Local Sparsity Enables Unsupervised LLM Safety Detection

Xin Chen[1,2] **Gil Kur**[1*] **Alexander Shevchenko**[1*] **Andreas Krause**[1]

[1]ETH Zürich [2]Solaris Research

Abstract

Deployment-time safety methods for large language models (LLMs) are predominantly supervised and assume access to unsafe training data. Nevertheless, new attacks and harm categories regularly arise, not captured by models trained in such a supervised fashion. An alternative approach is to view this problem through the lens of anomaly detection, namely, to rely solely on modeling safe data and flagging out-of-distribution inputs. However, LLM activations lie in a high-dimensional space, raising concerns about whether anomaly detection is statistically feasible. We show that, under the linear representation hypothesis (LRH), there may indeed be hope. In the LRH concept space, which is typically recovered via a sparse autoencoder (SAE), nearby points share a small common active support. Using this local sparsity insight, we propose a framework for locally masked SAE-based anomaly detection, supported by theoretical justifications. We validate it on various architectures and datasets, including both capability-testing datasets and safety-specific datasets. Finally, when we allow algorithms to use 1% out-of-distribution data for calibration, locally sparse methods achieve near-optimal performance, demonstrating their ability to capture meaningful safety information while using only 1-2% of SAE neurons for computation.

1 Introduction

Supervised safety classifiers for Large Language Models (LLMs) [Sharma et al., 2025, Bai et al., 2022, Chen et al., 2025b, Zou et al., 2024] assume access to labeled unsafe examples. In practice, this turns out to be a significant limitation since unsafe distribution may change drastically during deployment. Examples of such distribution shift include: the emergence of new jailbreaks [Andriushchenko et al., 2024, Nasr et al., 2025], harmful behaviors absent from the labeled set [Turner et al., 2025], and inputs from domains the model never encountered before [Oh et al., 2025]. In this view, a supervised procedure fit to a fixed unsafe distribution might generalize poorly beyond it.

To address this limitation, we take an unsupervised view and frame the problem as anomaly detection [Schölkopf et al., 1999, Perera et al., 2021]. Namely, given safe data $\{x_i\}_{i=1}^n \sim \mathbb{P}$, we aim to construct a score function $s : \mathcal{X} \to \mathbb{R}$ that takes large values on inputs atypical under $\mathbb{P}$. Crucially, this construction does not require unsafe data, and is therefore not biased toward any particular harm category. Despite its conceptual appeal, this approach remains underexplored for modern LLMs. The reason is that LLM embeddings are high-dimensional, and unsupervised anomaly detection methods are known to perform poorly in this setting [Aggarwal et al., 2001, Zimek et al., 2012].

While this might appear to be the case at first sight, LLM activations exhibit far more structure than their nominal dimension suggests. The Linear Representation Hypothesis (LRH) [Park et al., 2023,

*Equal contribution. Correspondence to: xin.chen@inf.ethz.ch

Code is available at https://github.com/lasgroup/unsupervised-llm-safety.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

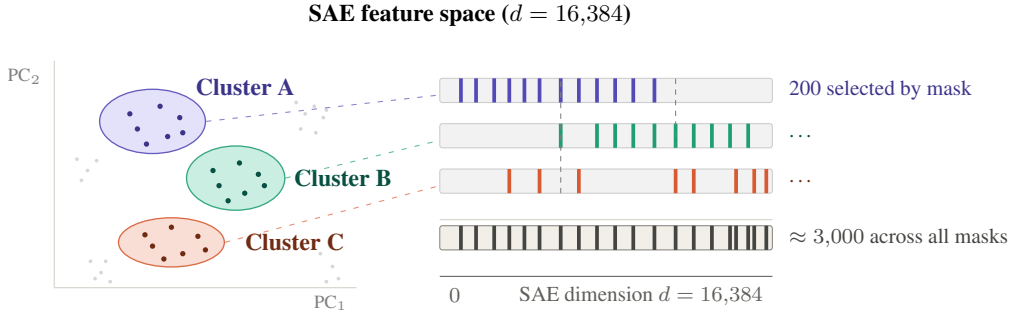

Figure 1: Per-cluster masks in SAE's feature space. *Left:* 2D projection of $\mathbb{R}^d$ with three highlighted clusters. *Right:* the indices of features selected by the mask m_c for each cluster. Each mask retains 200 features, and the masks are partially overlapping; their union covers approximately 3,000 features, still much smaller than d.

2025a] posits that each embedding is a *sparse* combination of concept vectors. This concept space can be approximately recovered using a sparse autoencoder (SAE) [Cunningham et al., 2023, Paulo et al., 2024, He et al., 2024]. Hübotter et al. [2025] further observe that within local "neighborhoods", the active features are largely shared (cf. Figure 1). Consequently, anomaly detection can in principle operate on a much lower-dimensional representation. However, this structure is *local*: the relevant low-dimensional subspace varies across neighborhoods. Methods that instead impose a single, globally compressed representation, such as standard autoencoders, substantially underperform (cf. CVDD and Deep SVDD in Figure 2). We leverage this local structure in the following contributions.

Modeling. In Section 3, we isolate four modeling choices through which local sparsity can be exploited for anomaly detection: **(i)** embedding space, **(ii)** clustering, **(iii)** choice of sparse subspace within each cluster, and **(iv)** scoring rule. We instantiate this modeling with two methods: FREQMASK-KM, which is training-free at the cluster level, and LEARNEDMASK-LORA, which fits per-cluster low-rank adapters.

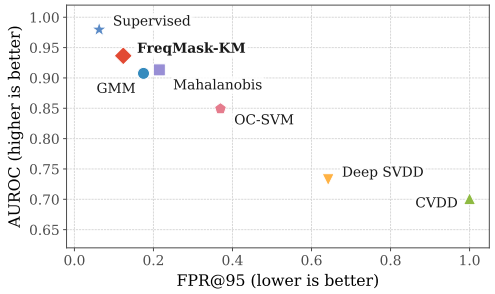

Figure 2: **Local sparsity in LLM activations enables one-class safety detection.** AUROC vs. FPR@95 on LLaMA3-8B for FREQMASK-KM against one-class baselines (Mahalanobis, CVDD) and a supervised reference.

Numerics. In Section 4, we validate our findings across six instruction-tuned models (Qwen2-1.5B [Bai et al., 2023], Ministral-8B [Liu et al., 2026], LLaMA3-8B [Grattafiori et al., 2024], Qwen3-8B [Yang et al., 2025], GPT-oss-20B [Agarwal et al., 2025], and Gemma 4-26B [Team, 2026]) and three categories of unsafe behavior. We show that several sparsity-aware algorithms perform on par or better than standard baselines. Ablations confirm that *local sparsity* is essential for anomaly detection on LLM representations—any successful method must exploit it, either explicitly by design (as ours does) or implicitly through inductive bias.

Theoretical justification. In Section 6, we prove a sample-complexity bound for learning the corresponding anomaly detector. Notably, the bound scales only *logarithmically* in the concept-space dimension d_2 (hidden dimension of SAE) and the ambient dimension d_1, and *only linearly* in the local sparsity. This explains why anomaly detection does not fail in this high-dimensional regime.

2 Background: Linear Representation Hypothesis and Local Sparsity

The Linear Representation Hypothesis posits that high-level concepts are encoded as linear directions in LLM activation space [Mikolov et al., 2013, Park et al., 2023]. Language models represent more concepts than their residual stream has dimensions by packing them into near-orthogonal directions, a.k.a. *superposition* [Elhage et al., 2022]. Sparse autoencoders approximately recover the corresponding concept space as an overcomplete dictionary, such that each input activates only a small subset of hidden neurons [Cunningham et al., 2023, Bricken et al., 2023, Templeton et al., 2024, Gao et al., 2024]. Hübotter et al. [2025] puts this idea further by showing that the active features

substantially overlap across suitable "neighbors". In this section, we make these observations formal and define the set of notations used throughout the rest of the paper.

Residual stream and SAE. Let $\mathcal{X}$ denote the input space of token sequences. For a fixed layer ℓ of a transformer, we define the corresponding residual-stream activation map by $a^{(\ell)} : \mathcal{X} \to \mathbb{R}^{d_1}$, where d_1 stands for the dimension of residual-stream. A sparse autoencoder at layer ℓ is defined by its encoder-decoder pair: $E : \mathbb{R}^{d_1} \to \mathbb{R}^{d_2}, D : \mathbb{R}^{d_2} \to \mathbb{R}^{d_1}, d_2 \gg d_1$, where $E = \mathrm{ReLU}(W_E x + b_E)$ with $D = W_D x + b_D$, for $W_E, W_D^\top \in \mathbb{R}^{d_2 \times d_1}$, $b_E \in \mathbb{R}^{d_2}$, $b_D \in \mathbb{R}^{d_1}$ and $\mathrm{ReLU}(a) = \max\{0, a\}$ is applied component-wise. The optimal configuration of parameters $\Theta = (W_E, W_D, b_E, b_D)$ is selected to minimize *regularized* L_2-reconstruction loss over activations $a^\ell(x)$ of safe inputs $x \sim \mathbb{P}$:

$$\mathcal{L}_{\mathrm{SAE}}(E, D) \;:=\; \mathbb{E}_{x\sim\mathbb{P}}\left[\left\|a^{(\ell)}(x) - D\big(E(a^{(\ell)}(x))\big)\right\|_2^2\right] \;+\; \lambda\,\mathbb{E}_{x\sim\mathbb{P}}\left[\left\|E(a^{(\ell)}(x))\right\|_1\right] \to \min_{\Theta}, \quad (1)$$

where $\lambda > 0$ is a regularization coefficient. $\mathrm{ReLU}(\cdot)$ activation in conjunction with L_1-penalty on SAE features $z(x) := E(a^{(\ell)}(x)) \in \mathbb{R}^{d_2}$ ensures that the resulting representations $z(x)$ are indeed sparse. The corresponding active features of $z(x)$ form an active support $\phi(x) := \{\, j \in [d_2] : z(x)_j \neq 0 \,\}$ of non-zero entries of cardinality $|\phi(x)| = \|z(x)\|_0$.

Global (average) sparsity. The key property of a well-trained SAE is that, despite extreme overparameterization $d_2 \gg d_1$, each input activates only a small subset of concepts. Formally:

$$\mathbb{E}_{x\sim\mathbb{P}} \|z(x)\|_0 \;=:\; s \;\ll\; d_2. \quad (2)$$

Across the SAEs we use, average sparsity s is two to three orders of magnitude smaller than d_2.[2]

Local sparsity. Central to this work is a stronger property—the joint sparsity structure within a "neighborhood". For a fixed metric ρ on the concept space (see examples in Section 3.2), a safe input $x \in \mathcal{X}_0 := \mathrm{supp}(\mathbb{P})$ and a radius $r > 0$, we denote the safe-input neighborhood of x at scale r by

$$\mathcal{N}_r(x) \;:=\; \{\, x' \in \mathcal{X}_0 : \rho(z(x), z(x')) \le r \,\}.$$

The local-sparsity property posits that for some "not small" $r_0 \ge 0$, the *union* of active supports over the corresponding neighborhood is significantly smaller than the dimension of the concept space:

$$\mathbb{E}_{x\sim\mathbb{P}}\left|\bigcup_{x'\in\mathcal{N}_{r_0}(x)} \phi(x')\right| \;=:\; s_{\mathrm{loc}} \;\ll\; d_2. \quad (3)$$

In words, active features are largely *shared* across nearby points rather than being distinct. This implies that our data lies in a manifold containing "sparse patches", which justifies the use of "local sparsity" terminology. As we will see, this assumption is motivated by the empirical validation on six model families in Section 4, and it is central to our theoretical analysis in Section 6.

