all 43 benchmarks. $\rho$ is the median over categories for Stage 1, the mean over categories on unseen models for Stage 2, the correlation of the 5 selected categories against all 18 pruned categories for Stage 3, and the correlation with the ranking over all Stage 0-cleaned benchmarks for the released suite.

| Stage | Unit | Input | Output | Change | Ranking fidelity $\rho$ |
|---|---|---|---|---|---|
| Stage 0 (cleaning) | items | 386,324 | 270,408 | −30.0% | — |
| Stage 1 (category rep) | benchmarks | 43 | 18 | −58.1% | 0.951 (median) |
| Stage 2 (item pruning, VAW) | category-pool items | 222,376 | 11,498 | −94.8% | 0.925 (mean) |
| Stage 3 ($k$=5) | categories | 18 | 5 | −72.2% | **0.988** |
| **Released suite** | items | 386,324 | 9,377 | −97.57% | **0.959** |




(a) Spearman $\rho$ vs. retention $r$.

| Selector | 1% | 2% | 5% | 10% | 20% | 50% |
|---|---|---|---|---|---|---|
| Variance | 0.884 | 0.890 | 0.906 | 0.927 | 0.944 | 0.976 |
| AnchorPoints (Vivek et al., 2024) | **0.904** | **0.911** | 0.920 | 0.932 | 0.944 | 0.971 |
| IRT 2PL (Kipnis et al., 2025) | 0.900 | 0.901 | 0.909 | 0.921 | 0.935 | 0.968 |
| DISCO (Rubinstein et al., 2026) | 0.747 | 0.752 | 0.781 | 0.832 | 0.876 | 0.955 |
| DatBench $r_{pb}$ (Joshi et al., 2026) | 0.876 | 0.882 | 0.894 | 0.908 | 0.925 | 0.965 |
| **VAW (ours)** | 0.902 | 0.908 | **0.925** | **0.944** | **0.959** | **0.980** |

(b) Values at fixed $r$.
(Bold = best per column; underline = second.)

Figure 3: **VAW leads at the released 5% retention on models held out from Stage 2 selection.** Stage 2 item pruning on the 18 PRIMEBench category pools, with selection on half of the model panel and mean Spearman $\rho$ evaluated on the other half over 20 random splits.

## 6 Results

**End-to-end compression.** The full Stage 0→3 pipeline reduces the 386,324 items of the 43 benchmarks to 9,377 at $k$=5, and the released suite achieves $\rho$=0.959 on the full 30-VLM panel against the unpruned Stage 0-cleaned reference (Table 2).

**Stage 0 (data cleaning).** The *blind-solvability* filter ($\tau$=2/3) removes items solvable without the image, and the complementary *all-correct* filter removes items that every model solves. Stage 0 keeps all-wrong items to avoid prematurely discarding items that later models may solve; Stage 2 may subsequently prune them, since their inter-model variance is zero. Together the two filters retain 270,408 of the 386,324 items of the 43 benchmarks; per-filter breakdown is in Section B.

**Stage 1 (category representative selection).** For each of the 18 capability categories we pick the benchmark whose per-model score vector best correlates with the category mean (Equation (3)); after same-category item-level supplementation, the median per-category $\rho$ is 0.951. The per-category list and supplementation cost are in Section C.

**Stage 2 (item pruning with Vision-Aware Variance).** Within each category pool, Stage 2 keeps a small fraction $r$ of items. On the 18 PRIMEBench category pools evaluated on held-out models (Figure 3), VAW has the highest fidelity at $r = 5$–$50\%$, while AnchorPoints leads at $r \leq 2\%$. Ablations show that benchmark shares and coverage cells account for most of VAW's gain over plain Variance, and embedding-based cells outperform random partitions (Section D.4). Comparisons with selection and scoring on the same OpenVLM models are in Section H.2.

**Stage 3 (category-count pruning).** Stage 3 selects $k$ of the Stage 2-pruned category pools by exhaustively evaluating all $\binom{18}{k}$ subsets. Its selection objective compares the ranking induced by the selected categories with that induced by all 18 pruned categories. At the released setting $k = 5$, Stage 3 selects {Academic Knowledge, Chart Understanding, Hallucination Detection, Visual Reasoning, Web/UI Understanding} and reaches $\rho = 0.988$ against this 18-category pruned reference.

This differs from the end-to-end fidelity of the released suite ($\rho = 0.959$ above), which is measured against the ranking from all Stage 0-cleaned benchmarks. We further evaluate generalisation under the unseen-model protocol: Stages 1–3 are fitted on 15 build models and evaluated on the other 15. Averaged over 20 splits, VAW at $r$=5% followed by Stage 3 reaches $\rho = 0.892$ against the same Stage 0 reference, within 0.01 of Stage 3 without item pruning (0.901; Section D.3).

**External validation.** On 231 of 265 OpenVLM models scored on at least half of the 14 benchmarks overlapping with **PRIMEBench**, Stage 3 exhaustive selection at $k = 5$ achieves $\rho = 0.988$ against the 14-benchmark ranking. Era-stratified spectral analysis shows that the top five components hold 86–89% of the energy in every release-date cohort (Table 19); details are in Section H.

## 7 Discussion

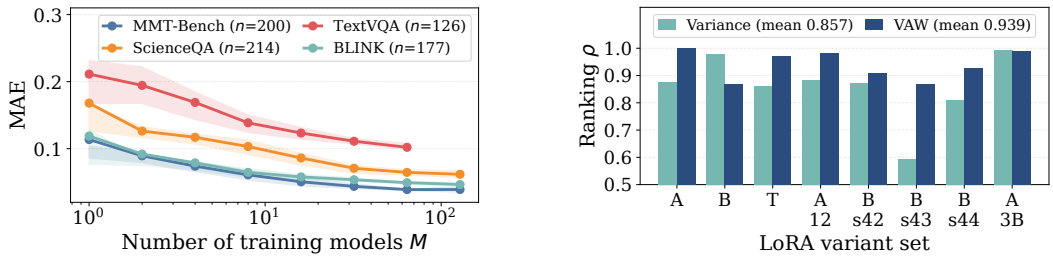
The results show that **PRIMEBench** preserves model rankings at a fraction of the original evaluation cost. We ask why VAW works, whether the selection ages, what it misses, and where else it applies.

**(1) Where the gain of VAW comes from.** VAW is vision-aware in two places: its coverage cells are drawn on vision-text embeddings, and its score adds a vision-dependence tilt. At our panel size, most of the gain comes from the cells. In the held-out ablation (Section D.4), VAW with benchmark shares alone reaches 0.908 at $r$=5%, essentially matching Variance (0.906); adding vision-text coverage cells raises it to 0.925 and outperforms random partitions of the same size. The tilt follows a theoretical scaling with panel size $M$: the plug-in variance concentrates at rate $M^{-1/2}$, so VAW anneals the tilt at the same rate, $\alpha/\sqrt{M}$, and its selection approaches the variance selector as $M$ grows. The OpenVLM scan shows that the best in-cell tilt tends to decrease with $M$ and is of the same order as the deployed $\alpha/\sqrt{M}$ on large panels. At $r$=2%, the cross-seed-selected tilt improves $\rho$ over variance-only selection in the same cells at every tested size from $M$=4 to 192; from $M$=14 onward, the gain is 0.005–0.017 and exceeds two standard errors at every tested size. These results support smaller tilts on larger panels (Sections F.2 and G).

**(2) Shelf life: does a frozen benchmark selection age as newer models arrive?** We order the OpenVLM models with a known release date by release date, select the Stage 3 subset on the oldest 132 models, freeze it, and evaluate it as the panel grows to 166, 200, and 203 models. Its ranking fidelity stays at $\rho \approx 0.99$, within 0.0004 of the best subset refitted on each enlarged panel. Because the OpenVLM score matrix is low-rank, even random five-benchmark subsets preserve rankings reasonably well, so we also rank the frozen subset among all five-benchmark subsets; it stays at or above the 99.9th percentile throughout the dated range. The benchmark-level selection therefore remains stable over the dated range studied (Section I.1).

**(3) Capability Frontier: items that every current family finds hard.** Stage 2 favours items that separate today's models, so items that most current models still fail tend to be pruned, even though such hard cases may become informative as capabilities improve. We therefore release a second set, the *Capability Frontier*: among items that every current model family finds hard (each family's mean score at most half the maximum, panel mean at least 0.10), we keep those with the largest difficulty-weighted variance (the plug-in variance times the mean shortfall $1 - \bar{x}_i$); as a heuristic budget, the set size is the in-band pool size times the panel's largest per-generation accuracy gain. The resulting 1,653 items still separate the oldest and newest Qwen generations (Section I.2). Whether they will separate future models is untested, so we treat the set as a diagnostic of the current panel rather than a forecast.

**(4) Beyond ranking: predicting a left-out benchmark from the others.** Ranking fidelity is our primary objective, but benchmark scores may also support performance estimation. We predict a model's score on a left-out OpenVLM benchmark from its scores on the others, and vary the number $M$ of models used to fit the predictor. The prediction error falls as $M$ grows (Figure 4a), paralleling LLM studies that estimate a benchmark score from a few of its items (Polo et al., 2024; Rubinstein et al., 2026); our setting instead predicts across benchmarks. Because VLM leaderboards still have sparse cross-benchmark coverage, these estimates may improve as coverage grows. The predictors are full benchmark scores, not the released suite (Section I.3).




(a) Left-out benchmark prediction.

(b) Ranking LoRA variants of one model.

Figure 4: **Benchmark scores predict a left-out benchmark better as models accumulate, and VAW ranks close variants of one model better than Variance.** (a) Mean absolute error (MAE) with the $p25$–$p75$ band over 30 splits; $n$: models scored. (b) Variant-ranking $\rho$ at $r{=}5\%$ on the eight stable LoRA variant sets (A, B: learning-rate grids; T: one run; 12: twelve rates; s42–s44: seeds; 3B: Qwen2.5-VL-3B-Instruct).

**(5) Another use: ranking a few checkpoints of one model.** Model developers often use a benchmark to choose among checkpoints or hyper-parameters of one model rather than to rank model families, a harder test because such variants differ little. We fine-tune Qwen2-VL-2B on ScienceQA with LoRA (low-rank adaptation; rank 16, two epochs) and form three kinds of variant sets: *learning-rate grids*, the final adapters of runs with 8 or 12 learning rates (a wide grid A and a narrower grid B, whose 12-rate version is repeated with three training seeds); a *trajectory* of 8 checkpoints spread over one run; and the 12-rate grid A repeated on Qwen2.5-VL-3B-Instruct. No variant takes part in item selection, and we compare their ranking on the selected items with that on the full test set. On the eight variant sets whose full-test ranking is stable (bootstrap $\rho \geq 0.90$), VAW raises this $\rho$ over Variance by 0.082 on average at $r{=}5\%$ (Figure 4b): it is ahead in six sets, behind on the 8-rate grid B and level on the 3B grid, and its lead ranges from 0.04 to 0.28 across the three seeds of the 12-rate grid B (Section I.4). Here only 8 or 12 closely related variants are ranked, so a few uninformative items can more easily reorder them; VAW's gain over Variance here is about four times its gain on the broader model panel (0.019). The two settings differ and the variants of one run are not independent draws, so this comparison is descriptive, but it suggests that VAW may be particularly useful for ranking small sets of closely related model variants.

**(6) Do evaluation-side item scores also prune training data? A negative control.** An item score that finds informative evaluation items could in principle also select informative training data. We fine-tune a SigLIP+BERT multiple-choice model on ScienceQA training subsets chosen by six selectors (Random, Variance, VAW, DatBench $r_{pb}$, AnchorPoints and IRT 2PL) at retentions from 5% to 90%, with three seeds. No selector beats random sampling by more than the seed spread, and Variance trails it by 10.5–20.2 points of test accuracy at $r \leq 30\%$ (Section I.5). *Training selection and evaluation compression are distinct problems*: the former rewards informative training examples, the latter items that separate models. We therefore do not recommend VAW or other evaluation-side scores for training-data pruning; uniform random sampling remains the safer baseline there.

## 8 Conclusion

We presented **PRIMEBench**, a vision-aware hierarchical framework for compressing VLM evaluation across both benchmarks and items. Its four-stage pipeline reduces the items of the original 43-benchmark suite by over 97% while closely preserving model rankings, and its item selector, Vision-Aware Variance, performs best at the released 5% retention on models held out from selection. We further validate the benchmark-level compression mechanism on a substantially larger external OpenVLM panel.

Overall, our results show that hierarchical compression can substantially reduce the cost of broad VLM evaluation while retaining the ranking information needed to compare new models. Practical limitations and broader implications are discussed in Sections J and K.

## AI USE STATEMENT

We used generative AI tools to assist with implementing and debugging evaluation and analysis code, providing feedback on experimental design and analysis, checking and refining mathematical exposition and claims, drafting and copy-editing portions of the manuscript, and cross-checking bibliographic metadata. GPT-based models were also used as judges for a subset of benchmark evaluations following the corresponding evaluation protocols. All AI-assisted suggestions and outputs were reviewed and verified by the authors, who made the final methodological and scientific decisions and take full responsibility for the content of the paper and the released artefacts.

## ETHICS STATEMENT

This work compresses publicly released vision–language benchmarks; it collects no new data, involves no human subjects, and requires no new human annotation. All 43 benchmarks are used under their respective terms of use (Section A.2), and we redistribute only item index lists, not images or annotations.

## REPRODUCIBILITY STATEMENT

All stages are specified in Section 4 and the appendix, and every random choice uses a fixed seed. The supplementary material provides the code and the item index lists of the released suite.

## REFERENCES

Harsh Agrawal, Karan Desai, Yufei Wang, Xinlei Chen, Rishabh Jain, Mark Johnson, Dhruv Batra, Devi Parikh, Stefan Lee, and Peter Anderson. Nocaps: Novel object captioning at scale. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, 2019.

Xiang An, Yin Xie, Kaicheng Yang, Wenkang Zhang, Xiuwei Zhao, Zheng Cheng, Yirui Wang, Songcen Xu, Changrui Chen, Didi Zhu, et al. Llava-onevision-1.5: Fully open framework for democratized multimodal training. *arXiv preprint arXiv:2509.23661*, 2025.

Xiang An, Yin Xie, Feilong Tang, Yunyao Yan, Huajie Tan, Didi Zhu, Changrui Chen, Xiuwei Zhao, Bin Qin, Kaicheng Yang, et al. Llava-onevision-2: Towards next-generation perceptual intelligence. *arXiv preprint arXiv:2605.25979*, 2026.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-VL: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2023.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025a.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, Junyang Lin, et al. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025b.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, et al. Are we on the right way for evaluating large vision-language models? *Advances in Neural Information Processing Systems*, 2024a.

Xingyu Chen, Zihan Zhao, Lu Chen, JiaBao Ji, Danyang Zhang, Ao Luo, Yuxuan Xiong, and Kai Yu. Websrc: A dataset for web-based structural reading comprehension. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015.

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024b.

Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, et al. How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites. *Science China Information Sciences*, 2024c.

Kanzhi Cheng, Qiushi Sun, Yougang Chu, Fangzhi Xu, Li YanTao, Jianbing Zhang, and Zhiyong Wu. Seeclick: Harnessing gui grounding for advanced visual gui agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2024.

Haodong Duan, Junming Yang, Yuxuan Qiao, Xinyu Fang, Lin Chen, Yuan Liu, Xiaoyi Dong, Yuhang Zang, Pan Zhang, Jiaqi Wang, et al. Vlmevalkit: An open-source toolkit for evaluating large multi-modality models. In *Proceedings of the 32nd ACM international conference on multimedia*, 2024.

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. MME: A comprehensive evaluation benchmark for multimodal large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A Smith, Wei-Chiu Ma, and Ranjay Krishna. Blink: Multimodal large language models can see but not perceive. In *European Conference on Computer Vision*. Springer, 2024.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the V in VQA matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, et al. Hallusionbench: an advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2024.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2024.

Wenyi Hong, Wenmeng Yu, Xiaotao Gu, Guo Wang, Guobing Gan, Haomiao Tang, Jiale Cheng, Ji Qi, Junhui Ji, Lihang Pan, et al. Glm-4.5 v and glm-4.1 v-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning. *arXiv preprint arXiv:2507.01006*, 2025.

Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019.

Siddharth Joshi, Haoli Yin, Rishabh Adiga, Ricardo Monti, Aldo Carranza, Alex Fang, Alvin Deng, Amro Abbas, Brett Larsen, Cody Blakeney, et al. Datbench: Discriminative, faithful, and efficient vlm evaluations. *arXiv preprint arXiv:2601.02316*, 2026.

Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. ReferItGame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *European conference on computer vision*. Springer, 2016.

Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, et al. Dynabench: Rethinking benchmarking in nlp. In *Proceedings of the 2021 conference of the North American chapter of the Association for Computational Linguistics: human language technologies*, 2021.

Alex Kipnis, Konstantinos Voudouris, Luca Schulze Buschoff, and Eric Schulz. metabench-a sparse benchmark of reasoning and knowledge in large language models. In *International Conference on Learning Representations*, 2025.

Jing Yu Koh, Robert Lo, Lawrence Jang, Vikram Duvvur, Ming Lim, Po-Yu Huang, Graham Neubig, Shuyan Zhou, Russ Salakhutdinov, and Daniel Fried. Visualwebarena: Evaluating multimodal agents on realistic visual web tasks. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2024.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. LLaVA-onevision: Easy visual task transfer. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856.

Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. Seed-bench: Benchmarking multimodal llms with generative comprehension. *arXiv preprint arXiv:2307.16125*, 2023a.

Mingxin Li, Yanzhao Zhang, Dingkun Long, Keqin Chen, Sibo Song, Shuai Bai, Zhibo Yang, Pengjun Xie, An Yang, Dayiheng Liu, et al. Qwen3-vl-embedding and qwen3-vl-reranker: A unified framework for state-of-the-art multimodal retrieval and ranking. *arXiv preprint arXiv:2601.04720*, 2026.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, 2023b.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*, 2022.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 2023.

Junpeng Liu, Yifan Song, Bill Yuchen Lin, Wai Lam, Graham Neubig, Yuanzhi Li, and Xiang Yue. Visualwebbench: How far have multimodal LLMs evolved in web page understanding and grounding? In *First Conference on Language Modeling*, 2024a.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*. Springer, 2024b.

Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xu-Cheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. Ocrbench: on the hidden mystery of ocr in large multimodal models. *Science China Information Sciences*, 2024c.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in neural information processing systems*, 2022.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations*, 2024.

Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

Andrés Marafioti, Orr Zohar, Miquel Farré, Merve Noyan, Elie Bakouch, Pedro Cuenca, Cyril Zakka, Loubna Ben Allal, Anton Lozhkov, Nouamane Tazi, et al. Smolvlm: Redefining small and efficient multimodal models. *arXiv preprint arXiv:2504.05299*, 2025.

Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*. IEEE, 2019.

Ahmed Masry, Jia Qing Tan, Shafiq Joty, Enamul Hoque, et al. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the association for computational linguistics: ACL 2022*, 2022.

Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. Docvqa: A dataset for vqa on document images. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2021.

Minesh Mathew, Viraj Bagal, Rubèn Tito, Dimosthenis Karatzas, Ernest Valveny, and CV Jawahar. Infographicvqa. In *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2022.

Rui Meng, Ziyan Jiang, Ye Liu, Mingyi Su, Xinyi Yang, Yuepeng Fu, Can Qin, Raghuveer Thirukovalluru, Xuan Zhang, Zeyuan Chen, Ran Xu, Caiming Xiong, Yingbo Zhou, Wenhu Chen, and Semih Yavuz. VLM2Vec-V2: Advancing multimodal embedding for videos, images, and visual documents. *Transactions on Machine Learning Research*, 2026.

OpenBMB. MiniCPM-V 4.6. https://huggingface.co/openbmb/MiniCPM-V-4.6, 2026. Official model card; accessed 2026-09-25.

Felipe Maia Polo, Lucas Weber, Leshem Choshen, Yuekai Sun, Gongjun Xu, and Mikhail Yurochkin. tinybenchmarks: evaluating LLMs with fewer examples. In *Forty-first International Conference on Machine Learning*, 2024.

Yusu Qian, Hanrong Ye, Jean-Philippe Fauconnier, Peter Grasch, Yinfei Yang, and Zhe Gan. Miabench: Towards better instruction following evaluation of multimodal llms. In *International Conference on Learning Representations*, 2025.

Alexander Rubinstein, Benjamin Raible, Martin Gubri, and Seong Joon Oh. Disco: Diversifying sample condensation for efficient model evaluation. In *International Conference on Learning Representations*, 2026.

Gayathri Saranathan, Cong Xu, Mahammad Parwez Alam, Tarun Kumar, Martin Foltin, Soon Yee Wong, and Suparna Bhattacharya. SubLIME: Subset selection via rank correlation prediction for data-efficient LLM evaluation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2025.

Madhuri Shanbhogue, Zhe Li, Shanfeng Zhang, Gustavo Hernández Ábrego, Shih-Cheng Huang, Aashi Jain, Daniel Salz, Sonam Goenka, Chaitra Hegde, Ji Ma, et al. Gemini embedding 2: A native multimodal embedding model from gemini. *arXiv preprint arXiv:2605.27295*, 2026.

Oleksii Sidorov, Ronghang Hu, Marcus Rohrbach, and Amanpreet Singh. Textcaps: a dataset for image captioning with reading comprehension. In *European conference on computer vision*. Springer, 2020.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019.

Charles Spearman. The proof and measurement of association between two things. *The American Journal of Psychology*, 1904.

Gemini Robotics Team, Saminda Abeyruwan, Joshua Ainslie, Jean-Baptiste Alayrac, Montserrat Gonzalez Arenas, Travis Armstrong, Ashwin Balakrishna, Robert Baruch, Maria Bauza, Michiel Blokzijl, et al. Gemini robotics: Bringing ai into the physical world. *arXiv preprint arXiv:2503.20020*, 2025a.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025b.

Gemma Team, Sherif El Abd, Vaibhav Aggarwal, Robin Algayres, Alek Andreev, Olivier Bachem, Ian Ballantyne, Cormac Brick, Victor Cărbune, Michelle Casbon, et al. Gemma 4 technical report. *arXiv preprint arXiv:2607.02770*, 2026.

Kimi Team, Angang Du, Bohong Yin, Bowei Xing, Bowen Qu, Bowen Wang, Cheng Chen, Chenlin Zhang, Chenzhuang Du, Chu Wei, et al. Kimi-vl technical report. *arXiv preprint arXiv:2504.07491*, 2025c.

NAVER Cloud HyperCLOVA X Team. Hyperclova x 8b omni. *arXiv preprint arXiv:2601.01792*, 2026.

Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai C Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, et al. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. *Advances in Neural Information Processing Systems*, 2024.

Alexandre B. Tsybakov. Optimal aggregation of classifiers in statistical learning. *The Annals of Statistics*, 2004.

Shunki Uebayashi, Kento Masui, Kyohei Atarashi, Han Bao, Hisashi Kashima, Naoto Inoue, Mayu Otani, and Koh Takeuchi. Evaluating cross-modal reasoning ability and problem characteristics with multimodal item response theory. *arXiv preprint arXiv:2603.02663*, 2026.

Rajan Vivek, Kawin Ethayarajh, Diyi Yang, and Douwe Kiela. Anchor points: Benchmarking models with much fewer examples. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2024.

Junyang Wang, Yuhang Wang, Guohai Xu, Jing Zhang, Yukai Gu, Haitao Jia, Jiaqi Wang, Haiyang Xu, Ming Yan, Ji Zhang, et al. Amber: An llm-free multi-dimensional benchmark for mllms hallucination evaluation. *arXiv preprint arXiv:2311.07397*, 2023.

Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Houxing Ren, Aojun Zhou, Mingjie Zhan, and Hongsheng Li. Measuring multimodal mathematical reasoning with math-vision dataset. *Advances in Neural Information Processing Systems*, 2024a.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024b.

Shaobo Wang, Cong Wang, Wenjie Fu, Yue Min, Mingquan Feng, Isabel Guan, Xuming Hu, Conghui He, Cunxiang Wang, Kexin Yang, et al. Rethinking llm evaluation: Can we evaluate llms with 200× less data? In *International Conference on Learning Representations*, 2026.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

Zirui Wang, Mengzhou Xia, Luxi He, Howard Chen, Yitao Liu, Richard Zhu, Kaiqu Liang, Xindi Wu, Haotian Liu, Sadhika Malladi, et al. Charxiv: Charting gaps in realistic chart understanding in multimodal llms. *Advances in Neural Information Processing Systems*, 2024c.

xAI. RealWorldQA. https://huggingface.co/datasets/xai-org/RealworldQA, 2024.

Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, Bin Zhang, Xiong Wang, Yunfei Chu, and Junyang Lin. Qwen2.5-omni technical report. *arXiv preprint arXiv:2503.20215*, 2025a. URL https://arxiv.org/abs/2503.20215.

Jin Xu, Zhifang Guo, Hangrui Hu, Yunfei Chu, Xiong Wang, Jinzheng He, Yuxuan Wang, Xian Shi, Ting He, Xinfa Zhu, et al. Qwen3-omni technical report. *arXiv preprint arXiv:2509.17765*, 2025b.

Kaining Ying, Fanqing Meng, Jin Wang, Zhiqian Li, Han Lin, Yue Yang, Hao Zhang, Wenbo Zhang, Yuqi Lin, Shuo Liu, jiayi lei, Quanfeng Lu, Runjian Chen, Peng Xu, Renrui Zhang, Haozhe Zhang, Peng Gao, Yali Wang, Yu Qiao, Ping Luo, Kaipeng Zhang, and Wenqi Shao. MMT-bench: A comprehensive multimodal benchmark for evaluating large vision-language models towards multitask AGI. In *Forty-first International Conference on Machine Learning*, 2024.

Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *European conference on computer vision*. Springer, 2016.

Tianyu Yu, Zefan Wang, Chongyi Wang, Fuwei Huang, Wenshuo Ma, Zhihui He, Tianchi Cai, Weize Chen, Yuxiang Huang, Ranchi Zhao, et al. Minicpm-v 4.5: Cooking efficient mllms via architecture, data, and training recipe. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. MM-vet: Evaluating large multimodal models for integrated capabilities. In *Proceedings of the 41st International Conference on Machine Learning*, 2024.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024.

Xiang Yue, Tianyu Zheng, Yuansheng Ni, Yubo Wang, Kai Zhang, Shengbang Tong, Yuxuan Sun, Botao Yu, Ge Zhang, Huan Sun, et al. Mmmu-pro: A more robust multi-discipline multimodal understanding benchmark. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2025.

Ge Zhang, Xinrun Du, Bei Chen, Yiming Liang, Tongxu Luo, Tianyu Zheng, Kang Zhu, Yuyang Cheng, Chunpu Xu, Shuyue Guo, et al. Cmmmu: A chinese massive multi-discipline multimodal understanding benchmark. *arXiv preprint arXiv:2401.11944*, 2024.

Jiahui Zhang, Yurui Chen, Yueming Xu, Ze Huang, Jilin Mei, Chunhui Chen, Yanpeng Zhou, Yu-Jie Yuan, Xinyue Cai, Guowei Huang, et al. From flatland to space: Teaching vision-language models to perceive and reason in 3d. *Advances in Neural Information Processing Systems*, 2025a.

Kaichen Zhang, Bo Li, Peiyuan Zhang, Fanyi Pu, Joshua Adrian Cahyono, Kairui Hu, Shuai Liu, Yuanhan Zhang, Jingkang Yang, Chunyuan Li, et al. Lmms-eval: Reality check on the evaluation of large multimodal models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, 2025b.

Taolin Zhang, Hang Guo, Wang Lu, Tao Dai, Shu-Tao Xia, and Jindong Wang. Sparseeval: Efficient evaluation of large language models by sparse optimization. In *The Fourteenth International Conference on Learning Representations*, 2026.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

# Appendix

## CONTENTS


**A Model and Benchmark Details** **17**

A.1 Model details . . . . . 17

A.2 Benchmark details . . . . . 17

**B Stage 0: Data Cleaning** **19**

**C Stage 1: Category Representative Selection** **21**

C.1 Category representative selection . . . . . 21

C.2 Multi-source item supplementation for low-$\rho$ categories . . . . . 21

**D Stage 2: Item Pruning with VAW** **23**

D.1 Selection rule . . . . . 23

D.2 Per-category variation . . . . . 23

D.3 Unseen-model protocols . . . . . 23

D.4 Cell-space ablation . . . . . 25

**E Stage 3: Category-Count Pruning** **27**

**F Theoretical Analysis** **28**

F.1 Notation and standing assumptions . . . . . 28

F.2 Behaviour of VAW as the number of models grows . . . . . 29

F.3 Formal proofs . . . . . 30

F.4 Spectral intuition for the low-rank structure of VLM score matrices . . . . . 33

**G Finite-$M$ Scan of the Vision Tilt inside Cells** **33**

**H External Validation on OpenVLM** **34**

H.1 OpenVLM data . . . . . 34

H.2 Stage 2 on OpenVLM: selector comparison . . . . . 34

H.3 Stage 3 on OpenVLM: benchmark selection . . . . . 34

H.4 Stage 3 on OpenVLM: low rank across release eras . . . . . 35

**I Additional Analyses** **35**

I.1 Shelf life of a frozen benchmark selection . . . . . 35

I.2 Capability Frontier: a descriptive stress set . . . . . 36

I.3 Left-out benchmark prediction across panel sizes . . . . . 36

I.4 Checkpoint and hyper-parameter ranking . . . . . 37

I.5 Evaluation-compression selectors do not transfer to training-data pruning . . . . . 37

**J Limitations** **38**

**K Broader Impact** **39**

# A Model and Benchmark Details

## A.1 Model details

The model panel has 30 vision–language and omni models from ten families (Table 3). Selection experiments split the panel into build models that choose the items and held-out models that are scored (Section D.3). Every model is evaluated through lmms-eval (Zhang et al., 2025b) with greedy decoding.

Table 3: **The panel has** 30 **vision–language and omni models from ten families.**

| Model | Family |
|---|---|
| Qwen-VL-Chat (Bai et al., 2023) | Qwen |
| Qwen2-VL-7B-Instruct (Wang et al., 2024b) | Qwen |
| Qwen2.5-VL-7B-Instruct (Bai et al., 2025b) | Qwen |
| Qwen2.5-Omni-3B (Xu et al., 2025a) | Qwen |
| Qwen2.5-Omni-7B (Xu et al., 2025a) | Qwen |
| Qwen3-VL-2B-Instruct (Bai et al., 2025a) | Qwen |
| Qwen3-VL-4B-Instruct (Bai et al., 2025a) | Qwen |
| Qwen3-VL-8B-Instruct (Bai et al., 2025a) | Qwen |
| Qwen3-VL-30B-A3B-Instruct (Bai et al., 2025a) | Qwen |
| Qwen3-Omni-30B-A3B-Instruct (Xu et al., 2025b) | Qwen |
| InternVL2-8B (Chen et al., 2024c) | InternVL |
| InternVL2.5-8B (Chen et al., 2024b) | InternVL |
| InternVL3-8B (Zhu et al., 2025) | InternVL |
| InternVL3.5-8B (Wang et al., 2025) | InternVL |
| Gemma3-4B-IT (Team et al., 2025b) | Gemma3 |
| Gemma3-12B-IT (Team et al., 2025b) | Gemma3 |
| Gemma4-E2B-IT (Team et al., 2026) | Gemma4 |
| Gemma4-E4B-IT (Team et al., 2026) | Gemma4 |
| Gemma4-26B-A4B-IT (Team et al., 2026) | Gemma4 |
| Gemma4-31B-IT (Team et al., 2026) | Gemma4 |
| LLaVA-OneVision-Qwen2-7B-OV (Li et al., 2025) | LLaVA-OneVision |
| LLaVA-OneVision-1.5-4B-Instruct (An et al., 2025) | LLaVA-OneVision |
| LLaVA-OneVision-1.5-8B-Instruct (An et al., 2025) | LLaVA-OneVision |
| LLaVA-OneVision-2-8B-Instruct (An et al., 2026) | LLaVA-OneVision |
| Llama-3.2-11B-Vision-Instruct (Grattafiori et al., 2024) | Llama |
| SmolVLM2-2.2B-Instruct (Marafioti et al., 2025) | SmolVLM |
| Kimi-VL-A3B-Instruct (Team et al., 2025c) | Kimi-VL |
| MiniCPM-V-4.5 (Yu et al., 2026) | MiniCPM-V |
| MiniCPM-V-4.6 (OpenBMB, 2026) | MiniCPM-V |
| HyperCLOVAX-SEED-Omni-8B (Team, 2026) | HyperCLOVAX |

## A.2 Benchmark details

Table 4 lists all benchmarks in the catalogue, grouped by the 18 capability categories used in Stage 1.