Implication for one-class anomaly detection. Equation (3) is the structural property our modeling heavily relies on. If we design an anomaly detection algorithm that operates on sparse local patches implicitly (or by inductive bias), the effective dimension it faces amounts to s_{loc} rather than d_2. We make it precise in the upcoming sections: Section 3 focuses on possible modeling choices (embedding space, clustering, local subspace, scoring) to ensure (3) holds; Section 4 confirms this picture empirically across six model families; and Section 6 shows that, as long as (3) holds, the sample complexity of the resulting estimator scales linearly in s_{loc} and only logarithmically in d_2.

3 Framework for Local Sparse Anomaly Detection

We now describe a framework that poses anomaly detection on LLM activations as local problems for which their effective dimension is determined by s_{loc} rather than by d_2. The framework consists of four stages: the *design of embedding space* in which the detector operates, the *clustering procedure* $\mathcal{C} = \{C_1, \ldots, C_{\mathcal{N}_c}\}$ to recover safe neighborhoods, the *local sparse mask* $m_c \in \{0,1\}^{d_2}$ identified within each cluster C_c, and the *scoring* rule used to evaluate new test examples. Each stage admits several principled choices, detailed in Sections 3.1 to 3.4. We treat these steps as a design space rather than a single algorithm: different model geometries may favor different instantiations, and the experiments in Section 4 support this modular view.

[2] For some of the models we consider the corresponding values are $(d_1, d_2, s) \approx (4096, 16384, 100)$.

3.1 Embedding Space

The first stage selects the space in which all subsequent phases operate. Recall that the concepts are entangled in the *dense* activations $a^{(\ell)} : \mathcal{X} \to \mathbb{R}^{d_1}$ for layer ℓ. Therefore, searching for the local sparse structure of Equation (3) directly in $a^{(\ell)}$ is hopeless. We address this by lifting the activations to the sparse space induced by an SAE. By construction, the corresponding map $z : \mathcal{X} \to \mathbb{R}^{d_2}$ with $d_2 \gg d_1$ produces sparse representations, i.e., $\|z(x)\|_0 \ll d_2$. This structure enables the subsequent clustering to identify neighborhoods compatible with Equation (3). We adopt a $\mathrm{ReLU}(\cdot)$ activation in the encoder $E(\cdot)$ with an L_1 penalty on $z(x)$, which enforces sparsity on average. Kantamneni et al. [2025] analyzes the role of SAE in supervised probing, which we discuss more in Appendix C.3.

3.2 Clustering

The clustering stage partitions the safe distribution into regions for which the local-sparsity property (3) is expected to hold. Formally, given a finite set of safe examples $\mathcal{D} = \{x_1, \ldots, x_n\} \subset \mathcal{X}_0$ and a fixed choice of metric ρ on the embedding space, the outcome of this stage is a partition $\mathcal{C} = \{C_1, \ldots, C_{\mathcal{N}_c}\}$ of $\mathcal{D}$ obtained by running K-means with metric ρ on the chosen embeddings of $\mathcal{D}$. Two design choices govern this partition: the metric ρ and the number of clusters $\mathcal{N}_c$.

Distance metric. The choice of metric set the trade-off between active support and activation "magnitude" when deciding which points are grouped together. Let $b(x) := \mathbf{1}[z(x) > 0] \in \{0,1\}^{d_2}$ be the active-feature indicator of sparse representations $z(x)$. We examine three points along this trade-off: **(i)** the Hamming distance $\rho_{L_0}(x, x') = \|b(x) - b(x')\|_0$, which depends on z only through b and groups the points *exlusively* by their active support $\phi(x)$, explicitly targeting small values of s_{loc}; **(ii-iii)** the magnitude-aware $\rho_{L_p}(x, x') = \|z(x) - z(x')\|_p$ for $p \in \{1, 2\}$, which retain activation influence, i.e., it groups points $\mathcal{D}$ by both which features fire and how strongly, at the cost of greater sensitivity to noise in the activation magnitudes.

Number of clusters. The number of clusters $\mathcal{N}_c$ trades off cluster homogeneity against per-cluster sample size. With too few clusters, the union of inner-cluster supports $\bigcup_{x \in C_c} \phi(x)$ grows and (3) degrades. With too many, per-cluster sample sizes $|C_c| \approx N/\mathcal{N}_c$ become small, and parametric per-cluster estimators (e.g., per-cluster covariance for Mahalanobis scoring) become ill-conditioned. An appropriate choice of $\mathcal{N}_c$ takes into account both the safe sample size N and the global sparsity s.

3.3 Local sparse subspace

At this stage, for each cluster $C_c \in \mathcal{C}$, we identify a sparse subspaces of $\mathbb{R}^{d_2}$ on which inputs are scored. The subspace is identified by its binary mask $m_c \in \{0,1\}^{d_2}$, applied to SAE features $z(x)$ via the Hadamard product $m_c \odot z(x)$. We consider two constructions below.

Frequency mask. For an index set $I \subseteq \mathcal{D}$, let $f^{(I)} \in [0,1]^{d_2}$ be the per-feature activation frequency over I, $f^{(I)}_j = \frac{1}{|I|} \sum_{x \in I} \mathbf{1}\big[z(x)_j \neq 0\big], j \in [d_2]$, and define the corresponding top-k mask by $m^{(I)} := \mathbf{1}\big[f^{(I)} \in \mathrm{top}_k(f^{(I)})\big]$, where $\mathrm{top}_k(\cdot)$ returns the indices of the k largest entries. We distinguish between two choices for I: **(i)** we set $I = \mathcal{D}$ and apply the same *global mask* in each cluster ($m_c := m^{(\mathcal{D})}$ for all $c \in [\mathcal{N}_c]$), leaving the clustering as the only source of local information; **(ii)** alternatively, we set $I = C_c$ and apply *per-cluster mask* $m_c := m^{(C_c)}$.

Per-cluster learned subspace. An alternative is to let the SAE adjust locally via per-cluster low-rank adapters [Hu et al., 2022]: $E_c = E + \Delta E_c$, $D_c = D + \Delta D_c$, with $\mathrm{rank}(\Delta E_c), \mathrm{rank}(\Delta D_c) \leq r \ll \min(d_1, d_2)$[3]. The mask m_c and the adapters $(\Delta E_c, \Delta D_c)$ are then jointly optimized on C_c against an auxiliary objective. Fitting E and D directly on each cluster, without the rank constraint above, was found in preliminary experiments to increase the held-out reconstruction loss and the average density of $z(x)$ on safe data, relative to the initial SAE. We therefore restricted per-cluster updates to the *low*s rank-r adapters $(\Delta E_c, \Delta D_c)$.

3.4 Scoring

The final stage assigns each test point x an anomaly score given its cluster $c^\star(x) \in [\mathcal{N}_c]$ and the local mask $m_{c^\star}$. We use the nearest-centroid assignment $c^\star(x) = \arg\min_{c \in [\mathcal{N}_c]} \rho\big(z(x), \mu_c\big), \mu_c = \frac{1}{|C_c|} \sum_{x' \in C_c} z(x')$, where ρ is the clustering metric (Section 3.2). We consider two scoring rules.

[3]With slight abuse of notation, we write E for W_E and D for W_D.

Algorithm 1 FREQMASK-KM

Require: Safe set $\mathcal{D}$, SAE encoder E, layer ℓ, clusters number $\mathcal{N}_c$, sparsity k

Fit:

$z_i \leftarrow E(a^{(\ell)}(x_i))$ for $x_i \in \mathcal{D}$ ▷ SAE features

$\{C_1, \ldots, C_{\mathcal{N}_c}\} \leftarrow \text{KMEANS}_{L_2}(\{z_i\})$

$\mu_c \leftarrow \text{mean}(\{z_i : i \in C_c\})$ ▷ cluster centroid

$b_i \leftarrow \mathbf{1}[z_i > 0]$ ▷ binarize

$f_j \leftarrow \frac{1}{N}\sum_i (b_i)_j$ ▷ pooled frequency

$m \leftarrow \mathbf{1}[f \in \text{top-}k(f)]$ ▷ global mask

Scoring x**:**

$z \leftarrow E(a^{(\ell)}(x))$

$c^\star \leftarrow \arg\min_c \|z - \mu_c\|_2$ ▷ assign cluster

Decision (given threshold τ)**:**

$s(x) \leftarrow \|m \odot z - m \odot \mu_{c^\star}\|_1$

return UNSAFE if $s(x) > \tau$, else SAFE

Algorithm 2 LEARNEDMASK-LORA

Require: Safe set $\mathcal{D}$, SAE (E, D), layer ℓ, clusters $\mathcal{N}_c$, sparsity λ

Training:

$z_i \leftarrow E(a^{(\ell)}(x_i))$ for $x_i \in \mathcal{D}$

$\{C_1, \ldots, C_{\mathcal{N}_c}\} \leftarrow \text{KMEANS}_{L_2}(\{z_i\})$

for each cluster c **do** ▷ Stage 1: binary mask

$m_c \leftarrow \arg\min_{m \in \{0,1\}^d} \mathcal{L}_{\text{rec}} + \lambda\|m\|_1$

for each cluster c **do** ▷ Stage 2: LoRA adapters

$(\Delta E_c, \Delta D_c) \leftarrow \arg\min \mathcal{L}_{\text{rec}}$

Scoring x**:**

$z \leftarrow E(a^{(\ell)}(x)),\ c^\star \leftarrow \arg\min_c \|z - \mu_c\|_2$

Decision (given threshold τ)**:**

$s(x) \leftarrow \|a^{(\ell)}(x) - D_{c^\star}(m_{c^\star} \odot E_{c^\star}(a^{(\ell)}(x)))\|_2$

return UNSAFE if $s(x) > \tau$, else SAFE

Centroid distance in the masked SAE feature space, $s_{\text{dist}}(x) = \left\|m_{c^\star} \odot \left(z(x) - \mu_{c^\star}\right)\right\|_p, p \in \{1, 2\}$, or, when per-cluster covariance estimates are reliable, its Mahalanobis variant. This is the standard typical-set realization: atypical points lie far from the local centroid in the active-feature subspace.

Reconstruction residual under the cluster's local encoder/decoder pair $(E_{c^\star}, D_{c^\star})$, $s_{\text{rec}}(x) = \left\|a^{(\ell)}(x) - D_{c^\star}\left(m_{c^\star} \odot E_{c^\star}(a^{(\ell)}(x))\right)\right\|_2$, matched to the learned-subspace construction in Section 3.3 for reconstruction loss as the auxiliary objective.

Per-cluster calibration. Both centroid and residual scores use global thresholding, i.e., the cut-off parameter for a decision rule is independent of the cluster assigned to the evaluation point. Empirically, we notice that cluster assignment is highly non-homogeneous for unsafe data – unsafe samples concentrate in a small subset of the safe clusters. Such discrepancy can be exploited for calibration at a cost of a small labeled set ($< 1\%$ of data). We demonstrate that the scoring rule can be adjusted per cluster via a simple affine map. We further discuss it in Section 5 and Appendix B.

3.5 Two concrete instantiations

We describe two instantiations of this framework, FREQMASK-KM and LEARNEDMASK-LORA, with no per-cluster learned parameters and with per-cluster low-rank adaptation, respectively. Pseudocode is given in Algorithms 1 and 2.