**Asset licences.** All upstream benchmarks and model checkpoints are publicly released for research use, and we use them for evaluation and for diagnostic fine-tuning on ScienceQA (Sections I.4 and I.5).

Table 4: **The catalogue mixes binary, continuous, GPT-judged, and aggregate scoring.** Scoring = the per-benchmark metric that feeds Stage 1; Items = the benchmark's items before Stage 0 cleaning (386,324 in total); MCQ = multiple-choice question, CoT = chain-of-thought, ANLS = average normalised Levenshtein similarity, MRA = mean relative accuracy, aAcc = HallusionBench all-question accuracy, Cover = AMBER object coverage, GPT = judged by a GPT model.

| Category | Benchmark | Scoring | Items |
|---|---|---|---|
| Comprehensive | MME (Fu et al., 2025) | Perception aggregate | 2,374 |
| | MMBench (Liu et al., 2024b) | GPT | 4,329 |
| | MM-Vet (Yu et al., 2024) | GPT | 218 |
| Academic Knowledge | MMMU (Yue et al., 2024) | MCQ exact match | 900 |
| | MMMU-Pro (Yue et al., 2025) | MCQ exact match | 1,730 |
| | CMMMU (Zhang et al., 2024) | MCQ exact match | 900 |
| Visual Reasoning | VQAv2 (Goyal et al., 2017) | Exact match | 214,354 |
| | GQA (Hudson & Manning, 2019) | Exact match | 12,578 |
| | BLINK (Fu et al., 2024) | Aggregate | 1,901 |
| Math Reasoning | MathVista (Lu et al., 2024) | CoT, GPT | 1,000 |
| | MathVision (Wang et al., 2024a) | Exact match | 304 |
| | OlympiadBench (He et al., 2024) | GPT | 150 |
| Document Understanding | DocVQA (Mathew et al., 2021) | ANLS | 5,349 |
| | InfoVQA (Mathew et al., 2022) | ANLS | 2,801 |
| Text Recognition | TextVQA (Singh et al., 2019) | Exact match | 5,000 |
| | OCRBench (Liu et al., 2024c) | Accuracy | 1,000 |
| Chart Understanding | ChartQA (Masry et al., 2022) | Relaxed accuracy | 2,500 |
| | CharXiv (descriptive) (Wang et al., 2024c) | GPT | 4,000 |
| | CharXiv (reasoning) (Wang et al., 2024c) | GPT | 1,000 |
| Science Knowledge | ScienceQA (Lu et al., 2022) | MCQ exact match | 4,241 |
| | AI2D (Kembhavi et al., 2016) | MCQ exact match | 3,088 |
| Object Grounding | RefCOCO (Kazemzadeh et al., 2014) | BLEU (region caption) | 1,975 |
| | RefCOCO+ (Kazemzadeh et al., 2014) | BLEU (region caption) | 1,975 |
| | RefCOCOg (Mao et al., 2016; Yu et al., 2016) | BLEU (region caption) | 5,023 |
| Spatial Reasoning | CV-Bench (Tong et al., 2024) | Exact match | 2,638 |
| | SPARBench (Zhang et al., 2025a) | MRA | 2,842 |
| Image Understanding | SEEDBench (Li et al., 2023a) | MCQ exact match | 14,233 |
| | RealWorldQA (xAI, 2024) | Exact match | 765 |
| | LLaVA-W (Liu et al., 2023) | GPT | 60 |
| Image Captioning | COCO-Cap (Chen et al., 2015) | BLEU | 5,000 |
| | NoCaps (Agrawal et al., 2019) | BLEU | 4,500 |
| | TextCaps (Sidorov et al., 2020) | BLEU | 3,166 |
| Hallucination Detection | POPE (Li et al., 2023b) | Accuracy | 9,000 |
| | HallusionBench (Guan et al., 2024) | aAcc | 951 |
| | AMBER (Wang et al., 2023) | Cover | 1,004 |
| Commonsense | OK-VQA (Marino et al., 2019) | Exact match | 5,046 |
| Multi-Turn/Multilingual | MMT-Bench (Ying et al., 2024) | MCQ exact match | 3,127 |
| | MIA-Bench (Qian et al., 2025) | GPT | 400 |
| Web/UI Understanding | ScreenSpot (Cheng et al., 2024) | Centre accuracy | 1,272 |
| | VisualWebBench (Liu et al., 2024a) | F1 | 314 |
| | WebSRC (Chen et al., 2021) | SQuAD F1 | 52,826 |
| Object Recognition | ERQA (Team et al., 2025a) | Exact match | 400 |
| Instruction Following | LLaVA-Bench-COCO (Liu et al., 2023) | GPT | 90 |

## B STAGE 0: DATA CLEANING

Stage 0 runs on all 43 benchmarks, with every item score scaled to $[0, 1]$. Both filters run once, before any split; the all-correct filter uses the whole panel, and the blind-solvability filter uses the image-free runs described next. The *all-correct* filter removes an item when every evaluated model with a valid score gets the maximum score. The *blind-solvability* filter is inspired by MMStar (Chen et al., 2024a): the models we also ran without images (Qwen-VL-Chat, Qwen2-VL, Qwen2.5-VL, Qwen3-VL 2B–8B, InternVL2–3.5 and Gemma3) answer each item from its text alone, and the item is removed when their mean score is at least $2/3$, a threshold not tuned on any ranking result. Benchmarks with too few valid image-free scores (COCO-Cap, MathVista, OlympiadBench, CharXiv, MM-Vet, MIA-Bench, LLaVA-Bench-COCO and LLaVA-W) go through the all-correct filter only.

All-wrong items are kept. Later models may solve them; with zero variance, Stage 2 rarely keeps them. Table 5 gives the per-benchmark counts. Stage 0 removes 115,916 of the 386,324 items (30.0%), and 11,247 of them fail both filters.

Table 5: **Stage 0 removes 115,916 of the 386,324 items of the 43 benchmarks.** The all-correct filter uses every evaluated model, the blind-solvability filter the image-free runs. An item can fail both filters, so the removed count is the union. The fraction removed ranges from 0.0% (14 benchmarks) to 78.3% (scienceqa).

| Benchmark | Items | All-correct | Blind-solvable | Removed | % |
|---|---|---|---|---|---|
| vqav2_val | 214,354 | 19,983 | 59,226 | 75,349 | 35.2 |
| websrc_val | 52,826 | 7,825 | 3,552 | 10,537 | 19.9 |
| seedbench | 14,233 | 3,212 | 3,620 | 5,780 | 40.6 |
| gqa | 12,578 | 316 | 2,660 | 2,904 | 23.1 |
| pope | 9,000 | 5,304 | 4,500 | 6,831 | 75.9 |
| docvqa_val | 5,349 | 13 | 107 | 119 | 2.2 |
| ok_vqa_val2014 | 5,046 | 188 | 94 | 264 | 5.2 |
| refcocog_bbox_test | 5,023 | 0 | 0 | 0 | 0.0 |
| textvqa_val | 5,000 | 912 | 151 | 1,028 | 20.6 |
| coco2017_cap_val | 5,000 | 0 | 0 | 0 | 0.0 |
| nocaps_val | 4,500 | 0 | 0 | 0 | 0.0 |
| mmbench_en_dev | 4,329 | 1,320 | 993 | 1,983 | 45.8 |
| scienceqa | 4,241 | 1,200 | 3,244 | 3,321 | 78.3 |
| charxiv_val_descriptive | 4,000 | 117 | 0 | 117 | 2.9 |
| textcaps_val | 3,166 | 0 | 0 | 0 | 0.0 |
| mmt_val | 3,127 | 395 | 652 | 949 | 30.3 |
| ai2d | 3,088 | 322 | 1,635 | 1,672 | 54.1 |
| sparbench | 2,842 | 0 | 33 | 33 | 1.2 |
| infovqa_val | 2,801 | 0 | 269 | 269 | 9.6 |
| cv_bench | 2,638 | 225 | 883 | 1,039 | 39.4 |
| chartqa | 2,500 | 496 | 208 | 584 | 23.4 |
| mme | 2,374 | 473 | 1,012 | 1,201 | 50.6 |
| refcoco_bbox_testA | 1,975 | 0 | 0 | 0 | 0.0 |
| refcoco+_bbox_testA | 1,975 | 0 | 0 | 0 | 0.0 |
| blink | 1,901 | 44 | 587 | 614 | 32.3 |
| mmmu_pro_standard | 1,730 | 22 | 162 | 172 | 9.9 |
| screenspot_rec_test | 1,272 | 0 | 0 | 0 | 0.0 |
| amber_g | 1,004 | 0 | 0 | 0 | 0.0 |
| mathvista_testmini_cot | 1,000 | 40 | 0 | 40 | 4.0 |
| ocrbench | 1,000 | 227 | 12 | 234 | 23.4 |
| charxiv_val_reasoning | 1,000 | 0 | 0 | 0 | 0.0 |
| hallusion_bench_image | 951 | 45 | 161 | 200 | 21.0 |
| mmmu_val | 900 | 30 | 264 | 268 | 29.8 |
| cmmmu_val | 900 | 9 | 152 | 153 | 17.0 |
| realworldqa | 765 | 23 | 176 | 190 | 24.8 |
| mia_bench | 400 | 0 | 0 | 0 | 0.0 |
| erqa | 400 | 6 | 56 | 58 | 14.5 |
| visualwebbench_webqa | 314 | 0 | 6 | 6 | 1.9 |
| mathvision_testmini | 304 | 0 | 1 | 1 | 0.3 |
| mmvet | 218 | 0 | 0 | 0 | 0.0 |
| olympiadbench_OE_MM_maths_en_COMP | 150 | 0 | 0 | 0 | 0.0 |
| llava_bench_coco | 90 | 0 | 0 | 0 | 0.0 |
| llava_in_the_wild | 60 | 0 | 0 | 0 | 0.0 |
| Total | 386,324 | 42,747 | 84,416 | 115,916 | 30.0 |

# C Stage 1: Category Representative Selection

## C.1 Category representative selection

The 18 categories follow the capability groupings used in recent frontier VLM technical reports (Bai et al., 2025a; Wang et al., 2025; Hong et al., 2025). Each benchmark is assigned to the category of its primary capability (Table 4), and the mapping is fixed before any selection. For each of the 18 categories, Stage 1 picks the benchmark whose per-model score correlates best with the category mean of $z$-normalised scores (Equation (3)), computed on the cleaned pool that Stage 2 consumes. Table 6 lists the representatives and Table 7 every candidate. The representative correlations have median 0.925; they fall below 0.90 in eight categories, lowest in Hallucination Detection (0.699) and Visual Reasoning (0.758), where no benchmark tracks the category mean closely.

Table 6: **Stage 1 representatives of the 18 capability categories.** The representative maximises the Spearman correlation between its per-model score and the category mean of $z$-normalised scores (Equation (3)); the last column gives the correlation across models after pooling the same-category supplements (Section C.2). Per-candidate correlations are in Table 7.

| Category | Representative | Stage 1 $\rho$ | Supplemented $\rho$ |
|---|---|---|---|
| Academic Knowledge | `mmmu_val` | 0.941 | 0.951 |
| Chart Understanding | `charxiv_val_descriptive` | 0.858 | 0.951 |
| Commonsense | `ok_vqa_val2014` | 1.000 | 1.000 |
| Comprehensive | `mmbench_en_dev` | 0.875 | 0.942 |
| Document Understanding | `infovqa_val` | 0.992 | 0.992 |
| Hallucination Detection | `amber_g` | 0.699 | 0.947 |
| Image Captioning | `nocaps_val` | 0.980 | 0.980 |
| Image Understanding | `seedbench` | 0.910 | 0.933 |
| Instruction Following | `llava_bench_coco` | 1.000 | 1.000 |
| Math Reasoning | `mathvision_testmini` | 0.898 | 0.950 |
| Multi-Turn/Multilingual | `mmt_val` | 0.863 | 0.924 |
| Object Grounding | `refcoco+_bbox_testA` | 0.950 | 0.953 |
| Object Recognition | `erqa` | 1.000 | 1.000 |
| Science Knowledge | `scienceqa` | 0.945 | 0.950 |
| Spatial Reasoning | `cv_bench` | 0.860 | 0.951 |
| Text Recognition | `textvqa_val` | 0.958 | 0.958 |
| Visual Reasoning | `vqav2_val` | 0.758 | 0.764 |
| Web/UI Understanding | `visualwebbench_webqa` | 0.794 | 0.800 |

## C.2 Multi-source item supplementation for low-$\rho$ categories

When the representative alone has $\rho < 0.95$, Stage 1 adds items from the other benchmarks of the same category, starting from the representative's full cleaned item set. Supplements are added in descending variance of their scaled scores, with ties broken by benchmark order and item index, and after each addition we recompute the Spearman correlation between the item-pooled score of the supplemented pool and the category mean. Supplementation stops at the first $\rho \geq 0.95$, or at the maximum $\rho$ when the target is not reached. Supplement items come from the Stage 0-cleaned pool, like the representatives. This pooled correlation (the last column of Table 6) can differ from the Stage 1 correlation of the representative, because it pools items of benchmarks with different metrics and sizes.

Table 8 shows three regimes. Six categories need no supplement, six reach $\rho \geq 0.95$ with 2–1,287 added items, and six stay below the target: Comprehensive (0.942), Hallucination Detection (0.947), Image Understanding (0.933), Multi-Turn/Multilingual (0.924), Visual Reasoning (0.764) and Web/UI Understanding (0.800). The supplements become part of the category pool, so the categories with large supplements also carry large pools after pruning.

Table 7: **Per-candidate Stage 1 correlations in the multi-benchmark categories.** Spearman $\rho$ between each candidate's per-model score and the category mean of $z$-normalised scores; the representative is in bold. MM-Vet is scored on 29 models because one model's score is missing.

| Category | Benchmark | $\rho$ |
|---|---|---|
| Academic Knowledge | **mmmu_val** | 0.941 |
| | mmmu_pro_standard | 0.933 |
| | cmmmu_val | 0.908 |
| Chart Understanding | chartqa | 0.668 |
| | **charxiv_val_descriptive** | 0.858 |
| | charxiv_val_reasoning | 0.840 |
| Comprehensive | mme | 0.540 |
| | **mmbench_en_dev** | 0.875 |
| | mmvet | 0.834 |
| Document Understanding | docvqa_val | 0.975 |
| | **infovqa_val** | 0.992 |
| Hallucination Detection | pope | 0.683 |
| | hallusion_bench_image | 0.510 |
| | **amber_g** | 0.699 |
| Image Captioning | coco2017_cap_val | 0.975 |
| | **nocaps_val** | 0.980 |
| | textcaps_val | 0.917 |
| Image Understanding | **seedbench** | 0.910 |
| | realworldqa | 0.858 |
| | llava_in_the_wild | 0.752 |
| Math Reasoning | mathvista_testmini_cot | 0.807 |
| | **mathvision_testmini** | 0.898 |
| | olympiadbench_OE_MM_maths_en_COMP | 0.848 |
| Multi-Turn/Multilingual | **mmt_val** | 0.863 |
| | mia_bench | 0.762 |
| Object Grounding | refcoco_bbox_testA | 0.945 |
| | **refcoco+_bbox_testA** | 0.950 |
| | refcocog_bbox_test | 0.860 |
| Science Knowledge | **scienceqa** | 0.945 |
| | ai2d | 0.916 |
| Spatial Reasoning | **cv_bench** | 0.860 |
| | sparbench | 0.680 |
| Text Recognition | **textvqa_val** | 0.958 |
| | ocrbench | 0.867 |
| Visual Reasoning | **vqav2_val** | 0.758 |
| | gqa | 0.687 |
| | blink | 0.562 |
| Web/UI Understanding | screenspot_rec_test | 0.534 |
| | **visualwebbench_webqa** | 0.794 |
| | websrc_val | 0.781 |

Table 8: **Twelve of the 18 categories reach $\rho \geq 0.95$, six of them through supplementation; six stay below it even at the best prefix of the supplements.** Tier 1 needs no supplement, tier 2 reaches the target, tier 3 does not. Items are added from the other benchmarks of the category in descending variance of their scaled scores, and $\rho$ is the correlation across models between the item-pooled score and the category mean. The supplements enter the category pool. Object Grounding is supplemented because its representative has $\rho$=0.9497 before rounding (0.950 in Table 6).

| Category | Representative | Tier | Rep. items | Added | Available | Final $\rho$ |
|---|---|---|---|---|---|---|
| Commonsense | `ok_vqa_val2014` | 1 | 4,782 | 0 | 0 | 1.000 |
| Document Understanding | `infovqa_val` | 1 | 2,532 | 0 | 5,230 | 0.992 |
| Image Captioning | `nocaps_val` | 1 | 4,500 | 0 | 8,166 | 0.980 |
| Instruction Following | `llava_bench_coco` | 1 | 90 | 0 | 0 | 1.000 |
| Object Recognition | `erqa` | 1 | 342 | 0 | 0 | 1.000 |
| Text Recognition | `textvqa_val` | 1 | 3,972 | 0 | 766 | 0.958 |
| Object Grounding | `refcoco+_bbox_testA` | 2 | 1,975 | 2 | 6,998 | 0.953 |
| Academic Knowledge | `mmmu_val` | 2 | 632 | 5 | 2,305 | 0.951 |
| Math Reasoning | `mathvision_testmini` | 2 | 303 | 66 | 1,110 | 0.950 |
| Science Knowledge | `scienceqa` | 2 | 920 | 93 | 1,416 | 0.950 |
| Spatial Reasoning | `cv_bench` | 2 | 1,599 | 673 | 2,809 | 0.951 |
| Chart Understanding | `charxiv_val_descriptive` | 2 | 3,883 | 1,287 | 2,916 | 0.951 |
| Image Understanding | `seedbench` | 3 | 8,453 | 293 | 635 | 0.933 |
| Multi-Turn/Multilingual | `mmt_val` | 3 | 2,178 | 393 | 400 | 0.924 |
| Hallucination Detection | `amber_g` | 3 | 1,004 | 709 | 2,920 | 0.947 |
| Comprehensive | `mmbench_en_dev` | 3 | 2,346 | 961 | 1,391 | 0.942 |
| Visual Reasoning | `vqav2_val` | 3 | 139,005 | 1,656 | 10,961 | 0.764 |
| Web/UI Understanding | `visualwebbench_webqa` | 3 | 308 | 37,414 | 43,561 | 0.800 |

# D Stage 2: Item Pruning with VAW

## D.1 Selection rule

Item scores are scaled linearly to $[0, 1]$, and an invalid record counts as missing, not as zero. IRT 2PL needs binary responses, so it receives the scores binarised, VQA-style soft and judge scores at half their scale, and captioning, referring-expression and the remaining graded scores at the benchmark median; every other selector and every ranking target use the scaled scores, so DatBench's $r_{pb}$ becomes the Pearson item–rest correlation on continuous scores, which equals the point-biserial correlation on binary ones.

Within benchmark $b$, the cells are drawn from the vision-text embeddings $E_i^{\mathrm{VL}}$ of Qwen3-VL-Embedding-8B (Li et al., 2026). They use item content only, so they stay fixed across panels and model subsamples. The 64-dimensional projection only speeds up $k$-means on the 4,096-dimensional embeddings of 222,149 items.

Items without a vision-text embedding enter no cell and have $\delta_i^{\mathrm{VL}} = 0$; a share that the cells cannot fill is completed with the benchmark's highest-scoring remaining items.

## D.2 Per-category variation

On unseen models VAW keeps $\rho \geq 0.90$ in thirteen of the 18 categories at $r$=5% (Table 9). The weakest are Spatial Reasoning (0.801, 113 kept items), Academic Knowledge (0.848, 100 items) and Object Recognition (0.858, 100 items). Academic Knowledge and Object Recognition are small pools (637 and 342 items) that keep only the 100-item floor, and Spatial Reasoning keeps 113 items, so a small set of items carries each of these categories. Users who need one of these categories at fine resolution should evaluate the full benchmark or use a larger retention $r$.

## D.3 Unseen-model protocols

A split divides the panel into build models, which select the items, and held-out models, which are scored. Selection reads only the build columns. Stages 0 and 1 use the responses of the whole panel and are run once, so the pools are fixed before any split; the cells use no model responses, and every

Table 9: **VAW keeps $\rho \geq 0.90$ on unseen models in thirteen of 18 categories at $r$=5%; the weakest are Spatial Reasoning, Academic Knowledge and Object Recognition.** Per-category Spearman $\rho$ on the held-out models, mean over the 20 random splits (Section D.3). Pool = cleaned representative plus supplements; $K = \max(\lfloor rN \rfloor, \min(N, 100))$. The macro $\rho$ values are 0.925 (VAW) and 0.906 (Variance, seeded random tie-breaking). The higher of the two selectors is in bold.

| Category | Representative | Pool | $K$ | VAW | Variance |
|---|---|---|---|---|---|
| Image Captioning | `nocaps_val` | 4,500 | 225 | **0.984** | **0.984** |
| Visual Reasoning | `vqav2_val` | 140,661 | 7,033 | **0.979** | 0.952 |
| Commonsense | `ok_vqa_val2014` | 4,782 | 239 | **0.970** | 0.968 |
| Document Understanding | `infovqa_val` | 2,532 | 126 | **0.966** | 0.955 |
| Object Grounding | `refcoco+_bbox_testA` | 1,977 | 100 | 0.942 | **0.949** |
| Text Recognition | `textvqa_val` | 3,972 | 198 | **0.948** | 0.935 |
| Chart Understanding | `charxiv_val_descriptive` | 5,170 | 258 | **0.936** | 0.906 |
| Web/UI Understanding | `visualwebbench_webqa` | 37,722 | 1,886 | **0.936** | 0.852 |
| Comprehensive | `mmbench_en_dev` | 3,307 | 165 | **0.928** | 0.883 |
| Multi-Turn/Multilingual | `mmt_val` | 2,571 | 128 | **0.907** | 0.902 |
| Science Knowledge | `scienceqa` | 1,013 | 100 | **0.930** | 0.907 |
| Hallucination Detection | `amber_g` | 1,713 | 100 | **0.887** | 0.748 |
| Image Understanding | `seedbench` | 8,746 | 437 | 0.884 | **0.897** |
| Spatial Reasoning | `cv_bench` | 2,272 | 113 | 0.801 | **0.837** |
| Math Reasoning | `mathvision_testmini` | 369 | 100 | **0.943** | 0.905 |
| Academic Knowledge | `mmmu_val` | 637 | 100 | 0.848 | **0.873** |
| Instruction Following | `llava_bench_coco` | 90 | 90 | **1.000** | **1.000** |
| Object Recognition | `erqa` | 342 | 100 | **0.858** | 0.855 |

Table 10: **On unseen models, VAW has the highest mean at $r$=5% and 10%, and AnchorPoints at $r$=2%.** Selection uses half of the panel and $\rho$ is measured on the other half, averaged over the 18 categories (all with a defined correlation) and 20 random splits; the last column gives $\rho$ on the chronological split at $r$=5%. Best in bold, second best underlined.

| Selector | $r$=2% | $r$=5% | $r$=10% | Chronological |
|---|---|---|---|---|
| **VAW (ours)** | <u>0.908</u> | **0.925** | **0.944** | **0.936** |
| Variance | 0.890 | 0.906 | 0.927 | 0.889 |
| IRT 2PL | 0.901 | 0.909 | 0.921 | 0.915 |
| AnchorPoints | **0.911** | <u>0.920</u> | <u>0.932</u> | <u>0.929</u> |
| DatBench $r_{pb}$ | 0.882 | 0.894 | 0.908 | 0.854 |
| DISCO | 0.752 | 0.781 | 0.832 | 0.780 |

variance, IRT or AnchorPoints clustering fit uses the build models alone. The protocol therefore tests Stage 2 selection on unseen models, not the construction of the pools; the end-to-end protocol below refits Stages 1 to 3 on the build models. For each category we compute the Spearman correlation over the held-out models between their mean on the full pool and their mean on the selected items, and average over the categories with a finite value. A category whose correlation is undefined in a split, because every held-out model receives the same score on the selected items, is left out of that split's mean. Of the six retentions, the tables report $r \in \{2, 5, 10\}\%$.

The random-split protocol draws 20 random $15/15$ splits with a fixed seed and reports the mean (Table 10). The chronological protocol orders the models by public release date, builds on the 15 oldest and scores the 15 newest; the $10/20$ and $20/10$ splits are sensitivity checks (VAW 0.917 and 0.938). Models without a verifiable release date belong to the newest release waves and sort last, and no undated model enters a build set. Each correlation needs at least 10 held-out models with finite scores. The per-split distribution is in Table 11.

Variance uses seeded random tie-breaking, as in the main comparison. At $r$=5% VAW reaches a mean of 0.925 against 0.920 for AnchorPoints, 0.909 for IRT 2PL and 0.906 for Variance. At $r$=2% AnchorPoints has the highest mean (0.911 against 0.908 for VAW). At $r$=5% VAW is the best in 10 of the 20 splits (Table 11); with 15 held-out models the split-to-split sd (0.01–0.03) is as large as the gaps between selectors. A single chronological split is subject to this variability as well, so we report it beside the random-split mean.

Table 11: **On unseen models VAW has the highest mean at $r$=5% and 10% and AnchorPoints at $r$=2%, but single splits overlap widely.** Macro Spearman $\rho$ (mean over categories) on the held-out models. Left: 20 random 15/15 splits; sd, min and max over splits at $r$=5%, and the number of splits in which the selector is the best of the 6 listed. Right: the chronological split, built on the 15 oldest models and scored on the 15 newest, at $r$=5%. The split sd is 0.01–0.03, so the ranking rests on the means, not on single splits. Best in bold, second best underlined.

| | Random 15/15 splits | | | | | | | Chronological |
|---|---|---|---|---|---|---|---|---|
| Selector | 2% | 5% | sd | min | max | best | 10% | 5% |
| VAW | <u>0.908</u> | **0.925** | 0.018 | 0.867 | 0.949 | **10** | **0.944** | **0.936** |
| IRT 2PL | 0.901 | 0.909 | 0.021 | 0.849 | 0.930 | 4 | 0.921 | 0.915 |
| AnchorPoints | **0.911** | <u>0.920</u> | 0.012 | 0.891 | 0.942 | <u>5</u> | <u>0.932</u> | <u>0.929</u> |
| Variance | 0.890 | 0.906 | 0.016 | 0.858 | 0.933 | 1 | 0.927 | 0.889 |
| DatBench $r_{pb}$ | 0.882 | 0.894 | 0.021 | 0.853 | 0.929 | 0 | 0.908 | 0.854 |
| DISCO | 0.752 | 0.781 | 0.033 | 0.722 | 0.840 | 0 | 0.832 | 0.780 |

**End-to-end comparison.** Stage 0 is a data-cleaning step applied once on all models: an item that every model answers correctly adds the same score to every model and cannot change a ranking, and a blind-solvable item does not belong to a vision–language ranking. For each split, Stage 1 is refitted on the build models (representatives and supplements from their scores), every Stage 2 selector selects items at retention $r$ on the build models, Stage 3 searches all $\binom{18}{5}$ category subsets on the build columns of the pruned matrix, and the held-out models are scored. The end-to-end $\rho$ compares the held-out ranking by the category means over all Stage 0-cleaned benchmarks with the ranking by the pruned $k$=5 leaderboard. This reference replaces the benchmark mean $\bar{s}_m$ of Section 3 by the mean of the category means, which weights each category equally. A second score keeps all pruned categories and omits Stage 3. On the 20 random splits (Table 12), Stage 3 only, without Stage 2 pruning, reaches 0.901. At $r$=5% every selector lies between 0.870 and 0.893, with AnchorPoints highest, Variance at 0.889 and VAW at 0.892, and each selector beats Stage 3 only in 6 to 8 of the 20 splits. The split-to-split sd of the end-to-end $\rho$ is 0.05–0.09, so the selectors overlap substantially, because Stage 3 chooses 5 of 18 categories from 15 build models and the chosen set changes from split to split. At $r$=5%, VAW is no more than 0.01 below every alternative, including Stage 3 only (VAW 0.892 against 0.901 for Stage 3 only), though not ahead of all of them. At $r$=2% and 10% it is 0.04 and 0.05 below the best selector. Without Stage 3 the 18 pruned categories preserve the ranking closely (VAW 0.956, IRT 2PL 0.958).

Table 12: **End to end, the Stage 2 selectors and Stage 3 alone differ little relative to split-to-split variability.** Spearman $\rho$ over held-out models between the ranking by the category means over all Stage 0-cleaned benchmarks and the $k$=5 leaderboard after Stage 2 at retention $r$ and Stage 3 on the build models; 20 random splits. *No Stage 3* scores all pruned categories; the next column counts splits in which the selector beats Stage 3 without Stage 2 pruning (Stage 3 only). The split-to-split sd of the end-to-end $\rho$ is 0.05–0.09, as large as the gaps between selectors. Variance here breaks ties by item index. Best in bold, second best underlined.

| | End-to-end | | | $r$=5% | |
|---|---|---|---|---|---|
| Stage 2 selector | 2% | 5% | 10% | No Stage 3 | >Stage 3 only |
| Stage 3 only | <u>0.901</u> | **0.901** | 0.901 | – | – |
| VAW | 0.869 | 0.892 | 0.866 | <u>0.956</u> | 6/20 |
| IRT 2PL | **0.907** | 0.890 | <u>0.909</u> | **0.958** | 7/20 |
| AnchorPoints | 0.899 | <u>0.893</u> | 0.898 | 0.950 | 8/20 |
| DatBench $r_{pb}$ | 0.870 | 0.870 | 0.883 | 0.944 | 8/20 |
| Variance | 0.873 | 0.889 | **0.917** | 0.947 | 7/20 |
| DISCO | 0.846 | 0.871 | 0.889 | 0.949 | 6/20 |

### D.4 CELL-SPACE ABLATION

The ablation keeps the score, the benchmark shares and the number of cells, and varies the space in which the cells are drawn: the vision-text embedding (the default), the text-only embedding of the

same encoder under the same projection, or a random partition of each benchmark's items into the same number of groups. Two further arms remove the cells and keep only the benchmark shares, or replace the cells with a DPP (determinantal point process). Inside each benchmark share it runs the greedy MAP (maximum a posteriori) selection of $L = \mathrm{diag}(q)\, G\, \mathrm{diag}(q)$ with an RBF (radial basis function) kernel $G$ on the projected embeddings, whose bandwidth is the median squared distance of 2,000 random item pairs, so no weight is tuned. All arms run on the 20 random splits of Section D.3 (Table 13).

With benchmark shares alone VAW reaches 0.908 at $r$=5%, level with Variance (0.906), and vision-text cells add 0.017 (18 of 20 splits). Vision-text cells are ahead of random partitions of the same size by 0.011 on average and in 16 of 20 splits, so the gain depends on grouping similar items together. Text-only cells are within about 0.005 of vision-text cells (vision-text minus text-only mean difference $-0.005$, positive in 6 of 20 splits), so the coverage gain does not require image inputs to build the cells. The DPP variant reaches 0.919 against 0.925 for the cells; both use the embeddings, but the DPP also needs a kernel and a bandwidth, while the number of cells is fixed by the budget, so the pipeline uses the cells. At $r$=10% the three cell spaces are within about 0.007, with text-only cells highest and the random partition lowest.