FREQMASK-KM. The pipeline operates on SAE features $z(x) \in \mathbb{R}^{d_2}$. Clustering uses Lloyd's algorithm with uniform random centroid initialization (FAISS [Johnson et al., 2019] `Kmeans` defaults), minimizing $\sum_{c=1}^{\mathcal{N}_c} \sum_{x \in C_c} \|z(x) - \mu_c\|_2^2$. The local subspace uses the global frequency mask m_{global}, shared across all clusters. Scoring is the ℓ_1 centroid distance: $s_{\text{dist}}(x) = \|m_{\text{global}} \odot (z(x) - \mu_{c^\star})\|_1$.

LEARNEDMASK-LORA. The pipeline operates on SAE features $z(x)$ and uses the same clustering as FREQMASK-KM. The local subspace is induced by per-cluster LoRA adapters $(\Delta E_c, \Delta D_c)$, fit with the SAE backbone frozen and m_c is learned by minimizing the per-cluster reconstruction loss $\mathcal{L}_{\text{rec}} = \mathbb{E}_{x \in C_c} \|a^{(\ell)}(x) - D_c(m_c \odot E_c(a^{(\ell)}(x)))\|_2^2$. Scoring is done via the reconstruction residuals $s_{\text{rec}}(x)$ under the assigned cluster's adapted encoder-decoder $(E_{c^\star}, D_{c^\star})$.

These two instances interpolate the design space defined in Section 3.3: the first fits no per-cluster parameters, the second fits a low-rank adapter per cluster against the downstream scoring objective. Section 4 shows that both reach competitive AUROC across six model families. Detailed ablation studies can be found in Appendix F and Appendix G, where we vary FREQMASK-KM and LEARNEDMASK-LORA one stage at a time: clustering metric, number of clusters $\mathcal{N}_c$, mask size k, mask on versus off, scoring rule, within-cluster overlap of the safe and unsafe active supports, extraction layer, SAE sparsity penalty, etc., along with many more detailed analysis.

4 Experiments

4.1 Setup

Models. We evaluate on six instruction-tuned model families: `qwen2-1.5b-instruct` [Bai et al., 2023], `ministral-8b-instruct` [Liu et al., 2026], `llama3-8b-instruct` [Grattafiori et al., 2024],

Table 1: AUROC and TPR@5%FPR for two framework instantiations, five unsupervised baselines, and a supervised linear probe across six instruction-tuned model families. The "Local" column marks whether the method scores on a sparse local subspace (•) or not (−). Results are means ± standard deviations across 5 random seeds. Local-sparsification instantiations are the best unsupervised method on 4 models, and for the rest, a close second (within 0.025 AUROC). TPR@5%FPR rankings are less uniform. In each column the best unsupervised result is in **bold** and the second best is underlined; the supervised probe is excluded from this ranking. The five ≥8B models perform substantially better than Qwen 1.5B, indicating that representation quality is key to successful anomaly detection.

AUROC (↑)

Method	Local	Ministral 8B	LLaMA3 8B	Qwen3 8B	GPT-oss 20B	Gemma 4 26B	Qwen 1.5B
BCE (linear probe, supervised)	–	0.993 ± 0.000	0.972 ± 0.003	0.993 ± 0.001	0.990 ± 0.001	0.983 ± 0.001	0.982 ± 0.002
FREQMASK-KM	•	$\mathbf{0.914 \pm 0.000}$	$\mathbf{0.937 \pm 0.000}$	0.915 ± 0.000	0.900 ± 0.000	0.856 ± 0.000	0.762 ± 0.000
LEARNEDMASK-LORA	•	0.896 ± 0.001	0.912 ± 0.001	0.917 ± 0.000	0.919 ± 0.001	$\mathbf{0.904 \pm 0.000}$	$\mathbf{0.794 \pm 0.001}$
Mahalanobis	–	0.892 ± 0.000	0.913 ± 0.000	0.914 ± 0.000	0.843 ± 0.000	0.901 ± 0.000	0.756 ± 0.000
GMM	–	0.881 ± 0.011	0.906 ± 0.002	$\mathbf{0.942 \pm 0.002}$	$\mathbf{0.925 \pm 0.008}$	0.861 ± 0.004	0.780 ± 0.005
One-Class SVM	–	0.829 ± 0.002	0.841 ± 0.011	0.885 ± 0.007	0.850 ± 0.002	0.886 ± 0.003	0.756 ± 0.004
Deep SVDD	–	0.389 ± 0.152	0.475 ± 0.183	0.389 ± 0.118	0.535 ± 0.024	0.665 ± 0.026	0.507 ± 0.101
CVDD	–	0.712 ± 0.029	0.665 ± 0.037	0.705 ± 0.029	0.706 ± 0.059	0.672 ± 0.042	0.545 ± 0.021

TPR@5%FPR (↑)

Method	Local	Ministral 8B	LLaMA3 8B	Qwen3 8B	GPT-oss 20B	Gemma 4 26B	Qwen 1.5B
BCE (linear probe, supervised)	–	0.964 ± 0.001	0.905 ± 0.014	0.957 ± 0.004	0.948 ± 0.003	0.942 ± 0.003	0.876 ± 0.014
FREQMASK-KM	•	0.675 ± 0.000	$\mathbf{0.457 \pm 0.000}$	0.429 ± 0.000	0.535 ± 0.000	0.264 ± 0.000	0.287 ± 0.000
LEARNEDMASK-LORA	•	0.685 ± 0.001	0.450 ± 0.009	0.581 ± 0.001	0.719 ± 0.002	$\mathbf{0.642 \pm 0.003}$	$\mathbf{0.359 \pm 0.006}$
Mahalanobis	–	0.458 ± 0.000	0.409 ± 0.000	0.458 ± 0.000	0.421 ± 0.000	0.641 ± 0.000	0.264 ± 0.000
GMM	–	0.664 ± 0.009	0.381 ± 0.010	$\mathbf{0.639 \pm 0.008}$	$\mathbf{0.725 \pm 0.018}$	0.624 ± 0.014	0.343 ± 0.039
One-Class SVM	–	$\mathbf{0.694 \pm 0.004}$	0.185 ± 0.020	0.252 ± 0.026	0.567 ± 0.006	0.512 ± 0.031	0.217 ± 0.004
Deep SVDD	–	0.022 ± 0.012	0.036 ± 0.015	0.038 ± 0.019	0.078 ± 0.033	0.116 ± 0.042	0.063 ± 0.018
CVDD	–	0.358 ± 0.080	0.348 ± 0.077	0.416 ± 0.056	0.428 ± 0.111	0.352 ± 0.087	0.132 ± 0.041

`qwen3-8b`, `gpt-oss-20b`, and `gemma-4-26b`. Activations are extracted from the residual stream at layer 20 for Qwen2-1.5B, layer 20 for Ministral, layer 25 for LLaMA3, layer 30 for Qwen3, layer 20 for GPT-oss, and layer 25 for Gemma-4-26B. Layers were chosen at mid-to-late depth, where safety-relevant features have emerged but the representation has not yet narrowed toward the output vocabulary, and were not tuned against downstream AUROC. For every input, we record the residual-stream activation of the model's own generated response at the final response token.

Data. The safe set pools general-capability and instruction-following data with the safe portions of the evaluation benchmarks, totaling approximately 206K examples per model; per-source composition, splits, and licenses are deferred to Appendix D.2. We evaluate on three unsafe categories: *BeaverTails* [Ji et al., 2023] (requests for harmful content), *ToxiGen* [Hartvigsen et al., 2022] (toxic and hateful language), and *HarmBench* [Mazeika et al., 2024] adversarial attacks, pooled across eight attack algorithms. Reported AUROC and TPR@5%FPR are means across these three categories.

Baselines, metrics, and method of focus. We compare against established one-class methods: global Mahalanobis distance [Nader et al., 2014], Gaussian Mixture Model (GMM), One-Class SVM [Schölkopf et al., 1999], Deep SVDD [Ruff et al., 2018], and CVDD [Ruff et al., 2019]. We also report a supervised linear probe as a reference ceiling; the probe is trained with access to unsafe labels and is not a competitor to the unsupervised framework. For each method, we report its AUROC and TPR@5%FPR (true positive rate at a 5% false positive rate), with additional details in Appendix D.4.

4.2 Unsupervised Anomaly Detection on LLM Activations Is Feasible

Table 1 reports 3-category AUROC and TPR@5%FPR across the six models. On four of the five ≥8B models the two local-sparsification instantiations land in a tight band. Gemma is the exception: its global-mask instantiation FREQMASK-KM is weaker, while the learned instantiation LEARNEDMASK-LORA stays in band and is the best unsupervised method on Gemma. FREQMASK-KM is the best unsupervised method on Ministral and LLaMA3; on Qwen3 and GPT-oss the GMM density baseline is marginally ahead (0.942 and 0.925), with a local instantiation second within 0.025 AUROC, and on Gemma LEARNEDMASK-LORA leads with Mahalanobis a close second (0.904 vs. 0.901). The representation-learning baselines trail substantially, indicating that their optimizations cannot find a useful representation in this setting. A roughly 0.04–0.08 AUROC gap to the supervised linear probe remains on the ≥8B models, which we read as the cost of dropping the assumption that the unsafe distribution is known in advance (Section 5).

Table 2: Decomposition of framework instantiations and baselines along three design axes: clustering, sparsification, and scoring. Performance is averaged across the five $\geq$8B models and 5 random seeds. The "Local" column marks whether scoring uses a sparse local subspace (•) or not (−). We use the mean AUROC from Table 1 across those five models. Methods in the upper block operate in the raw LLM activation space; methods in the lower block operate in an SAE feature space. Moving from activation space to SAE space in CVDD contributes a large share of the gain; once in SAE space, local-subspace constructions FREQMASK-KM and LEARNEDMASK-LORA land in the same AUROC band despite differing substantially.

Method	Local	Clustering	Sparsification	Scoring	AUROC
Mahalanobis	–	None	Full covariance	Mahalanobis distance	0.893
GMM	–	Mixture models	None	Mixture likelihood	0.903
CVDD	–	KMeans	None	L_2 to cluster centroid	0.692
CVDD (SAE space)	–	KMeans	None	L_2 to cluster centroid	0.806
FREQMASK-KM	•	KMeans	Global top-k	L_1 to cluster centroid	0.904
Local FREQMASK-KM	•	KMeans	per-cluster top-k	L_1 to cluster centroid	0.790
LEARNEDMASK-LORA	•	KMeans	per-cluster learned	SAE residual	**0.909**

Table 3: Comparison between the mask disabled against the deployed masked configuration. The masks improve the bigger models more significantly.

	Qwen 1.5B	Ministral 8B	LLaMA3 8B	Qwen3 8B	GPT-oss 20B	Gemma 4 26B
No mask	0.776	0.904	0.937	0.914	0.781	0.766
Mask	0.762	0.914	0.937	0.915	0.900	0.856
Δ	−0.014	+0.010	0.000	+0.001	+0.119	+0.090

TPR@5%FPR is less uniform than AUROC: the best method varies by model: One-Class SVM on Ministral, FREQMASK-KM on LLaMA3, GMM on Qwen3 and GPT-oss, and LEARNEDMASK-LORA on Gemma. Meanwhile, LEARNEDMASK-LORA stays within a few points of the best on every $\geq$8B model. No single method dominates across the $\geq$8B models, consistent with different LLMs having different internal representation geometries that may favor different algorithmic choices. On Qwen 1.5B, all unsupervised methods score below 0.80 AUROC with low TPR@5%FPR, indicating that representation quality bounds what unsupervised detection can recover. We defer the structural reading of these results to Section 4.3, where the decomposition table licenses it directly.

4.3 Decomposition: Which Design Choices Most Impact the Performance?

Table 2 decomposes framework instantiations and baselines along representation space, clustering, sparsification, and scoring. The lower block isolates the role of the SAE embedding: *CVDD (SAE space)* is exactly CVDD with the SAE encoder substituted for the activation embedding and training. The three framework instantiations share KMeans clustering and nearest-centroid cluster assignment, and differ in how the sparse subspace is identified: globally by activation frequency (FREQMASK-KM), per-cluster by frequency (Local FREQMASK-KM), or per-cluster by a learned mask trained jointly with a LoRA adapter against SAE reconstruction (LEARNEDMASK-LORA).

Representation space carries a large share of the gain. *CVDD (SAE space)* differs from *CVDD* only in the encoder. Substituting the SAE encoder for the activation embedding lifts mean AUROC from 0.692 to 0.806. Adding a sparse local subspace on top lifts it further: FREQMASK-KM and LEARNEDMASK-LORA reach 0.904 and 0.909 respectively, on par with or above the global density baselines (Mahalanobis 0.893, GMM 0.903) while operating in a much lower effective dimension.

The mask is necessary for the largest models. We study the necessity of the mask by disabling it, scoring ℓ_1 over all SAE features, and re-sweeping the cluster count $\mathcal{N}_c$ for the no-mask condition alone, so it is given its own best case (Table 3; full sweep and setup in Appendix F.2). The masked configuration matches or exceeds the best no-mask result on all five $\geq$8B models, and the gap grows with model size. Larger models pack more concepts into a given activation, so the local subspace is harder to recover from the dense SAE representation and the mask contributes more.

Within the SAE space, the mechanism for finding the subspace does not dominate. FREQMASK-KM and LEARNEDMASK-LORA differ substantially in cost: global frequency selection is essentially free, whereas LEARNEDMASK-LORA requires fitting $\mathcal{N}_c$ masks and per-cluster low-rank adapters. Both are competitive against the global density baselines while operating in a lower effective dimension. Their agreement is at the aggregate level rather than model by model: FREQMASK-KM leads by

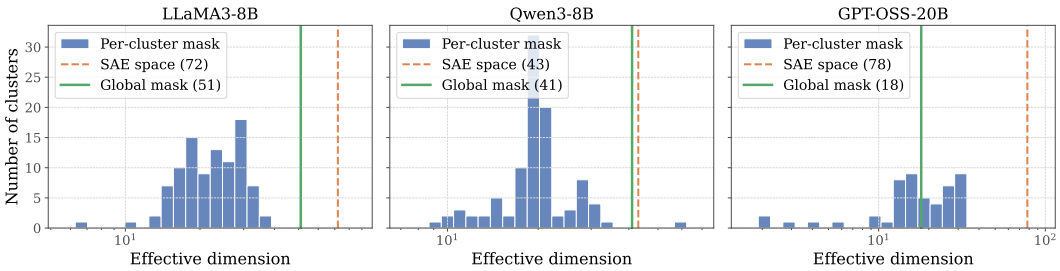

Figure 3: Effective dimension (d_{90}) across representation spaces for three of the model families; two more are shown in Appendix E (Figure 15), and Gemma-4-26B is omitted because its selected mask has only $k = 25$ features, which already bounds its masked d_{90} values by 25. Blue bars show the distribution of per-cluster d_{90} values, where each cluster's d_{90} is computed on that cluster's members within its own top-k mask $\mathcal{M}_c$ (the k SAE features that fire most often within the cluster, with k as for the global mask; Appendix E). Vertical lines mark d_{90} in the raw SAE feature space (dashed orange) and the global-mask subspace (solid green).

0.018 on Ministral and 0.025 on LLaMA3, the two are tied on Qwen3, and LEARNEDMASK-LORA leads by 0.019 on GPT-oss and 0.048 on Gemma (Table 1). Which construction wins is therefore model-dependent, but neither is far from the other on any ≥8B model. We read this as evidence for the structural claim: if local low effective dimension is the operative property, any reasonable procedure for surfacing that subspace should work. The comparability across instantiations supports the claim about structure rather than about any particular algorithm.

4.4 Local Effective Dimension Is Genuinely Low

The framework's feasibility and its robustness to design choice both rest on the premise that safe cluster members occupy low-dimensional subspaces of the SAE feature space. We test this by computing d_{90}, the smallest number of PCA components explaining 90% of variance, in three spaces: SAE space, the global-mask subspace spanned by the k features selected by FREQMASK-KM, and, for each cluster separately, a subspace of the same size k spanned by the features most frequently active within that cluster (the mask used by Local FREQMASK-KM). Definitions are deferred to Appendix E with extended results; Figure 3 shows the per-cluster distribution for three model families.

Across the three models, the per-cluster d_{90} is small in absolute terms (median 22 for LLaMA3-8B, 20 for Qwen3-8B, and 17.5 for GPT-OSS-20B), a tiny fraction of the ambient SAE dimension ($d_2 = 16{,}384$) and well below the raw SAE-space values of 72, 43, and 78 respectively. The per-cluster masks are not subsets of the global mask, so the three values describe different subspaces rather than successive restrictions of one another. Their relation is model-dependent: on LLaMA3-8B both masked subspaces are lower-dimensional than raw SAE space, with the per-cluster masks lowest (raw 72, global 51, per-cluster 22); on Qwen3-8B the raw and global-mask dimensions are already close (43 and 41), while the per-cluster masks give 20; and on GPT-OSS-20B the global mask alone brings the dimension from 78 to 18, and the per-cluster masks give a similar median (17.5). The two additional model families in Appendix E (Figure 15) follow the LLaMA3 pattern, with per-cluster medians of 36 (Qwen2-1.5B) and 40.5 (Ministral-8B). Across these five families the local effective dimension is consistently a small fraction of the ambient SAE dimension, which is what makes per-cluster scoring well posed.

5 Discussion and Limitations

Limits of the supervised assumption. Supervised methods work only insofar as the labels cover the unsafe distribution at deployment, and the evidence across safety subfields suggests this coverage is rarely complete [Andriushchenko et al., 2024, Mazeika et al., 2024]. The one-class framing trades a 0.04–0.08 AUROC gap to the supervised probe on the ≥8B models (Table 1) for independence from this assumption, which is a reasonable trade in regimes where it is unreliable.

Sensitivity to the safe distribution and calibration. Unsupervised detector assigns typicality score based on training representations of safe data. Consequently, benign regions of the input domain that are underrepresented during training may receive high anomaly scores regardless of their safety [Ruff et al., 2018]. Supervised detectors are also sensitive to such benign distribution shifts, while additionally relying on the unsafe training distribution remaining representative at deployment. The

Table 4: AUROC after per-cluster calibration fit from 1% of the unsafe data and 1% of the benign out-of-distribution data. Negatives are the held-out in-distribution safe points together with two novel benign domains absent from training (HellaSwag [Zellers et al., 2019] and HH-RLHF-helpful [Bai et al., 2022]); positives are the held-out unsafe points. In each column, the best result among the calibrated unsupervised methods is in **bold** and the second best is underlined. Means over 5 seeds. Methods that score on a local low-dimensional subspace obtain near-optimal performance.

Method	Qwen2-1.5B	Ministral-8B	LLaMA3-8B	Qwen3-8B	GPT-OSS-20B	Gemma-4-26B
FREQMASK-KM, uncalibrated	0.265	0.522	0.712	0.386	0.750	0.454
FREQMASK-KM	**0.986**	0.973	0.978	**0.985**	0.969	0.962
LEARNEDMASK-LORA	0.985	**0.990**	**0.987**	0.981	0.982	**0.976**
GMM	0.953	0.969	0.935	0.967	**0.985**	0.901
CVDD	0.890	0.920	0.960	0.964	0.789	0.317
BCE (supervised)	0.977	0.981	0.967	0.975	0.975	0.953

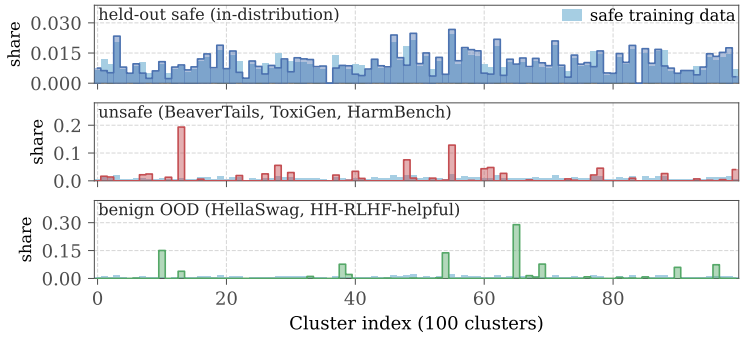

Figure 4: Data distribution over clusters across held-out safe, unsafe, and benign OOD data on LLaMA3-8B. Each panel overlays the safe training data (light blue) with one other data source. The held-out safe data follows the training occupancy closely, while unsafe data and novel benign data each concentrate in a few clusters and in different clusters from one another.

second requirement is arguably harder to satisfy, since unsafe inputs are open-ended and may evolve adversarially as new attack strategies emerge.

Importantly, the cluster structure learned from safe data makes it possible to correct the remaining sensitivity to benign shift using only a small labeled subset. This relies on empirical observation that unsafe data cluster assignment is highly non-homogeneous. That is, unsafe and novel benign data concentrate in small subsets of clusters, while safe data is spread more evenly (Figure 4). We employ this observation to design a simple correction method. Instead of modifying the learned representations (i.e., retraining), we adjust the scoring within each cluster via an affine transformation calibrated on a small labeled subset. In particular, with 1% of the unsafe data and 1% of the benign OOD data, AUROC rises from 0.265–0.750 to 0.962–0.990 across the six models, and benign false-positive rate displays substantial improvement from 0.12–0.92 to at most 0.041 (Tables 4 and 6). We note that this low-sample regime reflects a realistic scenario: while full unsafe and benign distributions are unavailable during deployment, it is feasible to obtain a small portion of each for calibration. We defer the remaining details and additional discussion to Appendix B.

6 Theory: Generalization under Local Sparsity

We study two questions separately: whether FREQMASK-KM estimates the safe geometry accurately, and whether this geometry retains the information needed to distinguish safe from unsafe inputs. The first question concerns estimation from safe data. The second also depends on the unsafe distribution and cannot be answered from local sparsity alone. We defer the full elaborated proof to Appendix I.

Population and empirical detectors. Let $\mathbb{P}_0$ and $\mathbb{P}_1$ be the distributions of safe and unsafe inputs. Fix the representation $z(x) := E(a^{(\ell)}(x)) \in \mathbb{R}_+^{d_2}$ and let P_y be the distribution of $z(X)$ for $X \sim \mathbb{P}_y$, $y \in \{0, 1\}$. We analyze an independent sample $\mathcal{D}_n = (Z_1, \ldots, Z_n) \sim P_0^n$, with K and k fixed. Thus, the result treats the SAE as given; it also applies conditionally on an SAE fitted using an

independent safe sample. Assume $\mathbb{E}_{P_0}\|Z\|_2^2 < \infty$, and fix population K-means centers

$$(\mu_1^\star, \dots, \mu_K^\star) \in \operatorname*{argmin}_{\mu_1,\dots,\mu_K} \mathbb{E}_{P_0} \min_{c\in[K]} \|Z - \mu_c\|_2^2. \tag{4}$$

Let $c^\star(z)$ be the nearest-center index, breaking ties by the smallest index, and let $\pi_c = P_0(c^\star(Z) = c)$. We assume $\pi_{\min} := \min_c \pi_c > 0$; then $\mu_c^\star = \mathbb{E}[Z \mid c^\star(Z) = c]$.