Inside vision-text cells the vision-dependence score $\delta^{\mathrm{VL}}$ changes the macro $\rho$ by +0.002 at $r$=2% (9 of 20 splits), +0.003 at $r$=5% and $-0.002$ at $r$=10%. These differences, computed before rounding, are descriptive; Section G reports a scan across panel sizes. The differences between cell spaces and against the DPP are at most 0.019, against split-to-split sds of 0.010–0.018 for these arms.

Table 13: **At $r$=5% embedding cells beat random partitions with either the vision-text or the text-only embedding.** Macro Spearman $\rho$ (mean over categories) on held-out models, mean over the 20 random splits of Section D.3. $-\delta$: $\alpha$=0, without the vision-dependence score. Best in bold, second best underlined.

| Selector | 2% | 5% | 10% |
|---|---|---|---|
| VAW, vision-text cells | 0.908 | 0.925 | <u>0.944</u> |
| VAW, text-only cells | **0.914** | **0.930** | **0.946** |
| VAW, random partition | 0.895 | 0.914 | 0.939 |
| VAW$-\delta$, vision-text cells | 0.906 | 0.921 | **0.946** |
| VAW$-\delta$, text-only cells | <u>0.910</u> | <u>0.929</u> | <u>0.944</u> |
| VAW$-\delta$, random partition | 0.895 | 0.914 | 0.937 |
| DPP, vision-text | 0.897 | 0.919 | 0.942 |
| DPP, text-only | 0.900 | 0.924 | 0.943 |
| DPP$-\delta$, vision-text | 0.898 | 0.921 | 0.943 |
| VAW, shares only | 0.884 | 0.908 | 0.930 |
| Variance, shares only | 0.893 | 0.908 | 0.932 |
| Variance | 0.890 | 0.906 | 0.927 |

**Encoder of the vision-dependence score.** We computed $\delta^{\mathrm{VL}}$ and the cells with three other encoders (Qwen3-VL-Embedding-2B, VLM2Vec-V2 (Meng et al., 2026) and Gemini-Embedding-2 (Shanbhogue et al., 2026)) and ran VAW with each on the held-out models of the 20 random splits (Table 14). At $r$=5% every other encoder is 0.011–0.012 below the default, so the choice of encoder moves the result by about 0.01; at $r$=2% and 10% every other encoder is above the default.

Table 14: **Swapping the encoder lowers VAW by $0.011$–$0.012$ at $r{=}5\%$.** Macro Spearman $\rho$ (mean over categories) on held-out models, mean over the 20 random $15/15$ splits, on the four categories that every encoder covers (Comprehensive, Document Understanding, Text Recognition, Science Knowledge). Each encoder supplies both $\delta^{\mathrm{VL}}$ and the cells; $\Delta\rho$ is against the default encoder at $r{=}5\%$. Variance (item-index tie-breaking) is a reference on the same splits and categories. Best in bold, second best underlined.

| Encoder | $r{=}2\%$ | $r{=}5\%$ | $r{=}10\%$ | $\Delta\rho$ |
|---|---|---|---|---|
| Qwen3-VL-Embedding-8B (default) | 0.907 | **0.943** | 0.949 | – |
| Qwen3-VL-Embedding-2B | <u>0.911</u> | 0.931 | 0.950 | $-0.012$ |
| VLM2Vec-V2 | **0.923** | <u>0.932</u> | **0.956** | $-0.011$ |
| Gemini-Embedding-2 | 0.909 | <u>0.932</u> | <u>0.954</u> | $-0.011$ |
| Variance | 0.897 | 0.914 | 0.940 | $-0.029$ |

## E Stage 3: Category-Count Pruning

Stage 3 enumerates all $\binom{18}{k}$ category subsets (Equation (7)) on the Stage 2 output at $r{=}5\%$; for $k{=}5$ it enumerates $8{,}568$ subsets. Exact ties in $\rho$ are broken by taking the first subset in alphabetical category order. We report two correlations for each $k$ (Table 15). The *Stage 3 convention* compares the ranking by the $k$ selected pruned categories with the ranking by all 18 pruned categories, which is the quantity the search maximises. The *end-to-end* correlation compares it with the ranking by the category means over all Stage 0-cleaned benchmarks, and is the one reported for the released suite. The released suite uses $k{=}5$, chosen from the spectrum of the category score matrix (Figure 2); Stage 3 maximises ranking fidelity at that $k$ and does not minimise the number of items. The selected subsets are not nested across $k$, and the category pools differ widely in size (Visual Reasoning alone keeps $7{,}033$ items), so the item count is not monotone in $k$: the $k{=}8$ subset omits Visual Reasoning and keeps $1{,}136$ items at an end-to-end $\rho$ of $0.962$ on the full panel. A user who needs fewer items can take this subset from Table 15. The released composition is in Table 16.

Table 15: **Exhaustive Stage 3 $k$-sweep on the Stage 2-pruned ($r{=}5\%$, VAW) category matrix.** Top-1 of the $\binom{18}{k}$ subsets by Spearman $\rho$ against the 18-category ranking on the pruned matrix (Stage 3 convention), also evaluated against the category means over all Stage 0-cleaned benchmarks (end-to-end).

| | $\rho$ | | | |
|---|---|---|---|---|
| $k$ | Stage 3 | End-to-end | Items | Selected categories |
| 3 | 0.976 | 0.919 | 7,391 | Academic Knowledge, Chart Understanding, Visual Reasoning |
| 5 | 0.988 | 0.959 | 9,377 | Academic Knowledge, Chart Understanding, Hallucination Detection, Visual Reasoning, Web/UI Understanding |
| 8 | 0.996 | 0.962 | 1,136 | Academic Knowledge, Chart Understanding, Commonsense, Document Understanding, Math Reasoning, Object Recognition, Science Knowledge, Spatial Reasoning |
| 10 | 0.995 | 0.958 | 9,882 | Academic Knowledge, Comprehensive, Hallucination Detection, Math Reasoning, Object Grounding, Object Recognition, Science Knowledge, Text Recognition, Visual Reasoning, Web/UI Understanding |

Table 16: **Composition of the released suite.** The $k{=}5$ categories of Table 15 and the items each keeps after Stage 2 at $r{=}5\%$.

| Category | Items kept |
|---|---|
| Academic Knowledge | 100 |
| Chart Understanding | 258 |
| Hallucination Detection | 100 |
| Visual Reasoning | 7,033 |
| Web/UI Understanding | 1,886 |
| Total | 9,377 |

**Cross-family agreement.** The item-level agreement between model families on the released suite, the Spearman correlation between two families' mean scores over its 9,377 items for the families with at least four models, is between 0.115 and 0.463 over the 6 family pairs, so the families differ in which items they find hard.

# F Theoretical Analysis

Section F.1 fixes the notation and the standing assumptions, Section F.2 states the results for the VAW score of Equation (4) on a single item pool, in the ranking-equivalent form $\widehat{v}_i + \alpha_M \delta_i^{\mathrm{VL}}$, and its paragraph on cells carries them into the cells of Stage 2; Section F.3 proves them, and Section F.4 gives the matrix reading of the low-rank observation behind Stage 3. In words, the plug-in variance of an item is estimated with noise of order $M^{-1/2}$ (Theorem 1); a tilt held fixed as the panel grows converges to a different selection whenever it changes the population top-$K$ (Theorem 2); a decaying tilt such as $\alpha_M$ approaches the population variance selector at an explicit rate when the boundary-atom mass of A4 vanishes (Theorem 3), and recovers it exactly with high probability under the boundary-gap condition A3 (Proposition 1); and for a fixed pool and fixed cells each statement holds inside every cell (Section F.2).

## F.1 Notation and standing assumptions

$M$ is the panel size and $N$ the size of the item pool. For item $i$, $x_{1,i}, \ldots, x_{M,i}$ are the per-model correctness indicators, $\widehat{p}_i = \frac{1}{M}\sum_{m=1}^{M} x_{m,i}$ is the empirical correctness rate, $p_i = \mathbb{E}[x_{m,i}]$ is the correctness probability, $v_i = p_i(1-p_i)$ is the population Bernoulli variance, and $\widehat{v}_i = \widehat{p}_i(1-\widehat{p}_i)$ is the plug-in variance estimator; for binary scoring the variance score of Equation (5) equals $\sqrt{M}\,\widehat{v}_i$. $\delta_i^{\mathrm{VL}} \in [0, D_\delta]$ is the vision-dependence score, $\alpha$ is the weight of Equation (4), and $\alpha_M$ the corresponding tilt on $\widehat{v}_i$, related by $\alpha_M = \alpha/\sqrt{M}$ (Equation (8)); the pipeline uses $\alpha = \alpha_0\sqrt{14} \approx 0.187$ with $\alpha_0 = 0.05$, that is $\alpha_M = \alpha_0\sqrt{14/M}$. $T_K(\cdot)$ returns the indices of the $K$ largest scores (ties broken by a fixed order that does not depend on the responses), $S_V^* = T_K(v_i)$ is the population variance top-$K$ set, $S_{\mathrm{VAW}}(M; a) = T_K(\widehat{v}_i + a\,\delta_i^{\mathrm{VL}})$ is the top-$K$ set on one pool for a tilt $a \geq 0$ on $\widehat{v}_i$ (the pipeline uses $a = \alpha_M$), $K/N$ is the effective retention after the 100-item floor and $\triangle$ the symmetric difference.

**Assumptions used by the theorems.** *(A1) i.i.d. Bernoulli within item.* For each item $i$ the per-model correctness indicators $x_{1,i}, \ldots, x_{M,i}$ are i.i.d. Bernoulli($p_i$). *Scores in* $[0,1]$. Stage 2 uses item scores scaled to $[0,1]$. For scores that are i.i.d. in $[0,1]$ within an item (A1 with the Bernoulli law replaced by a law on $[0,1]$), let $v_i$ be the population variance of the item score and $\widehat{v}_i = \frac{1}{M}\sum_m (x_{m,i} - \bar{x}_i)^2$ its plug-in estimator, which equals $\widehat{p}_i(1-\widehat{p}_i)$ for binary scores. Changing one model's score changes $\widehat{v}_i$ by at most $3/M$, and $\mathbb{E}[\widehat{v}_i] = (1-\frac{1}{M})v_i$ with $v_i \leq 1/4$, so McDiarmid's inequality gives $\Pr(|\widehat{v}_i - v_i| \geq \epsilon + \frac{1}{4M}) \leq 2\exp(-2M\epsilon^2/9)$. The results below are stated under (A1); with this bound in place of Theorem 1 they hold with $\sqrt{\log(2N/\eta)/(2M)}$ replaced by $3\sqrt{\log(2N/\eta)/(2M)} + \frac{1}{4M}$, so the generic concentration and the downstream selection rates are unchanged, while the $\Theta(1/M)$ refinement at $p_i = 1/2$ is specific to binary scores. When items have different numbers $M_i$ of valid records, $M_i$ replaces $M$ item by item, the ranked score is $\widehat{v}_i + \alpha\,\delta_i^{\mathrm{VL}}/\sqrt{M_i}$, the uniform bounds hold with $M = \min_i M_i$, and recovery requires $\min_i M_i$ to grow. *(A2) Bounded vision-dependence score.* $\delta_i^{\mathrm{VL}} \in [0, D_\delta]$ for some constant $D_\delta < \infty$; for the cosine distance used here $D_\delta = 2$. *(A3) Boundary-gap condition (used by Theorem 2 and Proposition 1).* The population top-$K$ boundary gap $\Delta_V := \min_{i \in S_V^*} v_i - \max_{j \notin S_V^*} v_j$ is strictly positive. *(A4) Margin condition with boundary-atom mass (used by Theorem 3).* There exist a boundary threshold $\tau_K \in [0,1]$ separating the population top-$K$ from the rest, a margin constant $L_V < \infty$, and a (possibly $N$-dependent) boundary-atom mass $\gamma_N \geq 0$ such that

$$\frac{1}{N}\#\{i : |v_i - \tau_K| \leq h\} \;\leq\; \gamma_N + 2L_V\,h \qquad \text{for every } h > 0,$$

where $\gamma_N$ absorbs any mass at the boundary. This is the standard nonparametric margin condition (Tsybakov, 2004) adapted to a finite item pool with explicit tie accounting.

### F.2 Behaviour of VAW as the number of models grows

**Bridging Equation (4) to the theorems below.** The released VAW score in Equation (4) is $q_i^{\mathrm{VAW}} = \widehat{\sigma}_i^2 + \alpha\,\delta_i^{\mathrm{VL}}$ with $\widehat{\sigma}_i^2 = \frac{1}{\sqrt{M}}\sum_m (x_{m,i} - \bar{x}_i)^2$, which for binary scoring equals $\sqrt{M}\,\widehat{v}_i$ where $\widehat{v}_i = \widehat{p}_i(1-\widehat{p}_i)$. For a common $M$ across the compared items, multiplying by the positive constant $1/\sqrt{M}$ does not change the top-$K$ selector or the per-cell maximiser of Stage 2 and yields the ranking-equivalent score

$$\frac{q_i^{\mathrm{VAW}}}{\sqrt{M}} \;=\; \widehat{v}_i + \alpha_M\,\delta_i^{\mathrm{VL}}, \qquad \alpha_M \;=\; \frac{\alpha}{\sqrt{M}}. \tag{8}$$

A fixed weight $\alpha$ in Equation (4) therefore induces an annealed tilt $\alpha_M = O(M^{-1/2})$ on the standard plug-in variance $\widehat{v}_i$. The results below are stated for the tilted top-$K$ selector on one pool of $N$ items, and the paragraph on cells below applies them to the cells. The theorems below are stated in this normalised parametrisation: Theorem 3 governs the implementation schedule realised by Equation (4) via $\alpha_M = \alpha/\sqrt{M}$, while Theorem 2 concerns the (different) regime in which a fixed tilt $a$ is held constant against $\widehat{v}_i$ rather than against $\widehat{\sigma}_i^2$.