The frequency mask is *global*, as in Algorithm 1. Set $p_j = P_0(Z_j > 0)$, let $S^\star \subset [d_2]$ contain the k largest p_j, and put $M^\star = \operatorname{diag}(\mathbf{1}_{S^\star})$. For a target safe false-positive rate $q \in (0,1)$, define

$$s^\star(z) = \|M^\star(z - \mu^\star_{c^\star(z)})\|_1, \tag{5}$$

$$\tau_q = \inf\{t : P_0(s^\star(Z) \le t) \ge 1 - q\}, \qquad f_q^\star(z) = \mathbf{1}\{s^\star(z) > \tau_q\}. \tag{6}$$

This is the population one-class detector, not the optimal safe/unsafe classifier. Write $\widehat{\mu}_c$, $\widehat{c}_n$, and $\widehat{M}$ for the fitted centers, nearest-center assignment, and empirical global frequency mask. Their score is

$$\widehat{s}_n(z) = \|\widehat{M}(z - \widehat{\mu}_{\widehat{c}_n(z)})\|_1, \qquad \widehat{f}_{n,q}(z) = \mathbf{1}\{\widehat{s}_n(z) > \tau_q\}. \tag{7}$$

We first compare the scores at the same population threshold. Independent safe-quantile calibration is treated separately in Appendix I.5, as a deployment procedure rather than a claim about the experimental protocol.

Local geometry and clustering error. For each population cluster, let $J_c \subseteq S^\star$ be the coordinates that have nonzero conditional variance. Put $s_{\max} = \max_c |J_c| \le k$. This is a uniform clusterwise measure of local sparsity, distinct from the average neighborhood quantity s_{loc} in (3).

Assumption 1. *For every $c \in [K]$, conditionally on $c^\star(Z) = c$,*

$$\mathbb{E}\|M^\star(Z - \mu_c^\star)\|_2^2 \le v^2, \qquad \|M^\star(Z - \mu_c^\star)\|_2 \le B \quad \textit{almost surely}. \tag{8}$$

Here v^2 bounds the masked within-cluster covariance trace. Let $I_c = \{i : c^\star(Z_i) = c\}$, $N_c = |I_c|$, and let $\overline{\mu}_c$ be the mean over I_c, with value zero when $N_c = 0$. After matching empirical and population cluster labels, define

$$\rho_n = \max_c \|M^\star(\widehat{\mu}_c - \overline{\mu}_c)\|_1, \qquad \eta_{0,n} = P_0(\widehat{c}_n(Z) \ne c^\star(Z) \mid \mathcal{D}_n). \tag{9}$$

The matching is any fixed measurable rule, chosen from the training sample. These are clustering errors, not assumptions that the errors vanish: ρ_n measures the effect of fitting the partition on the centroids, and $\eta_{0,n}$ measures changed assignments of fresh safe points. The theorem keeps both terms explicit.

Assumption 2. *For $1 \le k < d_2$, the ordered frequencies satisfy $\gamma := p_{(k)} - p_{(k+1)} > 0$. Moreover, for some $L_0 < \infty$,*

$$P_0(|s^\star(Z) - \tau_q| \le t) \le L_0 t, \qquad t \ge 0. \tag{10}$$

The frequency gap permits recovery of the mask. The second condition limits the mass of safe points whose decisions can change after a small perturbation of the score.

Theorem 1. *Suppose Assumptions 1 and 2 hold. Let $\delta \in (0,1)$ and let*

$$n\pi_{\min} \ge 8 \log \frac{4K}{\delta}, \qquad n \ge 8\gamma^{-2} \log \frac{4d_2}{\delta} \qquad \varepsilon_n = \rho_n + \sqrt{\frac{2s_{\max}}{n\pi_{\min}}} \left(v + B\sqrt{2 \log \frac{4K}{\delta}} \right). \tag{11}$$

Then, with probability at least $1 - \delta$ over $\mathcal{D}_n$, it holds

$$P_0(\widehat{f}_{n,q}(Z) \ne f_q^\star(Z) \mid \mathcal{D}_n) \le \eta_{0,n} + L_0 \varepsilon_n. \tag{12}$$

Proof idea. Each population cluster contains at least $n\pi_c/2$ observations with high probability. Its masked sample mean concentrates in ℓ_2; converting to ℓ_1 costs only $\sqrt{s_{\max}}$ because the error is supported on J_c. Uniform frequency concentration identifies the global mask. Outside changed assignments, the scores then differ by at most ε_n, and (10) bounds the probability of a changed decision. For balanced clusters, the sampling term is of order $\widetilde{O}(\sqrt{Ks_{\max}/n})$ when $L_0 v, L_0 B$ are bounded. In particular, if $K, s_{\max} \lesssim s_{\text{loc}}$, this becomes $\widetilde{O}(s_{\text{loc}}/\sqrt{n})$. These are additional scaling conditions, not consequences of pointwise sparsity. Mask recovery has logarithmic dependence on d_2; the theorem does not bound SAE training or the dimension dependence of ρ_n and $\eta_{0,n}$.

Acknowledgment

Xin Chen is supported by the Open Philanthropy AI Fellowship and the Vitalik Buterin Fellowship from the Future of Life Institute. The research received further support through ELSA (European Lighthouse on Secure and Safe AI) funded by the European Union under grant agreement No. 101070617 and the Swiss National Science Foundation under NCCR Automation, grant agreement 51NF40 180545. Alexander Shevchenko was supported by the Swiss National Science Foundation (SNSF) Grant 204439. Gil Kur conducted part of the work during his visit to the IDEAL Institute, hosted by Lev Reyzin, which was supported by NSF ECCS-2217023.

References

Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K Arora, Yu Bai, Bowen Baker, Haiming Bao, et al. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*, 2025.

Charu C Aggarwal, Alexander Hinneburg, and Daniel A Keim. On the surprising behavior of distance metrics in high dimensional space. In *International conference on database theory*, pages 420–434. Springer, 2001.

Maksym Andriushchenko, Francesco Croce, and Nicolas Flammarion. Jailbreaking leading safety-aligned llms with simple adaptive attacks. *arXiv preprint arXiv:2404.02151*, 2024.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. https://transformer-circuits.pub/2023/monosemantic-features/index.html.

Boyuan Chen, Sitong Fang, Jiaming Ji, Yanxu Zhu, Pengcheng Wen, Jinzhou Wu, Yingshui Tan, Boren Zheng, Mengying Yuan, Wenqi Chen, et al. Ai deception: Risks, dynamics, and controls. *arXiv preprint arXiv:2511.22619*, 2025a.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Xin Chen, Yarden As, and Andreas Krause. Learning safety constraints for large language models. In *International Conference on Machine Learning*, pages 7664–7685. PMLR, 2025b.

Xin Chen, Jie Zhang, and Florian Tramer. Learning to inject: Automated prompt injection via reinforcement learning. *arXiv preprint arXiv:2602.05746*, 2026.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600*, 2023.

Josef Dai, Xuehai Pan, Ruiyang Sun, Jiaming Ji, Xinbo Xu, Mickel Liu, Yizhou Wang, and Yaodong Yang. Safe rlhf: Safe reinforcement learning from human feedback. *arXiv preprint arXiv:2310.12773*, 2023.

Xuefeng Du, Chaowei Xiao, and Yixuan Li. Haloscope: Harnessing unlabeled llm generations for hallucination detection. *Advances in Neural Information Processing Systems*, 37:102948–102972, 2024.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, et al. Toy models of superposition. *arXiv preprint arXiv:2209.10652*, 2022.

Dylan Feng, Pragya Srivastava, Anca Dragan, and Cassidy Laidlaw. Benchmarking anomaly detection for large language model alignment. In *Submitted to The Fourteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=Tr74UMzBdw.

Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. *arXiv preprint arXiv:2406.04093*, 2024.

Sachin Goyal, Aditi Raghunathan, Moksh Jain, Harsha Vardhan Simhadri, and Prateek Jain. Drocc: Deep robust one-class classification. In *International conference on machine learning*, pages 3711–3721. PMLR, 2020.

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.

Chuan Guo, Alexandre Sablayrolles, Hervé Jégou, and Douwe Kiela. Gradient-based adversarial attacks against text transformers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5747–5757, 2021.

Thomas Hartvigsen, Saadia Gabriel, Hamid Palangi, Maarten Sap, Dipankar Ray, and Ece Kamar. Toxigen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection. In *Proceedings of the 60th annual meeting of the association for computational linguistics (volume 1: Long papers)*, pages 3309–3326, 2022.

Zhengfu He, Wentao Shu, Xuyang Ge, Lingjie Chen, Junxuan Wang, Yunhua Zhou, Frances Liu, Qipeng Guo, Xuanjing Huang, Zuxuan Wu, et al. Llama scope: Extracting millions of features from llama-3.1-8b with sparse autoencoders. *arXiv preprint arXiv:2410.20526*, 2024.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

Zesheng Hong, Yubiao Yue, Yubin Chen, Lele Cong, Huanjie Lin, Yuanmei Luo, Mini Han Wang, Weidong Wang, Jialong Xu, Xiaoqi Yang, et al. Out-of-distribution detection in medical image analysis: A survey. *arXiv preprint arXiv:2404.18279*, 2024.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *Iclr*, 1(2):3, 2022.

Jonas Hübotter, Patrik Wolf, Alexander Shevchenko, Dennis Jüni, Andreas Krause, and Gil Kur. Specialization after generalization: Towards understanding test-time training in foundation models.

arXiv preprint arXiv:2509.24510, 2025.

Jiaming Ji, Mickel Liu, Josef Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. Beavertails: Towards improved safety alignment of llm via a human-preference dataset. *Advances in Neural Information Processing Systems*, 36:24678–24704, 2023.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547, 2019.

Subhash Kantamneni, Joshua Engels, Senthooran Rajamanoharan, Max Tegmark, and Neel Nanda. Are sparse autoencoders useful? a case study in sparse probing. *arXiv preprint arXiv:2502.16681*, 2025.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *arXiv preprint arXiv:2302.09664*, 2023.

Alexander H Liu, Kartik Khandelwal, Sandeep Subramanian, Victor Jouault, Abhinav Rastogi, Adrien Sadé, Alan Jeffares, Albert Jiang, Alexandre Cahill, Alexandre Gavaudan, et al. Ministral 3. *arXiv preprint arXiv:2601.08584*, 2026.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. Is your code generated by chatGPT really correct? rigorous evaluation of large language models for code generation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023a. URL https://openreview.net/forum?id=1qvx610Cu7.

Xiaogeng Liu, Nan Xu, Muhao Chen, and Chaowei Xiao. Autodan: Generating stealthy jailbreak prompts on aligned large language models. *arXiv preprint arXiv:2310.04451*, 2023b.

Mantas Mazeika, Long Phan, Xuwang Yin, Andy Zou, Zifan Wang, Norman Mu, Elham Sakhaee, Nathaniel Li, Steven Basart, Bo Li, et al. Harmbench: A standardized evaluation framework for automated red teaming and robust refusal. *arXiv preprint arXiv:2402.04249*, 2024.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

Patric Nader, Paul Honeine, and Pierre Beauseroy. Mahalanobis-based one-class classification. In *2014 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6. IEEE, 2014.

Milad Nasr, Nicholas Carlini, Chawin Sitawarin, Sander V Schulhoff, Jamie Hayes, Michael Ilie, Juliette Pluto, Shuang Song, Harsh Chaudhari, Ilia Shumailov, et al. The attacker moves second: Stronger adaptive attacks bypass defenses against llm jailbreaks and prompt injections. *arXiv preprint arXiv:2510.09023*, 2025.

Ori Nizan and Ayellet Tal. k-nnn: nearest neighbors of neighbors for anomaly detection. In *Proceedings of the IEEE/CVF Winter conference on applications of computer vision*, pages 1005–1014, 2024.

Changdae Oh, Zhen Fang, Shawn Im, Xuefeng Du, and Yixuan Li. Understanding multimodal llms under distribution shifts: An information-theoretic approach. In *International Conference on Machine Learning*, pages 46943–46970. PMLR, 2025.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. *arXiv preprint arXiv:2311.03658*, 2023.

Kiho Park, Yo Joong Choe, Yibo Jiang, and Victor Veitch. The geometry of categorical and hierarchical concepts in large language models. In *The Thirteenth International Conference on Learning Representations*, 2025a.

Seongheon Park, Xuefeng Du, Min-Hsuan Yeh, Haobo Wang, and Yixuan Li. Steer llm latents for hallucination detection. *arXiv preprint arXiv:2503.01917*, 2025b.

Gonçalo Paulo, Alex Mallen, Caden Juang, and Nora Belrose. Automatically interpreting millions of features in large language models. *arXiv preprint arXiv:2410.13928*, 2024.

Pramuditha Perera, Poojan Oza, and Vishal M Patel. One-class classification: A survey. *arXiv preprint arXiv:2101.03064*, 2021.

ReasoningTransferability. math_sft_40K. `https://huggingface.co/datasets/ReasoningTransferability/math_sft_40K`, 2025. HuggingFace dataset.

Jie Ren, Jiaming Luo, Yao Zhao, Kundan Krishna, Mohammad Saleh, Balaji Lakshminarayanan, and Peter J Liu. Out-of-distribution detection and selective generation for conditional language models. *arXiv preprint arXiv:2209.15558*, 2022.

Alexander Robey, Eric Wong, Hamed Hassani, and George J Pappas. Smoothllm: Defending large language models against jailbreaking attacks. *arXiv preprint arXiv:2310.03684*, 2023.

Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In *International conference on machine learning*, pages 4393–4402. PMLR, 2018.

Lukas Ruff, Yury Zemlyanskiy, Robert Vandermeulen, Thomas Schnake, and Marius Kloft. Self-attentive, multi-context one-class classification for unsupervised anomaly detection on text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4061–4071, 2019.

Bernhard Schölkopf, Robert C Williamson, Alex Smola, John Shawe-Taylor, and John Platt. Support vector method for novelty detection. *Advances in neural information processing systems*, 12, 1999.

Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, Amanda Askell, Samuel R Bowman, Newton Cheng, Esin Durmus, Zac Hatfield-Dodds, Scott R Johnston, et al. Towards understanding sycophancy in language models. *arXiv preprint arXiv:2310.13548*, 2023.

Mrinank Sharma, Meg Tong, Jesse Mu, Jerry Wei, Jorrit Kruthoff, Scott Goodfriend, Euan Ong, Alwin Peng, Raj Agarwal, Cem Anil, et al. Constitutional classifiers: Defending against universal jailbreaks across thousands of hours of red teaming. *arXiv preprint arXiv:2501.18837*, 2025.

Taylor Shin, Yasaman Razeghi, Robert L Logan IV, Eric Wallace, and Sameer Singh. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pages 4222–4235, 2020.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`, 2023.

Gemma Team. Gemma 4 technical report, 2026. URL `https://arxiv.org/abs/2607.02770`.

Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, Hoagy Cunningham, Nicholas L Turner, Callum McDougall, Monte MacDiarmid, C. Daniel Freeman, Theodore R. Sumers, Edward Rees, Joshua Batson, Adam Jermyn, Shan Carter, Chris Olah, and Tom Henighan. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. *Transformer Circuits Thread*, 2024. URL `https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html`.

Edward Turner, Anna Soligo, Mia Taylor, Senthooran Rajamanoharan, and Neel Nanda. Model organisms for emergent misalignment. *arXiv preprint arXiv:2506.11613*, 2025.

Akifumi Wachi, Thien Q Tran, Rei Sato, Takumi Tanabe, and Youhei Akimoto. Stepwise alignment for constrained language model policy optimization. *Advances in Neural Information Processing Systems*, 37:104471–104520, 2024.

Eric Wallace, Shi Feng, Nikhil Kandpal, Matt Gardner, and Sameer Singh. Universal adversarial triggers for attacking and analyzing nlp. In *Proceedings of the 2019 conference on empirical*

methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP), pages 2153–2162, 2019.

Yuxin Wen, Neel Jain, John Kirchenbauer, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Hard prompts made easy: Gradient-based discrete optimization for prompt tuning and discovery. *Advances in Neural Information Processing Systems*, 36:51008–51025, 2023.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Fan Yin, Jayanth Srinivasa, and Kai-Wei Chang. Characterizing truthfulness in large language model generations with local intrinsic dimension. *arXiv preprint arXiv:2402.18048*, 2024.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 4791–4800, 2019.

Shenyi Zhang, Yuchen Zhai, Keyan Guo, Hongxin Hu, Shengnan Guo, Zheng Fang, Lingchen Zhao, Chao Shen, Cong Wang, and Qian Wang. {JBShield}: Defending large language models from jailbreak attacks through activated concept analysis and manipulation. In *34th USENIX Security Symposium (USENIX Security 25)*, pages 8215–8234, 2025.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

Arthur Zimek, Erich Schubert, and Hans-Peter Kriegel. A survey on unsupervised outlier detection in high-dimensional numerical data. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 5(5):363–387, 2012.

Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J Zico Kolter, and Matt Fredrikson. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*, 2023.

Andy Zou, Long Phan, Justin Wang, Derek Duenas, Maxwell Lin, Maksym Andriushchenko, Rowan Wang, Zico Kolter, Matt Fredrikson, and Dan Hendrycks. Improving alignment and robustness with circuit breakers. *Advances in Neural Information Processing Systems*, 37:83345–83373, 2024.

A Broader Impacts

The motivation for a one-class framing is that supervised safety methods depend on having labels for the unsafe distribution at deployment, a condition that is rarely fully met in practice (Section 5). A detector that scores typicality with respect to safe behavior, rather than similarity to a fixed catalog of harms, can in principle flag previously unseen attack strategies and out-of-domain inputs without requiring those specific examples to be labeled in advance. To the extent the framework in this paper extends, it offers a route to safety methods that degrade more gracefully as adversarial techniques evolve, complementing rather than replacing existing supervised pipelines.

The main risk we see runs in the other direction. The sensitivity to the safe distribution discussed in Section 5 can produce false positives on legitimate benign inputs whose distribution differs from the safe training set, and an operator may respond by loosening the threshold, which simultaneously reduces detection of genuinely unsafe inputs. Operators relying on this kind of detector in isolation, without the calibration of Section 5 or a comparable safeguard against benign distribution shift, risk worse safety outcomes than a well-calibrated supervised classifier on a stable distribution. We explicitly do not advocate for unsupervised methods as standalone replacements for supervised safety systems.

B Per-Cluster Calibration from a Small Labeled Budget

This section details the recalibration procedure used in Section 5, gives the observation that motivates it, and reports the full results across all six models.

Motivation. The framework of Section 3 is fit without supervision, whereas the supervised probe of Section 4.2 requires enough labels to train a classifier. Many deployments fall between these two regimes: they can label a small sample, but not one large enough to characterize the unsafe distribution. We use such a sample to rescale the scores of an already-trained detector instead of training a classifier on it. The procedure reads only precomputed scores and cluster assignments, so the detector and its training are left untouched and no GPU is required.

The rescaling is fit per cluster because the clusters differ from one another in three respects, all measured on the deployed detector and shown in Figures 5 and 6 and Figure 4. First, the score is on a different scale in each cluster. On LLaMA3-8B the single global threshold that costs 5% false positives overall is higher than every safe score in 71 of the 98 clusters that contain at least ten safe points, and lower than the median safe score in 7 of them, so that one threshold flags no safe point in the first group and more than half of the safe points in the second. Second, the clusters differ in how well the within-cluster distance separates safe from unsafe points, from near-perfect separation to none. Third, unsafe points and benign out-of-distribution points are concentrated in different clusters. To make this precise, we count for each cluster how many unsafe evaluation points it contains, and then take clusters in decreasing order of that count until they account for half of all unsafe points. Between 2 and 9 clusters are enough, out of the 50 to 200 the model has, and a large fraction of the remaining clusters contain no unsafe point at all. Repeating the same count for the benign out-of-distribution points takes even fewer clusters, between 1 and 3. On every one of the six models, these two groups of clusters have no cluster in common. This describes where each kind of point is concentrated and not where it is allowed to occur: individual clusters, including some in the two groups, do contain both kinds of point. Figure 4 shows this per cluster on LLaMA3-8B, and Figures 7 to 11 repeat the same view on the other five: held-out safe data follows the training occupancy, while the two atypical groups each collapse onto a few clusters. Lowering the score in the clusters that hold most of the benign out-of-distribution points therefore need not lower it in the clusters that hold most of the unsafe points.

The rescaling has to vary across clusters to have any effect on the metrics we report. Both AUROC and recall at a fixed FPR depend on the scores only through the ranking they induce. A single global map $s \mapsto a_0 s + b_0$ with $a_0 > 0$ is strictly increasing, so it leaves that ranking, and therefore both metrics, unchanged. Coefficients that differ across clusters are what allow inputs assigned to different clusters to be reordered relative to one another. A global fit is still computed, but it enters the procedure as the shrinkage target (a_0, b_0) defined below, not as a calibration in its own right.

Setup. A one-class detector routes each input x to its nearest latent cluster $c(x)$ and produces a score $s(x)$, the (masked) distance to that cluster's centroid. At deployment a single global threshold on $s(x)$ yields the safe/unsafe decision.