**Theorem 1** (Variance-estimator concentration)**.** *Let $x_{1,i},\dots,x_{M,i} \overset{\text{iid}}{\sim} \mathrm{Bernoulli}(p_i)$ with $v_i = p_i(1-p_i)$. The plug-in variance estimator $\widehat{v}_i = \widehat{p}_i(1-\widehat{p}_i)$ satisfies*

$$\mathbb{E}[\widehat{v}_i] = \big(1 - \tfrac{1}{M}\big)v_i, \qquad \mathrm{Var}(\widehat{v}_i) \le \tfrac{v_i}{M} \le \tfrac{1}{4M},$$

*and for every $\epsilon > 0$, $\Pr(|\widehat{v}_i - v_i| \ge \epsilon) \le 2\exp(-2M\epsilon^2)$. At $p_i = 1/2$ the rate sharpens to $\mathrm{sd}(\widehat{v}_i) = \Theta(1/M)$.*

**Theorem 2** (Fixed normalised tilt is inconsistent when it changes the population top-$K$)**.** *Assume A1, A2. Either of the following regimes is admissible: (i) the item count $N$ is fixed as $M \to \infty$; (ii) $N = N_M$ grows but the perturbed boundary gap $\Delta_{\mathrm{Add,N}}(a)$, the population top-$K$ gap of $v_i + a\delta_i^{\mathrm{VL}}$ (written $\Delta_{\mathrm{Add}}(a)$ in the proof), satisfies $\Delta_{\mathrm{Add,N}}(a) \gg \sqrt{\log N/M}$ so that uniform concentration drives the empirical perturbed top-$K$ set to its population counterpart. For any fixed $a > 0$ independent of $M$, the additive VAW score $q_i^{\mathrm{Add}}(M;a) = \widehat{v}_i + a\delta_i^{\mathrm{VL}}$ converges in probability to the perturbed population score $v_i + a\delta_i^{\mathrm{VL}}$. If the perturbed population top-$K$ set $S^*_{\mathrm{Add}}(a) := T_K(v_i + a\delta_i^{\mathrm{VL}})$ differs from the population variance top-$K$ set $S_V^* = T_K(v_i)$ and both have positive boundary gaps, then*

$$\Pr\big(S_{\mathrm{VAW}}(M;a) = S_V^*\big) \;\to\; 0 \quad \text{as } M \to \infty.$$

*Remark on Equation (4). The fixed-tilt impossibility above is stated for the normalised additive score $\widehat{v}_i + a\,\delta_i^{\mathrm{VL}}$. The released score Equation (4) uses $\widehat{\sigma}_i^2 = \sqrt{M}\,\widehat{v}_i$ in place of $\widehat{v}_i$, so a fixed weight $\alpha$ in Equation (4) is ranking-equivalent to the annealed schedule $\alpha_M = \alpha/\sqrt{M}$ on $\widehat{v}_i$ (cf. Equation (8)); this case is governed by Theorem 3, not by Theorem 2.*

**Proposition 1** (Fixed-pool exact recovery under the boundary-gap condition)**.** *Assume A1, A2, A3 with positive top-$K$ variance gap $\Delta_V > 0$, and a non-negative annealing schedule $\alpha_M \ge 0$. Then for every $\eta \in (0,1)$, with probability at least $1-\eta$,*

$$\max_i |\widehat{v}_i - v_i| \;\le\; \sqrt{\frac{\log(2N/\eta)}{2M}}.$$

*Consequently, whenever the panel size $M$ and schedule $\alpha_M$ satisfy*

$$D_\delta \alpha_M + \sqrt{\frac{\log(2N/\eta)}{2M}} \;<\; \Delta_V/2,$$

*the VAW selector recovers $S_V^*$ exactly with probability at least $1-\eta$: $\Pr(S_{\mathrm{VAW}}(M;\alpha_M) = S_V^*) \ge 1-\eta$.*

**Theorem 3** (Annealed tilt approaches the population variance selector — density-margin rate)**.** *Assume A1, A2, A4 with constant $L_V$ and boundary-atom mass $\gamma_N$, and a non-negative annealing schedule $\alpha_M \ge 0$ with $\alpha_M \to 0$. For any confidence parameter $\eta \in (0,1)$, the VAW selector with annealed schedule $\alpha_M \to 0$ satisfies, with probability at least $1-\eta$,*

$$\frac{|S_{\mathrm{VAW}}(M;\alpha_M) \,\triangle\, S_V^*|}{K} \;=\; O\!\left(\frac{N}{K}\Big[\gamma_N + \alpha_M + \sqrt{\tfrac{\log(N/\eta)}{M}}\Big]\right),$$

*where the $\gamma_N$ term carries any boundary-atom mass that A4 cannot continuously bound. For a fixed retention $r = K/N > 0$, $N/K = 1/r$ is a constant and the rate simplifies to $O\big(\gamma_N + \alpha_M + \sqrt{\log(N/\eta)/M}\big)$; the schedule realised by Equation (4) via the $1/\sqrt{M}$ variance scaling is $\alpha_M = \alpha/\sqrt{M}$ on $\widehat{v}_i$ (cf. Equation (8)), so the $\alpha_M$ and $\sqrt{\log N/M}$ terms scale as $O(M^{-1/2})$ up to log factors and the symmetric-difference fraction vanishes whenever the displayed right-hand side tends to zero.*

**What these theorems do not claim.** Theorem 3 shows that the annealed tilt approaches the population variance selector at large $M$ when all terms of its bound vanish; for a fixed pool under the boundary-gap condition (A3), Proposition 1 gives exact recovery. At the panel sizes used here ($M \leq 30$ with pools of 90 items or more in our panel, and up to 265 models with pools of 218 items or more on OpenVLM), the union-bound radius over all items used in Proposition 1 at confidence 0.95 exceeds $1/8$, half the largest possible gap between two variances, so the bounds describe the large-$M$ regime and give no finite-sample guarantee at these panel sizes. The analysis does not show that variance selection is the strongest selector at large $M$, which would need a minimax lower bound, nor that the tilt improves on variance at small $M$, which would need a finite-sample risk bound; the properties above bound the effect of the tilt in the limit under these conditions and do not show that it helps. Whether it helps is an empirical question, answered by the scan of Section G. The results also differ in kind from the IRT-based selection of tinyBenchmarks (Polo et al., 2024) and metabench (Kipnis et al., 2025), which fit a parametric response model and select items with it. Our statements assume no ability model and bound the estimation error of a model-free item statistic. Four limits apply. The pool is chosen with model responses: Stage 0 removes items by their responses, Stage 1 picks benchmarks by response correlations and adds supplements in descending variance of the same scores (Section C.2), so A1 is assumed for the fixed pool rather than inherited from the pipeline. A1 also treats the models as independent draws, while models of one family share architecture and data, so the effective panel size is smaller than $M$; extending the bounds to dependent families would need a dependence model and a matching concentration bound, which we do not provide. The margin constant $L_V$ of A4 is a population quantity that we do not estimate. The bounds hold within one pool, and transfer across pools requires the distribution of $(v_i, \delta_i^{\mathrm{VL}})$ to be stable.

**Inside the cells.** Stage 2 keeps one item per cell (Section 4.3). A cell is itself an item pool, so for a fixed pool and fixed cells the results above apply inside every cell with $K = 1$, and one union bound over all $N$ items covers every cell at once. The population choice of a cell is its variance maximiser, which the annealed tilt recovers as $M$ grows when the cell's best two items are separated (Proposition 1 with $K = 1$, which needs the deviation radius below half the smallest such gap over all cells). As for the whole pool, the statements treat the pool and its cells as fixed; the pool itself is chosen with model responses in Stages 0 and 1, which is the first of the limits listed above.

**Scope of the design.** The in-cell score $q_i^{\mathrm{VAW}} = \widehat{\sigma}_i^2 + \alpha\, \delta_i^{\mathrm{VL}}$ of Equation (4) is constrained in four ways, the first three by construction and the fourth under the assumptions of Section F.1. First, what it adds to the variance score *uses no response information*, since the cells and $\delta_i^{\mathrm{VL}}$ come from frozen vision-text embeddings (Qwen3-VL-Embedding-8B), not from model responses; the pool they act on is fixed by Stages 0 and 1, which do use responses. Second, it is *VLM-targeted*, since $\delta_i^{\mathrm{VL}}$ measures how far the image moves an item's representation, which a text-only criterion cannot see. Third, it is *annealed*, since the ranked score $q_i^{\mathrm{VAW}}/\sqrt{M}$ equals the plug-in variance plus $(\alpha/\sqrt{M})\, \delta_i^{\mathrm{VL}}$, so the tilt on the plug-in variance shrinks as $M^{-1/2}$, the worst-case rate at which that variance concentrates (Theorem 1). Fourth, it *recovers variance-only selection*. Setting $\alpha{=}0$ gives variance-only selection in the same cells (VAW$-\delta$), and for a fixed pool and fixed cells Proposition 1 holds within each cell with $K = 1$, so under a boundary-gap condition (A3) on the cells the VAW choice recovers the choice by the items' true variances as $M$ grows. Section G reports its measured effect at each panel size.

### F.3 Formal proofs

This subsection proves Theorems 1 to 3 and Proposition 1, with the notation and assumptions A1–A4 of Section F.1; the spectral low-rank structure (Figure 2) is an empirical observation, discussed in Section F.4.

#### F.3.1 PROOF OF THEOREM 1 (VARIANCE-ESTIMATOR CONCENTRATION)

The proof computes the mean from binomial moments and transfers Hoeffding concentration from $\widehat{p}_i$ to $\widehat{v}_i$ through the 1-Lipschitz map $x \mapsto x(1-x)$; the sharper rate at $p_i = 1/2$ follows because the quadratic expansion of that map is exact.

Let $S_M = \sum_{m=1}^{M} x_{m,i} = M\widehat{p}_i$, so $S_M \sim \mathrm{Binomial}(M, p_i)$.

**Mean.** $\mathbb{E}[\widehat{v}_i] = \mathbb{E}[\widehat{p}_i] - \mathbb{E}[\widehat{p}_i^2]$. Since $\mathbb{E}[\widehat{p}_i] = p_i$ and $\mathrm{Var}(\widehat{p}_i) = p_i(1-p_i)/M = v_i/M$, we have $\mathbb{E}[\widehat{p}_i^2] = p_i^2 + v_i/M$. Therefore

$$\mathbb{E}[\widehat{v}_i] = p_i - p_i^2 - v_i/M = v_i(1 - 1/M).$$

**Variance.** Write $\widehat{v}_i = g(\widehat{p}_i)$ for $g(x) = x(1-x)$. The function $g$ is 1-Lipschitz on $[0,1]$ (since $|g'(x)| = |1-2x| \le 1$), so for any pair of i.i.d. copies $X, Y$ of $\widehat{p}_i$ we have $(g(X) - g(Y))^2 \le (X - Y)^2$ deterministically; taking expectations and using the identity $\mathrm{Var}(g(\widehat{p}_i)) = \frac{1}{2}\mathbb{E}[(g(X) - g(Y))^2]$ yields the finite-sample bound

$$\mathrm{Var}(\widehat{v}_i) \;\le\; \mathrm{Var}(\widehat{p}_i) \;=\; v_i/M \;\le\; 1/(4M).$$

**Concentration.** Apply Hoeffding's inequality to $\widehat{p}_i$: $\Pr(|\widehat{p}_i - p_i| \ge \epsilon) \le 2\exp(-2M\epsilon^2)$. The 1-Lipschitz property of $g$ gives the deterministic bound $|\widehat{v}_i - v_i| = |g(\widehat{p}_i) - g(p_i)| \le |\widehat{p}_i - p_i|$. Chaining the two bounds:

$$\Pr\big(|\widehat{v}_i - v_i| \ge \epsilon\big) \;\le\; \Pr\big(|\widehat{p}_i - p_i| \ge \epsilon\big) \;\le\; 2\exp(-2M\epsilon^2),$$

which is exactly the bound stated in the theorem.

**Refined rate at $p_i = 1/2$.** At $p_i = 1/2$, $g'(p_i) = 0$, and because $g$ is quadratic the second-order Taylor expansion is *exact* with no remainder:

$$\widehat{v}_i - v_i \;=\; g(\widehat{p}_i) - g(p_i) \;=\; -(\widehat{p}_i - 1/2)^2.$$

For $S_M \sim \mathrm{Binomial}(M, 1/2)$, $\mathrm{Var}((\widehat{p}_i - 1/2)^2) = (M-1)/(8M^3) = \Theta(1/M^2)$, so $\mathrm{sd}(\widehat{v}_i) = \Theta(1/M)$, sharper than the generic $\Theta(M^{-1/2})$ rate. □

#### F.3.2 PROOF OF THEOREM 2 (FIXED NORMALISED TILT IS INCONSISTENT WHEN IT CHANGES THE POPULATION TOP-$K$)

The strategy is uniform convergence of the perturbed empirical score to its population counterpart $v_i + a\delta_i^{\mathrm{VL}}$: the perturbed population top-$K$ set has its own positive boundary gap, so the empirical selector locks onto that perturbed set, which differs from $S_V^*$ by assumption.

Fix $a > 0$ independent of $M$. Define $u_i^{(M)} := \widehat{v}_i + a\delta_i^{\mathrm{VL}}$ and the population perturbed score $u_i := v_i + a\delta_i^{\mathrm{VL}}$. By Theorem 1, $\widehat{v}_i \xrightarrow{p} v_i$ at rate $O(M^{-1/2})$; since $\delta_i^{\mathrm{VL}}$ is fixed (label-free, not estimated from the panel), $u_i^{(M)} \xrightarrow{p} u_i$ uniformly in $i \in \{1, \dots, N\}$ by union bound: for any $\epsilon > 0$,

$$\Pr\big(\max_i |u_i^{(M)} - u_i| \ge \epsilon\big) \le 2N\exp(-2M\epsilon^2) \to 0 \text{ as } M \to \infty.$$

For the growing-$N$ regime $N = N_M$, set $\epsilon_M = \Delta_{\mathrm{Add,N_M}}(a)/2$ so that

$$\Pr\big(\max_i |u_i^{(M)} - u_i| \ge \Delta_{\mathrm{Add,N_M}}(a)/2\big) \le 2N_M\exp\big(-M\Delta_{\mathrm{Add,N_M}}(a)^2/2\big) \to 0,$$

which holds under the gap rate $\Delta_{\mathrm{Add,N}}(a) \gg \sqrt{\log N/M}$ stated in the theorem. Let $S^*_{\mathrm{Add}}(a) := T_K(u_i)$. Suppose $S^*_{\mathrm{Add}}(a) \ne S_V^* := T_K(v_i)$, and let

$$\Delta_{\mathrm{Add}}(a) := \min_{i \in S^*_{\mathrm{Add}}(a)} u_i - \max_{j \notin S^*_{\mathrm{Add}}(a)} u_j > 0$$

denote the population top-$K$ boundary gap of the perturbed score (positive by assumption). On the event $\{\max_i |u_i^{(M)} - u_i| < \Delta_{\mathrm{Add}}(a)/2\}$, the empirical top-$K$ set $T_K(u_i^{(M)}) = S_{\mathrm{VAW}}(M; a)$ equals $S^*_{\mathrm{Add}}(a)$ exactly (the gap is wider than the maximum perturbation, so no item flips across the boundary). The probability of this event tends to 1 as $M \to \infty$, so

$$\Pr\big(S_{\mathrm{VAW}}(M; a) = S^*_{\mathrm{Add}}(a)\big) \to 1.$$

Since $S^*_{\mathrm{Add}}(a) \ne S_V^*$, $\Pr(S_{\mathrm{VAW}}(M; a) = S_V^*) \to 0$. □

### F.3.3 Proof of Theorem 3 (annealed tilt approaches the population variance selector)

We bound the uniform deviation of the annealed score by an estimation-noise term plus the vanishing tilt, show that every misclassified item lies in a near-boundary band twice as wide as that deviation, and let the margin condition A4 convert the band width into an item count.

Let $\alpha_M \to 0$ as $M \to \infty$ and fix $\eta \in (0,1)$. Write $u_i^{(M)} := \widehat{v}_i + \alpha_M \delta_i^{\mathrm{VL}}$ and decompose

$$u_i^{(M)} - v_i = (\widehat{v}_i - v_i) + \alpha_M \delta_i^{\mathrm{VL}}.$$

The first term is bounded uniformly in $i$ by Hoeffding + union bound: with probability at least $1-\eta$,

$$\max_i |\widehat{v}_i - v_i| \le \sqrt{\frac{\log(2N/\eta)}{2M}} = O\left(\sqrt{\frac{\log(N/\eta)}{M}}\right).$$

The second term is bounded deterministically: $|\alpha_M \delta_i^{\mathrm{VL}}| \le \alpha_M D_\delta = O(\alpha_M)$. Therefore $\max_i |u_i^{(M)} - v_i| = O(\alpha_M + \sqrt{\log(N/\eta)/M})$ with probability at least $1-\eta$.