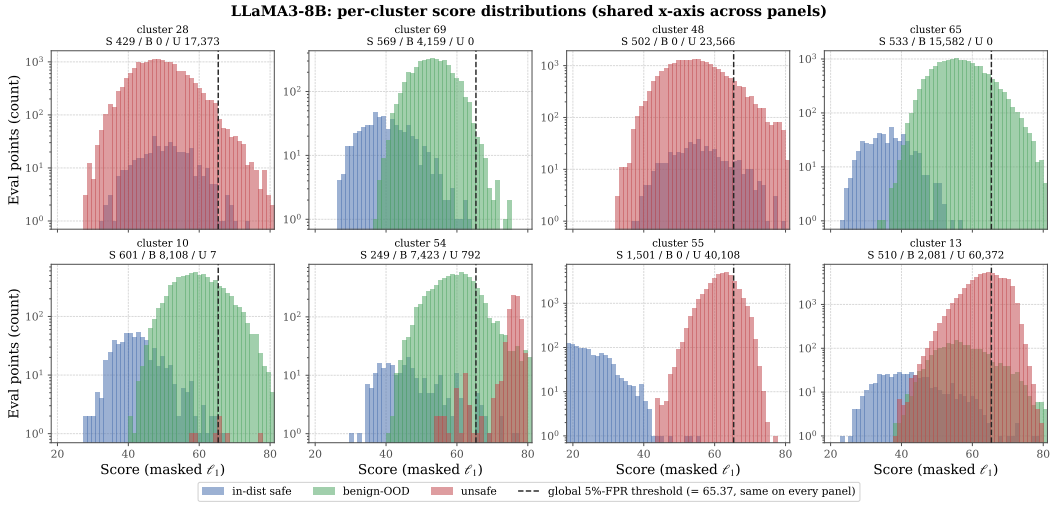

Figure 5: Per-cluster score distributions on LLaMA3-8B for the three groups: in-distribution safe, benign out-of-distribution, and unsafe. Counts are on a log axis, since the smaller group is invisible at linear scale (cluster 13 holds 60,372 unsafe points against 510 safe). All panels use the same x-range, so scores are directly comparable across panels. The dashed line is the *same* global threshold in every panel, the one costing 5% false positives over all in-distribution safe points. Its position within each cluster's score distribution differs: in cluster 55 it lies in the middle of the unsafe distribution, while in clusters 65 and 69 most benign out-of-distribution points fall below it. Clusters are selected by a fixed rule (the three largest by benign-OOD count, the three largest by unsafe count, and the two largest containing no unsafe point, deduplicated) and ordered by median score. The other five models behave similarly; Figure 6 reports the per-cluster score spread for all six.

Labeled sample. The calibration sample is drawn once per run and contains three groups: a fraction p of the unsafe inputs, the same fraction p of the benign out-of-distribution inputs, and half of the in-distribution safe inputs. The safe pool is not budget-limited because in-distribution safe data is abundant by assumption, whereas labeled unsafe examples and labeled examples of a novel benign domain are equally costly to obtain, so they are drawn at a common rate. We refer to p as the coupled budget; it is the horizontal axis of Figures 12 to 14. The inputs not drawn form the evaluation set, so the two are disjoint and no calibration point is scored at evaluation time.

Calibration map. We replace the raw score with a per-cluster affine map of it,

$$g(x) = a_{c(x)}\, s(x) + b_{c(x)}, \qquad \Pr(\text{unsafe} \mid x) = \sigma(g(x))\,, \qquad \sigma(z) = \frac{1}{1+e^{-z}}, \tag{13}$$

so that every cluster c has its own coefficient pair $(a_c, b_c) \in \mathbb{R}^2$. Benign out-of-distribution points enter the fit with the safe label. Scoring uses $g(x)$ with a single global threshold, and all reported metrics are threshold-free except the false-positive rate defined below.

Importance weights. The three groups are sampled at different rates: a fraction p of the unsafe points, the same fraction p of the benign out-of-distribution points, and one half of the in-distribution safe points. The calibration sample therefore has a different composition than the data the detector is applied to. At $p = 1\%$ it holds one labeled unsafe point for every 100 in the unsafe pool, but one labeled safe point for every 2 in the safe pool, so unsafe points are under-represented in it by a factor of 50 relative to safe points. An unweighted fit would place the coefficients, and in particular the offsets b_c, where they suit that composition rather than the full data. We therefore weight each point by the reciprocal of its group's sampling fraction,

$$w_i = \begin{cases} 1/p & \text{if } i \text{ is unsafe,} \\ 1/p & \text{if } i \text{ is benign out-of-distribution,} \\ 2 & \text{if } i \text{ is in-distribution safe,} \end{cases} \tag{14}$$

so that each labeled point counts once for every point of its group that it stands for. The weight 2 on in-distribution safe points is that same rule applied to a sampling fraction of one half: each

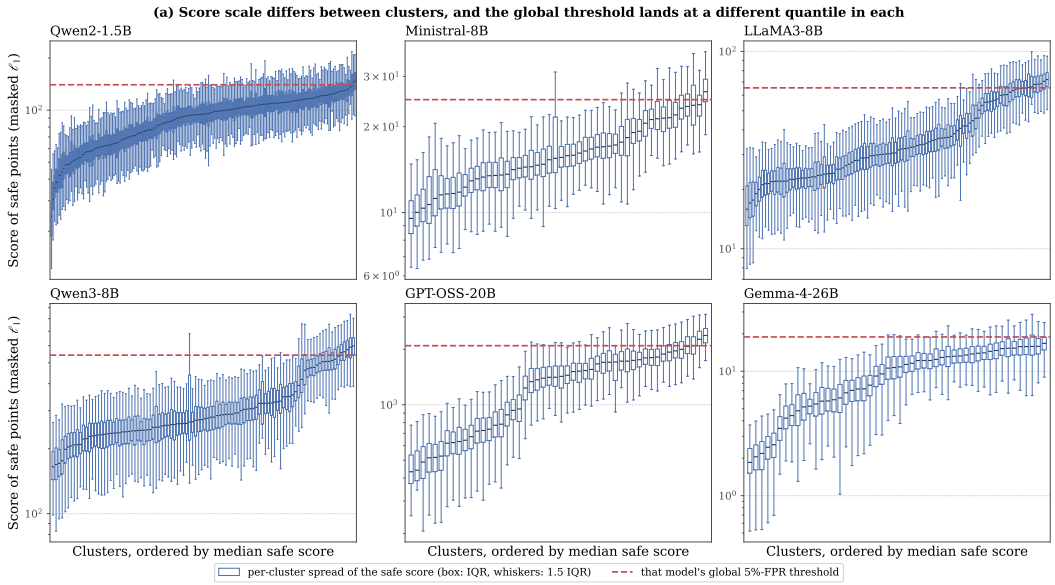

Figure 6: Per-cluster spread of the score over in-distribution safe points, for all six models, clusters ordered by median, with each model's own global 5%-FPR threshold as a dashed line. The per-cluster medians span roughly an order of magnitude within every model, and the thresholds themselves differ by two orders of magnitude across models (18.9 on Gemma-4-26B to 2094 on GPT-OSS-20B), so no score scale is shared either within or across models.



Figure 7: Per-cluster occupancy on Qwen2-1.5B (200 clusters). Axes, colors and panel order are as in Figure 4. Total variation distance from the training occupancy is 0.11 for held-out safe, 0.72 for unsafe and 0.86 for benign out-of-distribution.

labeled safe point stands for itself and for one safe point that was not sampled. With these weights the weighted group sizes in the calibration sample equal the true group sizes, and the weighted sum in the objectives below is an unbiased estimate of the same sum taken over the full data.

Global anchor. The labeled sample is small and is divided over 50 to 200 clusters, so many clusters receive few labeled points and some receive none. Fitting every cluster independently on its own points is therefore not possible: a cluster whose labeled points are all safe or all unsafe is linearly separable and has no finite logistic fit, and a cluster with no labeled point has nothing to fit at all. We introduce (a_0, b_0) as the estimate to fall back on in these cases. It is obtained by weighted logistic regression over *all* labeled points pooled together, ignoring cluster membership:

$$(a_0, b_0) = \operatorname*{argmin}_{a,b} \sum_i w_i \, \ell(y_i,\, a s_i + b) + \varepsilon \left(a^2 + b^2\right), \tag{15}$$

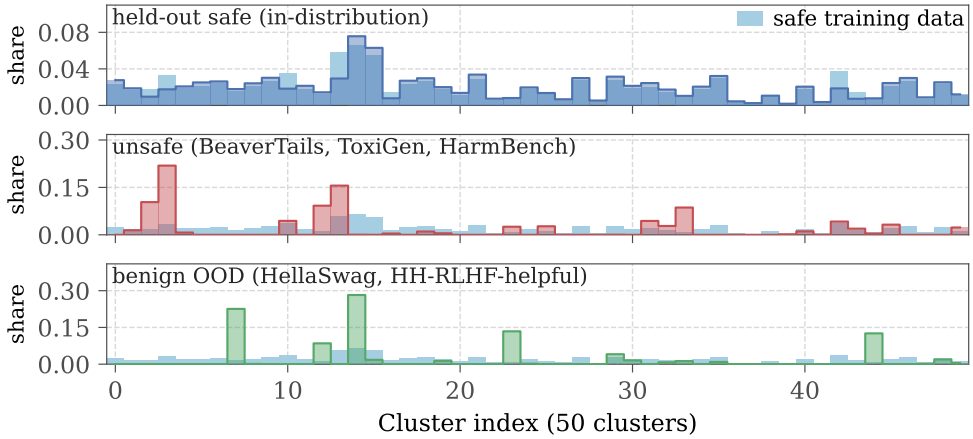

Figure 8: Per-cluster occupancy on Ministral-8B (50 clusters). Axes, colors and panel order are as in Figure 4. Total variation distance from the training occupancy is 0.10 for held-out safe, 0.64 for unsafe and 0.76 for benign out-of-distribution.

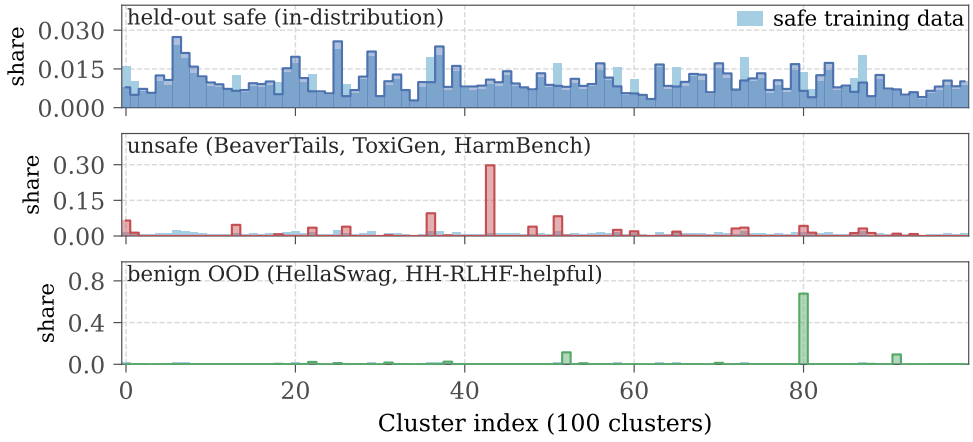

Figure 9: Per-cluster occupancy on Qwen3-8B (100 clusters). Axes, colors and panel order are as in Figure 4. Total variation distance from the training occupancy is 0.11 for held-out safe, 0.73 for unsafe and 0.90 for benign out-of-distribution.

where $y_i \in \{0, 1\}$ labels input i as safe (including benign-OOD) or unsafe and ℓ is the logistic loss,

$$\ell(y, z) = -\Big[\, y \log \sigma(z) + (1 - y) \log\big(1 - \sigma(z)\big)\Big]. \tag{16}$$

The term $\varepsilon\,(a^2 + b^2)$ with $\varepsilon = 10^{-3}$ is a small ridge that keeps the minimizer finite when the labeled sample is close to linearly separable.

Two properties make this pair a suitable fallback. It is estimated from every labeled point at once, so it remains well determined at budgets at which individual clusters hold almost no labeled data. It is also the best affine map available to a procedure that may not use cluster membership, so taking it as the target of the penalty in the next paragraph states the assumption that a cluster behaves like the pooled data unless its own labeled points show otherwise. Shrinking toward $(0, 0)$ instead, which is what a plain ridge penalty would do, would drive a_c and b_c toward zero and discard the score entirely in exactly those clusters that hold too little labeled data to estimate anything. Note that (a_0, b_0) is never used to score inputs on its own: as shown above, a single global affine map leaves AUROC and recall unchanged.