Let $\tau_K$ denote the population boundary threshold separating top-$K$ from the rest under A4, and write $\xi_M = \max_\ell |u_\ell^{(M)} - v_\ell|$. We show by a pairwise gap argument that $S_{\mathrm{VAW}}(M;\alpha_M) \triangle S_V^*$ is contained in the near-boundary set $\{i : |v_i - \tau_K| \le 2\xi_M\}$. For any $i \in S_V^* \setminus S_{\mathrm{VAW}}(M;\alpha_M)$ and $j \in S_{\mathrm{VAW}}(M;\alpha_M) \setminus S_V^*$, the empirical top-$K$ rule gives $u_j^{(M)} \ge u_i^{(M)}$, hence $v_i \le u_i^{(M)} + \xi_M \le u_j^{(M)} + \xi_M \le v_j + 2\xi_M$. Since $v_i \ge \tau_K$ (top side) and $v_j \le \tau_K$ (bottom side), both lie within $2\xi_M$ of $\tau_K$, so $|v_i - \tau_K| \le 2\xi_M$ and $|v_j - \tau_K| \le 2\xi_M$. Applying A4 with $h = 2\xi_M$, whose $\gamma_N$ term absorbs boundary ties (including the boundary item itself in finite pools), the number of items that can be misclassified is at most

$$\begin{aligned} N \cdot \left[\gamma_N + 2L_V \cdot 2 \max_i |u_i^{(M)} - v_i|\right] &= N \cdot \left[\gamma_N + 4L_V \max_i |u_i^{(M)} - v_i|\right] \\ &= O\left(N\left[\gamma_N + \alpha_M + \sqrt{\tfrac{\log(N/\eta)}{M}}\right]\right). \end{aligned}$$

This bound is density-based with explicit tie accounting: we do not assume the boundary-gap condition (A3), and we do not condition on an event in which exactly zero items cross the boundary; the $\gamma_N$ term handles both the $K$-th order-statistic atom and any other boundary ties. Dividing by $K$ gives the symmetric-difference bound

$$\frac{|S_{\mathrm{VAW}}(M;\alpha_M) \triangle S_V^*|}{K} = O\left(\frac{N}{K}\left[\gamma_N + \alpha_M + \sqrt{\tfrac{\log(N/\eta)}{M}}\right]\right).$$

For the implementation schedule $\alpha_M = \alpha/\sqrt{M}$ realised by Equation (4) (cf. Equation (8)), $\alpha_M = O(M^{-1/2})$ matches the estimator-noise term up to constants and log factors, so the two $M$-dependent terms vanish at rate $O(M^{-1/2})$ up to logarithmic factors; the $\gamma_N$ term must vanish separately under the assumed asymptotic regime (e.g., $\gamma_N \to 0$ under A4 in the large-$N$ limit, or $\gamma_N \le c/N$ for a constant $c$ independent of $N$, which vanishes as $N \to \infty$, if the only boundary atom is the order-statistic itself). □

### F.3.4 Proof of Proposition 1 (fixed-pool exact recovery)

Hoeffding's inequality and a union bound give a uniform deviation radius $\zeta_M$ for the tilted score; once $\zeta_M$ is below half the population boundary gap $\Delta_V$, no item can cross the top-$K$ boundary, which forces exact recovery.

By Hoeffding's inequality applied to each $\widehat{p}_i$ followed by the 1-Lipschitz bound $|\widehat{v}_i - v_i| \le |\widehat{p}_i - p_i|$ from Theorem 1, $\Pr(|\widehat{v}_i - v_i| \ge \epsilon) \le 2\exp(-2M\epsilon^2)$ for each $i$. By a union bound over $N$ items,

$$\Pr\Big(\max_i |\widehat{v}_i - v_i| \ge \epsilon\Big) \;\le\; 2N\exp(-2M\epsilon^2).$$

Setting $2N\exp(-2M\epsilon^2) = \eta$ yields $\epsilon = \sqrt{\log(2N/\eta)/(2M)}$. Combining with the deterministic bound $|\alpha_M \delta_i^{\mathrm{VL}}| \le D_\delta \alpha_M$, the maximum perturbation between the empirical score $\widehat{v}_i + \alpha_M \delta_i^{\mathrm{VL}}$ and

the population variance $v_i$ is, with probability at least $1-\eta$,

$$\max_i |(\widehat{v}_i + \alpha_M \delta_i^{\mathrm{VL}}) - v_i| \;\le\; D_\delta \alpha_M + \sqrt{\frac{\log(2N/\eta)}{2M}}.$$

Define $\zeta_M = D_\delta \alpha_M + \sqrt{\log(2N/\eta)/(2M)}$. With probability at least $1-\eta$, $\max_i |(\widehat{v}_i + \alpha_M \delta_i^{\mathrm{VL}}) - v_i| \le \zeta_M$. If $\zeta_M < \Delta_V/2$, then for every $i \in S_V^*$ and $j \notin S_V^*$,

$$(\widehat{v}_i + \alpha_M \delta_i^{\mathrm{VL}}) - (\widehat{v}_j + \alpha_M \delta_j^{\mathrm{VL}}) \;\ge\; (v_i - v_j) - 2\zeta_M \;\ge\; \Delta_V - 2\zeta_M \;>\; 0,$$

so the empirical top-$K$ set equals $S_V^*$ exactly. □

### F.4 Spectral intuition for the low-rank structure of VLM score matrices

The low-rank structure enters the pipeline as an empirical observation. The leading singular values of the category matrix carry most of its energy (Figure 2), which motivates the small number of categories in Stage 3 (Section 4.4).

*Intuition.* Eckart–Young is a matrix-level statement: it bounds the squared Frobenius residual of the rank-$k$ SVD approximation $\mathbf{S}_k^{\mathrm{svd}}$ of the $z$-normalised model-by-category matrix $\mathbf{S}$, $\|\mathbf{S} - \mathbf{S}_k^{\mathrm{svd}}\|_F^2 = V - V_k = V R_k$ (with $V$ the total and $V_k$ the top-$k$ squared Frobenius energy, and $R_k$ the relative residual energy), not the projection residual of an arbitrary single vector. Under a unit-Frobenius normalisation $V = 1$ (the $z$-normalised matrix of Figure 2 has $V = 540$ instead), the matrix-level translation into an entrywise bound on the mean score vector follows from a Cauchy–Schwarz argument: $\|\mathbf{s}_{\mathrm{full}} - \mathbf{s}_k^{\mathrm{svd}}\|_\infty \le \sqrt{R_k / B_{\mathrm{cat}}}$ for the column-mean score vectors of $\mathbf{S}$ and $\mathbf{S}_k^{\mathrm{svd}}$ over its $B_{\mathrm{cat}}$ columns; without the $V = 1$ normalisation the bound reads $\sqrt{V R_k / B_{\mathrm{cat}}}$. Before item pruning $95\%$ of the energy needs 4 components (participation-ratio effective rank $(\sum_j \lambda_j)^2 / \sum_j \lambda_j^2$ over the singular values $\lambda_j$, 4.53), within the number of categories Stage 3 keeps; after Stage 2 keeps $5\%$ of the items it needs 7 (effective rank 7.73; top-5 $92.2\%$), more than Stage 3 keeps, because sampling noise and the variance criterion, which keeps the items that separate models, spread energy over more components. Stage 3 only has to reproduce the ranking by the mean score, which is one linear functional of the matrix, so a number of categories below the effective rank can preserve it; the exhaustive $k$-sweep measures how far (Table 15).

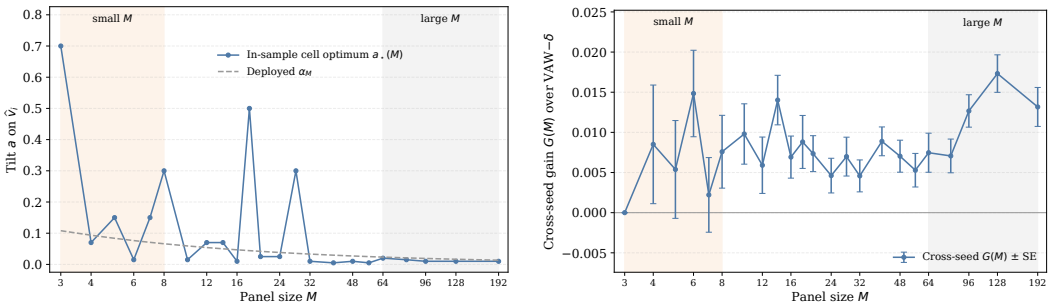
## G Finite-$M$ Scan of the Vision Tilt inside Cells

This section asks whether the tilt $\alpha_M \delta_i^{\mathrm{VL}}$ improves ranking fidelity inside the cells, and how the best tilt depends on the panel size. Without cells the in-sample optimum of the normalised tilt is at most $0.02$ at every $M$, with a gain over Variance of at most $+0.007$ under seeded random tie-breaking. Figure 5 shows the scan inside cells.

The scan uses the 10 OpenVLM benchmarks whose items align with our embeddings (AI2D, DocVQA, InfoVQA, MMT-Bench, MM-Vet, MathVista, OCRBench, RealWorldQA, SEEDBench, TextVQA) and the models that cover at least $95\%$ of a benchmark; at the largest panel sizes only the benchmarks with enough such models remain (down to seven). For each of 23 panel sizes $M$ from 3 to 192 it draws 30 model subsamples. Five seeds split each benchmark's items into halves, and for each seed and fold the training half selects $K = \max(\lfloor 0.02\, n_{\mathrm{train}} \rfloor, 1)$ items ($r$=2%), with $n_{\mathrm{train}}$ the number of items in the training half, while all items are evaluated; $\rho$ is the Spearman correlation over the $M$ models between the kept-item mean and the all-item mean. The cells are drawn by $k$-means from the training-half items' projected vision-text embeddings, fixed across $M$, subsamples and $a$, and for each of 19 tilts $a$ from 0 to 2 and the pipeline's $\alpha_M$, each cell contributes its item with the largest $\widehat{v}_i + a\,\delta_i^{\mathrm{VL}}$. The baseline is VAW$-\delta$ in the same cells ($a = 0$).

The in-sample $a^\star(M)$ maximises the mean gain over the baseline. Because a maximum over a grid is optimistic, the reported gain is a cross-seed gain: the tilt is chosen on two split seeds, and the gain $G(M)$ and its standard error are measured on the other three. We report the trend of $a^\star(M)$ over the 23 panel sizes and, at each size, $G(M)$ with $t = G(M)/\mathrm{SE}$. Ties follow the seeded random order.

The best tilt shrinks as the panel grows: the Spearman correlation between $M$ and $a^\star(M)$ is $-0.664$, with $a^\star$ between $0.005$ and $0.02$ for every $M \ge 32$, of the same order as the deployed $\alpha_M$ ($0.014$–$0.023$ for $M \ge 64$). The tilt chosen on the selection seeds gives a positive cross-seed gain at every $M$




(a) In-sample best tilt $a^\star(M)$ inside cells and the deployed $\alpha_M$.

(b) Cross-seed gain $G(M)$ over VAW$-\delta$ in the same cells, $\pm$ one SE.

Figure 5: **Inside cells the best tilt tends to decrease with the panel size, and cross-seed tilt selection improves ranking fidelity.** In-cell scan on the 10 OpenVLM benchmarks with aligned $\delta^{\mathrm{VL}}$ at $r$=2%: 23 panel sizes $M \in [3, 192]$, 30 model subsamples per $M$, 5 item splits $\times$ 2 folds, and one cell per selected item. The tilt is chosen on two split seeds and its gain over VAW$-\delta$ ($a$=0 in the same cells) is measured on the other three, with seeded random tie-breaking. Shaded: small ($M \leq 8$) and large ($M \geq 64$) panels. The best tilt falls with $M$ (Spearman $-0.664$), and the cross-seed gain is positive at every $M \geq 4$ and exceeds two standard errors at every $M \geq 14$.

from 4 to 192. From $M$=14 on the gain exceeds two standard errors at every size ($G = 0.005$–$0.017$), and at every $M \geq 64$ it is $G = 0.007$–$0.017$ with $t = 3.1$–$7.4$; among $M \leq 8$ it exceeds two standard errors only at $M$=6 ($G = 0.015$, $t = 2.8$). These results support smaller tilts on larger panels, in the direction of the annealing of Section F.2, and inside the cells the cross-seed-selected tilt adds a consistent gain on moderate and large panels.

# H External Validation on OpenVLM

## H.1 OpenVLM data

OpenVLM publishes per-item predictions of up to 265 models on the 14 benchmarks that overlap with our catalogue: AI2D, ChartQA, DocVQA, InfoVQA, MathVista, MME, MMMU, MMT-Bench, MM-Vet, OCRBench, RealWorldQA, ScienceQA, SEEDBench and TextVQA. Per-item correctness uses each benchmark's own metric, with rule-based matching as an offline stand-in for the GPT grading of MM-Vet. All predictions are produced with the image, so the blind-solvability filter of Stage 0, which needs an image-free run, cannot be applied to these records; the all-correct filter, Stage 2 and Stage 3 can. The full pipeline, including the released suite, is validated end to end only on our own panel.

## H.2 Stage 2 on OpenVLM: selector comparison

Every selector keeps $\mathrm{round}(rN)$ items of each benchmark, and $\rho$ compares the ranking on the kept items with the full-benchmark ranking on the same models, so this comparison is same-model. With hundreds of models IRT 2PL leads at $r \leq 2\%$ (0.922 and 0.938) and AnchorPoints from $r$=5% to 20% (Table 17), as expected for response-pattern fits scored on the models they were fitted to. On the 8 benchmarks with a closed answer space, each scored by 200 or more models, DISCO's selection score reaches 0.910 against 0.907 for Variance at $r$=5%.

## H.3 Stage 3 on OpenVLM: benchmark selection

The reference ranking is the mean of per-benchmark $z$-scores over each model's available benchmarks, without imputation, on the 231 models that cover at least 7 of the 14 benchmarks. Exhaustive selection at $k$=5 reaches $\rho$=0.988 (Table 18) with AI2D, MMMU, OCRBench, RealWorldQA and ScienceQA, and greedy selection is within 0.002 of exhaustive at every $k$. The selection is re-fitted on OpenVLM, so this checks the Stage 3 mechanism on a larger and more diverse panel; whether a frozen selection holds for newer models is tested by the shelf-life analysis of Section I.1.

Table 17: **With up to 263 models per benchmark, IRT 2PL leads at $r \leq 2\%$ and AnchorPoints at $r$=5–20%.** Mean Spearman $\rho$ over the 14 OpenVLM benchmarks between the full-benchmark ranking and the ranking on the kept items; selection and scoring use the same OpenVLM models (same-model), each benchmark selected on its own; 200–263 models score ten benchmarks, and ChartQA, DocVQA, InfoVQA and TextVQA have 33, 61, 53 and 126. Best in bold, second best underlined.

| Selector | 1% | 2% | 5% | 10% | 20% | 50% |
|---|---|---|---|---|---|---|
| Variance | 0.818 | 0.889 | 0.932 | 0.953 | <u>0.973</u> | **0.993** |
| DatBench $r_{pb}$ | 0.805 | 0.847 | 0.903 | 0.929 | 0.959 | 0.986 |
| AnchorPoints | <u>0.869</u> | <u>0.931</u> | **0.955** | **0.966** | **0.974** | <u>0.992</u> |
| IRT 2PL | **0.922** | **0.938** | <u>0.952</u> | <u>0.962</u> | 0.971 | 0.988 |

Table 18: **Five of the 14 OpenVLM benchmarks recover the full ranking at $\rho$=0.988.** $\rho$ of exhaustive Stage 3 selection against the 14-benchmark ranking on the 231 models that cover at least 7 benchmarks, with 95% CIs from 10,000 paired bootstrap resamples stratified over 95 model families inferred from model names (53 of them singletons); the CI is computed on the selected subset and omits selection uncertainty. The last column fills missing scores with the benchmark mean over all 265 models and re-selects the subset.

| $k$ | $\rho$ | 95% CI | $\rho$, mean imputation |
|---|---|---|---|
| 3 | 0.978 | [0.973, 0.982] | 0.951 |
| 5 | 0.988 | [0.984, 0.990] | 0.978 |
| 8 | 0.994 | [0.993, 0.995] | 0.990 |

### H.4 Stage 3 on OpenVLM: low rank across release eras

We split the 265 OpenVLM models into three release eras and recompute the spectrum of each era's $z$-normalised matrix over the 14 benchmarks (Table 19). In every era the top five components hold 86–89% of the energy and the participation-ratio effective rank is 7.2–8.1, about half the number of benchmarks; this concentration is what allows a small $k$ in Stage 3.

Table 19: **The top five components hold 86–89% of the OpenVLM spectral energy in every release era.** Spectrum of each era's $z$-normalised model-by-benchmark matrix; the last era has 13 benchmarks because one has no scores, and each model's era is inferred from its name. This concentration is what lets a few benchmarks stand in for all of them.

| Release era | Models | Benchmarks | Components for 95% | Participation ratio | Top-5 energy (%) |
|---|---|---|---|---|---|
| Before Jun 2024 | 73 | 14 | 8 | 7.2 | 88.8 |
| Jun–Dec 2024 | 114 | 14 | 9 | 7.7 | 88.1 |
| From Jan 2025 | 78 | 13 | 8 | 8.1 | 86.0 |

## I Additional Analyses

### I.1 Shelf life of a frozen benchmark selection

We resolve Hugging Face Hub release dates for 203 of the 265 OpenVLM models, sort them by date and append the 62 undated models last. On each cumulative panel the models with at least 7 of the 14 benchmarks are kept first, and the per-benchmark $z$-normalisation is fitted on them afterwards. The exhaustive $k$=5 selection on the 132 oldest dated models is {AI2D, MathVista, MME, MMMU, OCRBench}. Frozen, it reaches $\rho$=0.988, 0.990, 0.989 and 0.989 at $M$=132, 166, 200 and 203 (Table 20), at or above the 99.9th percentile of all 2,002 five-benchmark subsets on each panel, at most 0.0004 below the best subset. On the 55 newer dated models alone, with the normalisation of the $M$=203 panel kept, it reaches 0.989, at the 88th percentile. As a separate stress test, appending the 62 undated models lowers it to 0.971 (65th percentile), where the best subset reaches 0.988. On the dated panels, a greedy selection frozen the same way reaches 0.986–0.988. Because a random five-benchmark subset already reaches $\rho \approx 0.97$ on average on this low-rank matrix, we rank the frozen selection among all subsets rather than against a fixed threshold. The comparison is relative to

other subsets and does not test the item-level release, and Hub release dates are an imperfect proxy for model generation, so we treat the trace as a stress test, not a causal claim.

Table 20: **A five-benchmark selection frozen on the 132 oldest dated OpenVLM models keeps $\rho \approx 0.99$ as newer dated models join, at or above the 99.9th percentile of all 2,002 five-benchmark subsets; appending the 62 undated models lowers it to 0.971.** Cumulative panels in release order; $M$ counts dated models, and the last row appends the 62 undated models. Scored = models with at least 7 of the 14 benchmarks. Gap = best subset $\rho$ minus frozen $\rho$.

| $M$ | Scored | Frozen $\rho$ | Random-five mean | Percentile | Gap |
|---|---|---|---|---|---|
| 132 | 126 | 0.988 | 0.965 | 100.0 | 0.0000 |
| 166 | 150 | 0.990 | 0.970 | 100.0 | 0.0000 |
| 200 | 178 | 0.989 | 0.971 | 99.9 | 0.0001 |
| 203 | 181 | 0.989 | 0.971 | 99.9 | 0.0003 |
| 203 dated + 62 undated | 231 | 0.971 | 0.962 | 64.8 | 0.0168 |

### I.2 Capability Frontier: a descriptive stress set

The Capability Frontier collects the items that most current VLMs fail. An item is a candidate when every family's mean score on it is at most 0.5 (the ten families of Table 3) and at least two models are scored on it; 33,262 items qualify. The Frontier selects only from the 16,692 of them with a panel mean score in $[0.10, 0.50]$, which excludes items that almost every model fails, and keeps those with the largest difficulty-weighted variance $\widehat{v}_i(1 - \bar{x}_i)$. As a heuristic budget, its size applies the largest per-step accuracy gain along the panel's model axes, the Qwen per-generation gain on the full pool (Table 21), to this in-band pool, so that 16,692 $\times$ 0.09902 gives 1,653 items.

We use the Frontier as a descriptive stress set for the current panel. On its items the Qwen-VL-Chat→Qwen3-VL-8B generation gap is 0.62× its full-pool value, so the Frontier still separates the Qwen generations. Because the Qwen family's own mean score helps define the pool, this ratio partly reflects the selection itself. VQAv2 supplies 33% of the items, but model scores weight benchmarks equally, so VQAv2 does not dominate them. We have not tested whether Frontier items become discriminative as newer generations arrive; the shelf-life check of Section I.1 is benchmark-level on OpenVLM and does not bear on these items. The item list is produced by the same pipeline and panel as the released suite and released with it.

Table 21: **The Qwen generation axis gives the largest per-step gain, 0.099, and sets the Frontier size.** Accuracy gain on the full pool between the endpoint models of each axis, divided by the number of steps. The Frontier keeps 16,692×0.09902≈1,653 items.

| Axis | Endpoints | Gain | Steps | Per step |
|---|---|---|---|---|
| Qwen generations | Qwen-VL-Chat → Qwen3-VL-8B | +0.297 | 3 | 0.099 |
| InternVL generations | InternVL2-8B → InternVL3.5-8B | +0.051 | 3 | 0.017 |
| Qwen3 parameter scale | Qwen3-VL-2B → Qwen3-VL-8B | +0.088 | 2 | 0.044 |

### I.3 Left-out benchmark prediction across panel sizes

We evaluate left-out benchmark prediction on four OpenVLM targets: MMT-Bench, ScienceQA, TextVQA, and BLINK. For each target, a Ridge regressor takes the model's scores on the remaining aggregate benchmarks as input (13 features, or 14 for BLINK) and predicts its score on the target benchmark. Depending on the target, 126–214 models have a score on it. We vary the number of training models from $M$=1 to 128 (64 for TextVQA) and report MAE on the remaining models over 30 random splits.

Table 22 lists the MAE of each target at each $M$, and Figure 4a plots it. The MAE decreases from 0.11–0.21 at $M$=1 to 0.04–0.10 at the largest available number of training models. The improvement becomes smaller at larger $M$: between $M$=64 and 128, MMT-Bench remains at 0.039 MAE, while ScienceQA and BLINK improve by at most 0.003. TextVQA is available for 126 models, so $M$=64 is its largest tested training size. The predictors are full benchmark scores, not the released suite, so the analysis probes cross-benchmark redundancy rather than the compressed suite.

Table 22: **Left-out benchmark error falls with the number of training models and flattens beyond $M$=64.** Mean MAE over 30 random splits of a Ridge regressor that predicts each target benchmark from the other OpenVLM aggregate benchmarks, by the number $M$ of training models. Models = models scored on the target, with a missing feature score replaced by its mean over the training models; TextVQA has no tested training size above $M$=64.

| Target | Models | $M$=1 | 2 | 4 | 8 | 16 | 32 | 64 | 128 |
|---|---|---|---|---|---|---|---|---|---|
| MMT-Bench | 200 | 0.114 | 0.089 | 0.074 | 0.061 | 0.051 | 0.044 | 0.039 | 0.039 |
| ScienceQA | 214 | 0.168 | 0.126 | 0.117 | 0.103 | 0.086 | 0.071 | 0.065 | 0.062 |
| TextVQA | 126 | 0.211 | 0.194 | 0.169 | 0.139 | 0.123 | 0.111 | 0.102 | – |
| BLINK | 177 | 0.119 | 0.092 | 0.079 | 0.065 | 0.058 | 0.054 | 0.049 | 0.047 |

### I.4 Checkpoint and hyper-parameter ranking

A compressed suite is also used to rank variants of one model. The variant sets are LoRA fine-tunes (rank 16, two epochs) of Qwen2-VL-2B on ScienceQA: learning-rate grid A (eight rates, final adapter of each), a narrower grid B, eight checkpoints spread over one training run, eight adapters from the converged end of the middle-rate run, and a denser 15-checkpoint trajectory. Grid A is also run with twelve rates, grid B with twelve rates at three training seeds, and grid A with twelve rates on Qwen2.5-VL-3B-Instruct. Items are selected on the ScienceQA records of 30 panel models, and no variant takes part in selection. Fidelity is the Spearman $\rho$ between the ranking of the variants on the selected items and on the full 4,241-item test set. A variant set supports a conclusion only if its full-test ranking is itself stable, with bootstrap $\rho \geq 0.90$, a conservative ceiling that the converged tail and the dense trajectory fail.

On the eight ceiling-passing variant sets at $r$=5% (Table 23), VAW beats Variance by more than one adjacent swap in six variant sets and loses in one, for a mean gain of 0.082. The mean gain comes mostly from the twelve-rate grid B at seeds 43 and 44, where Variance falls to 0.592 and 0.808, while at seed 42 VAW is also ahead (0.908 against 0.870); across the three seeds the VAW lead ranges from +0.04 to +0.28. Because the variants of one training run are not independent draws, we report this comparison descriptively.

Table 23: **VAW raises the checkpoint-ranking fidelity over Variance in six of the eight ceiling-passing variant sets.** Spearman $\rho$ (average ranks for ties) between the ranking of LoRA variants on the selected ScienceQA items and on the full 4,241-item test set; items are selected on 30 panel models, which never include the variants. $n$ = variants; ceiling = bootstrap $\rho$ of the full-test ranking, and variant sets below 0.90 support no conclusion. Without ties, one adjacent swap moves $\rho$ by 0.024 ($n$=8), 0.007 ($n$=12) or 0.004 ($n$=15). Means over the eight ceiling-passing variant sets at $r$=5%: Variance 0.857, VAW 0.939. With 920 pool items, both retentions select the 100-item floor, so their columns coincide. The higher of the two selectors is in bold.

| | | | $r$=5% | | $r$=10% | |
|---|---|---|---|---|---|---|
| Variant set | $n$ | Ceiling | Variance | VAW | Variance | VAW |
| LR grid A | 8 | 1.00 | 0.874 | **1.000** | 0.874 | **1.000** |
| LR grid B | 8 | 0.99 | **0.976** | 0.868 | **0.976** | 0.868 |
| Trajectory | 8 | 0.98 | 0.861 | **0.970** | 0.861 | **0.970** |
| Converged tail | 8 | 0.62 | -0.252 | **0.540** | -0.252 | **0.540** |
| Trajectory, dense | 15 | 0.87 | 0.585 | **0.882** | 0.585 | **0.882** |
| LR grid A, 12 rates | 12 | 1.00 | 0.881 | **0.982** | 0.881 | **0.982** |
| LR grid B, 12 rates, seed 42 | 12 | 0.99 | 0.870 | **0.908** | 0.870 | **0.908** |
| LR grid B, 12 rates, seed 43 | 12 | 0.99 | 0.592 | **0.868** | 0.592 | **0.868** |
| LR grid B, 12 rates, seed 44 | 12 | 0.99 | 0.808 | **0.924** | 0.808 | **0.924** |
| LR grid A, 12 rates, 3B | 12 | 1.00 | **0.991** | 0.989 | **0.991** | 0.989 |

### I.5 Evaluation-compression selectors do not transfer to training-data pruning

To test whether evaluation-compression selectors carry over to training-data selection, we fine-tune a SigLIP-base vision encoder with a BERT-base text encoder and a multiple-choice scalar head on ScienceQA, with no ScienceQA-specific training before this experiment. For each ratio

$r \in \{0.05, 0.1, 0.2, 0.3, 0.5, 0.7, 0.9\}$ and each of three seeds (42, 1337, 2024), we train on the selected fraction of the 11,453-item training pool (a stratified 90% of the 12,726 training items; the other 10% is the early-stopping set) and report accuracy on the full 4,241-item test split. We compare six selectors: Random, Variance, VAW (the released selector, which keeps one item from each of $K$ coverage cells of the vision-text embeddings), DatBench $r_{pb}$ (Joshi et al., 2026), AnchorPoints (Vivek et al., 2024) (here $k$-means on SigLIP image embeddings, with no response signal, unlike the response-based AnchorPoints used for evaluation) and IRT 2PL (Kipnis et al., 2025). The response-based selectors score the training items on the correctness matrix of the 30 panel models. Variance, VAW and DatBench select the same items for every seed; AnchorPoints and IRT vary with it.

No selector beats Random by more than the seed spread (Table 24). At $r \leq 0.3$ VAW is within 3.1 points of test accuracy of Random and ahead of it only at $r$=0.3, by 0.8 points against seed standard deviations of 2–3 points; Variance trails Random by 10.5–20.2 points, and the best of DatBench, AnchorPoints and IRT trails it by 1.8–9.5 points. At $r$=0.5 and 0.7 Random leads every selector by at least 1.3 points, and at $r$=0.9 DatBench and IRT edge past it by 1.7 and 1.1 points. The result indicates that training selection and evaluation compression are different problems, since the first rewards informative training examples and the second rewards items that separate models. We therefore do not recommend evaluation-side item scores, VAW included, for training-data pruning; in this setting uniform random sampling is the safer baseline.

Table 24: **No evaluation-compression selector reliably beats random sampling for training-data selection.** ScienceQA test accuracy, mean over 3 seeds, of a SigLIP+BERT model fine-tuned on the selected fraction $r$ of the training pool. VAW is ahead of Random only at $r$=0.3, and DatBench $r_{pb}$ and IRT 2PL only at $r$=0.9, each within the seed spread. Best in bold, second best underlined.

| Selector | 0.05 | 0.1 | 0.2 | 0.3 | 0.5 | 0.7 | 0.9 |
|---|---|---|---|---|---|---|---|
| Random | **0.542** | **0.619** | **0.707** | <u>0.735</u> | **0.782** | **0.805** | 0.804 |
| Variance | 0.437 | 0.473 | 0.505 | 0.565 | 0.676 | 0.748 | 0.796 |
| VAW | <u>0.541</u> | <u>0.607</u> | <u>0.676</u> | **0.743** | <u>0.751</u> | 0.773 | 0.803 |
| DatBench $r_{pb}$ (Joshi et al., 2026) | 0.512 | 0.549 | 0.612 | 0.670 | 0.723 | 0.769 | **0.821** |
| AnchorPoints (Vivek et al., 2024) | 0.524 | 0.545 | 0.572 | 0.633 | 0.739 | <u>0.792</u> | 0.782 |
| IRT 2PL (Kipnis et al., 2025) | 0.452 | 0.477 | 0.526 | 0.569 | 0.704 | 0.761 | <u>0.815</u> |

## J LIMITATIONS

**Panel size and evaluation noise.** With 10 to 20 held-out models per split, the split-to-split variation of $\rho$ is comparable to the gaps between the stronger selectors, so we report means over 20 splits, with the number of splits each selector wins, and make no claim that depends on a single split.

**External validation.** On OpenVLM we validate Stage 3 at the benchmark level and compare Stage 2 selectors under same-model scoring; the blind-solvability filter cannot be rerun there, because the records carry no image-free predictions. The released suite is validated end to end only on our panel.

**Scope of the theory.** The theoretical results show that the effect of the annealed tilt on the plug-in variance shrinks at the rate of the estimation noise and that, for a fixed pool and fixed cells with a boundary-gap condition (A3), the cell selector recovers the per-cell variance choice as $M$ grows; at the panel sizes used here they give no finite-sample guarantee, and they do not show that the tilt or the cells improve ranking fidelity. Evidence on those questions comes from the experiments of Sections D.4 and G.

**Size of the vision contribution.** Embedding cells beat random partitions by about 0.011 in $\rho$, and text-only cells do as well as vision-text cells, so the gain comes from grouping similar items and does not require image inputs to build the cells.

**Cross-benchmark prediction.** Scores on the other OpenVLM benchmarks predict a left-out benchmark with an error that falls as the number of training models grows (Figure 4a), but the error

is still $0.04$–$0.10$ MAE at the largest panels available, and we have not tested the released suite itself as an estimator.

**Judges and modality scope.** Scores on the judge-scored benchmarks inherit the errors of their judge. The panel contains only open-weight models on image–text benchmarks; proprietary API models and video or audio benchmarks are not covered. Stages 1 and 3 consume only the score matrix and are modality-agnostic, and Stage 0 extends to another modality by rerunning inference with that modality removed; Stage 2 would need a dependence score and an embedding for the new modality.

## K Broader Impact

PRIMEBench lowers the cost of VLM evaluation by replacing repeated full-suite evaluation with a compact, ranking-preserving subset. This can make systematic VLM evaluation more accessible to small-budget research groups, reduce duplicated GPU expenditure across model releases, and lower the environmental cost of large-scale benchmarking. The released suite also provides a common reference point for comparing new models without requiring every group to rerun hundreds of thousands of benchmark items.

The main risk is that aggressive compression may narrow the community's attention towards capabilities that are already well represented in existing benchmarks. A compact suite should therefore not be treated as a substitute for comprehensive evaluation in high-stakes deployment settings or for auditing newly emerging capabilities. To mitigate this, PRIMEBench is released as a hierarchical pipeline rather than only as a fixed item set: users can stop at less compressed stages when broader coverage is required, and future versions can refresh the item pool as new model families and new capability axes appear.