Figure 10: Per-cluster occupancy on GPT-OSS-20B (50 clusters). Axes, colors and panel order are as in Figure 4. Total variation distance from the training occupancy is 0.17 for held-out safe, 0.67 for unsafe and 0.78 for benign out-of-distribution.



Figure 11: Per-cluster occupancy on Gemma-4-26B (50 clusters). Axes, colors and panel order are as in Figure 4. Total variation distance from the training occupancy is 0.17 for held-out safe, 0.62 for unsafe and 0.87 for benign out-of-distribution.

Per-cluster estimation. Each cluster then minimizes the same weighted loss, restricted to the points assigned to that cluster, with an ℓ_2 penalty pulling its coefficients toward the anchor,

$$(a_c, b_c) = \operatorname*{argmin}_{a,b} \sum_{i\,:\,c(x_i)=c} w_i\, \ell(y_i,\, a s_i + b) + \lambda \,\|(a, b) - (a_0, b_0)\|_2^2 , \tag{17}$$

initialized at (a_0, b_0). The likelihood term is a sum over the labeled points assigned to cluster c while λ is fixed, so the penalty dominates the objective for clusters holding few labeled points and the likelihood dominates for clusters holding many; a cluster with no labeled points retains the anchor exactly. The estimator therefore interpolates between the global map and an unconstrained per-cluster fit as a function of how many labeled points each cluster receives. We select $\lambda \in \{1, 10, 100\}$ by cross-validation within the labeled sample, scoring folds by AUROC. Note that a negative a_c is permitted, which is what allows the procedure to repair clusters in which the raw score ranks unsafe points as *closer* to the centroid than safe ones; such clusters exist on every model (Figure 6).

Evaluation protocol. The test pool is fixed and always contains all three groups, so the numbers before and after calibration are computed on identical data. Negatives are the held-out in-distribution safe points together with the held-out benign out-of-distribution points (HellaSwag, 10,042 examples, and HH-RLHF-helpful, 43,835 examples); positives are the held-out unsafe pool. A detector that flags a novel benign domain is therefore penalized by this metric. Two quantities are reported. AUROC

Table 5: Recall at the threshold costing 5% false positives on the in-distribution safe points, at the coupled budget $p = 1\%$. Higher is better. The first row is the deployed detector before calibration.

Method	Qwen2-1.5B	Ministral-8B	LLaMA3-8B	Qwen3-8B	GPT-OSS-20B	Gemma-4-26B
FREQMASK-KM, uncalibrated	0.000	0.051	0.137	0.000	0.295	0.004
FREQMASK-KM	0.970	0.840	0.913	0.978	0.777	0.681
LEARNEDMASK-LORA	0.957	0.977	0.981	0.956	0.902	0.837
GMM	0.531	0.882	0.743	0.819	0.894	0.402
CVDD	0.620	0.527	0.638	0.783	0.117	0.002
BCE (supervised)	0.910	0.947	0.905	0.926	0.911	0.800

Table 6: False-positive rate on the held-out benign out-of-distribution domains, measured at the threshold costing 5% false positives on the in-distribution safe points, at the coupled budget $p = 1\%$. Lower is better. The first row is the deployed detector before calibration. Means over 5 seeds.

Method	Qwen2-1.5B	Ministral-8B	LLaMA3-8B	Qwen3-8B	GPT-OSS-20B	Gemma-4-26B
FREQMASK-KM, uncalibrated	0.920	0.730	0.155	0.832	0.123	0.622
FREQMASK-KM	0.002	0.012	0.015	0.003	0.041	0.026
LEARNEDMASK-LORA	0.000	0.002	0.005	0.007	0.007	0.004
GMM	0.068	0.001	0.011	0.010	0.002	0.009
CVDD	0.001	0.014	0.038	0.014	0.000	0.831
BCE (supervised)	0.070	0.018	0.045	0.032	0.017	0.035

uses this pooled negative set, so it is a strictly harder quantity than the AUROC of Table 1, whose negatives are in-distribution safe only; the two are not comparable. The benign false-positive rate is the fraction of held-out benign-OOD points scored above the threshold that costs 5% false positives on held-out in-distribution safe points alone, which is the same operating point at which recall is reported, so recall and the benign false-positive rate are computed at the same threshold. All numbers are means over five seeds.

Results across models. Figures 12 to 14 report every method on all six models as a function of the coupled budget. At the largest budget we measure ($p = 0.02$), per-cluster calibration raises AUROC from 0.267–0.750 to 0.962–0.987 for FREQMASK-KM, and from 0.308–0.476 to 0.978–0.992 for LEARNEDMASK-LORA. Unlike the unsafe-only budget studied in earlier versions of this experiment, the gain here is monotone in the budget at the low end: even at the smallest budget we measure ($p = 3 \times 10^{-5}$, of the order of ten labeled examples of each kind) the calibrated score is above the uncalibrated one on every model and both detectors, reaching 0.730–0.892 for FREQMASK-KM and 0.645–0.824 for LEARNEDMASK-LORA. The reason is that the uncalibrated detector is very poor on this test pool to begin with, so even a coarse per-cluster offset estimated from a handful of points is an improvement, whereas against in-distribution negatives alone the uncalibrated detector is already strong and a noisy calibration can reduce AUROC at the smallest budgets.

Table 5 reports recall at the 5% operating point, the third quantity reported for this experiment, alongside the AUROC of Section 5.

C Related Work

C.1 LLM Safety

The term "safety" is used across several research communities to refer to related but distinct concerns. The concerns range from *harmful content generation*, addressed by RLHF-based alignment [Dai et al., 2023, Wachi et al., 2024] and content-filtering systems [Sharma et al., 2025]; to *adversarial robustness* against jailbreaks and prompt injections [Mazeika et al., 2024, Chen et al., 2026]; to *alignment properties* such as deception [Chen et al., 2025a] and sycophancy [Sharma et al., 2023]; and to *application-domain boundaries*, where an input outside a deployed system's intended domain is treated as unsafe even when it is not harmful in absolute terms [Hong et al., 2024].

Each of these definitions has developed its own methodology, and in each case, the dominant paradigm is supervised or otherwise commits to a specific hypothesis about what unsafe inputs look like, whether through labeled training data [Park et al., 2025b], identified concept directions [Zhang et al., 2025], or assumed brittleness under perturbation [Robey et al., 2023]. Our contribution is to

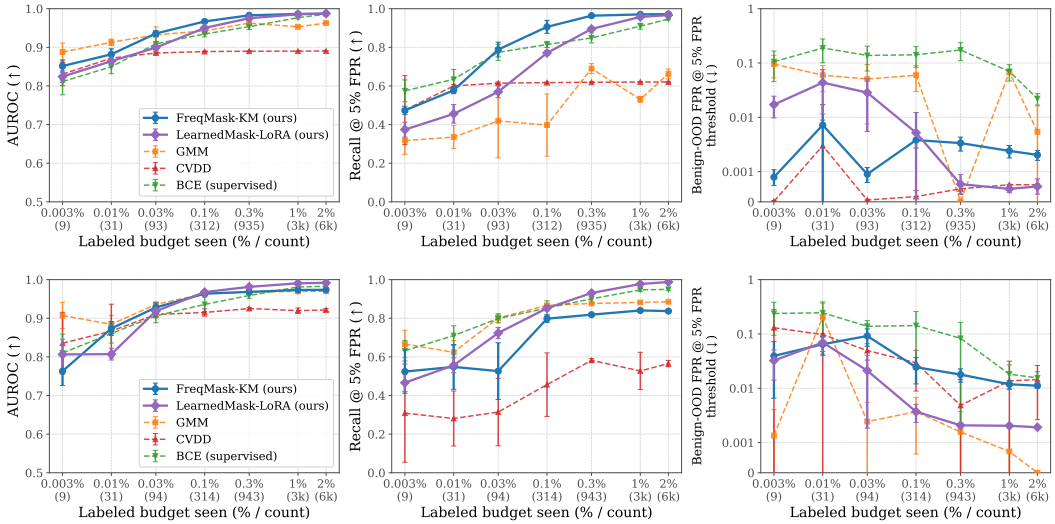

Figure 12: Calibration against the coupled budget for Qwen2-1.5B (top) and Ministral-8B (bottom). AUROC (left, negatives are in-distribution safe together with the benign domains), recall at 5% FPR (middle), and benign out-of-distribution FPR at that same threshold (right, lower is better, symmetric-log axis so exact zeros remain visible). Our methods are solid, baselines dashed. Markers are means over 5 seeds and bars are ±std; at the smallest budgets the sample is of the order of ten labeled examples, so several baselines have a spread comparable to their mean.

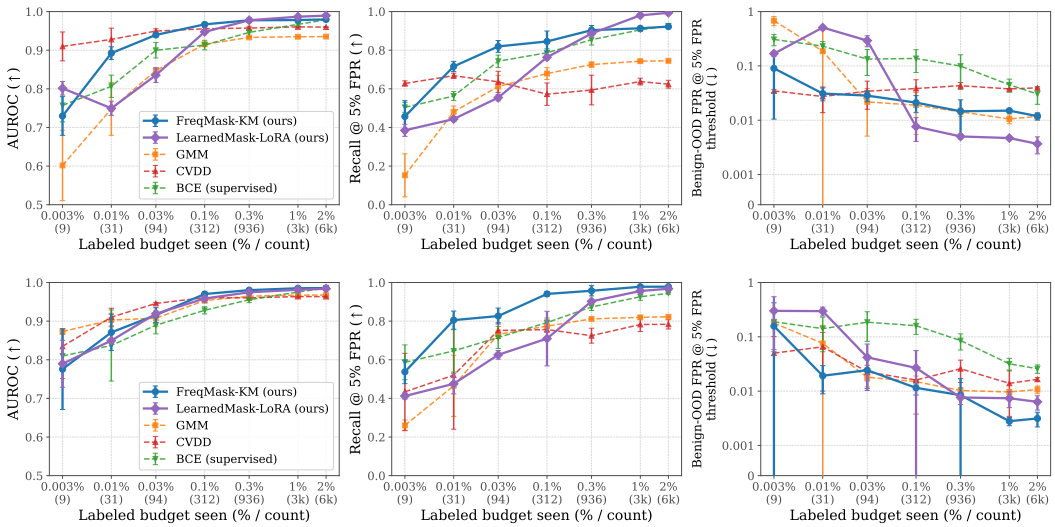

Figure 13: Calibration against the coupled budget for LLaMA3-8B (top) and Qwen3-8B (bottom). Axes, markers, and line styles are as in Figure 12.

show that the same structure used by representation-based safety methods [Chen et al., 2025b, Zou et al., 2024] is exploitable under a framing that does not commit to any single definition of safety.

C.2 One-Class Anomaly Detection and Discussions in LLM

One-class anomaly detection addresses a different problem from classification: Given samples from a distribution p_{safe}, the task is to score a new input by how typical it is under p_{safe}, without reference to any unsafe distribution. Classical approaches fall into different categories, such as density estimation (Gaussian mixtures [Nader et al., 2014], kernel density estimation [Goyal et al., 2020]) and distance-based scoring (k-nearest-neighbors [Nizan and Tal, 2024], variance reduction [Ruff et al., 2018,

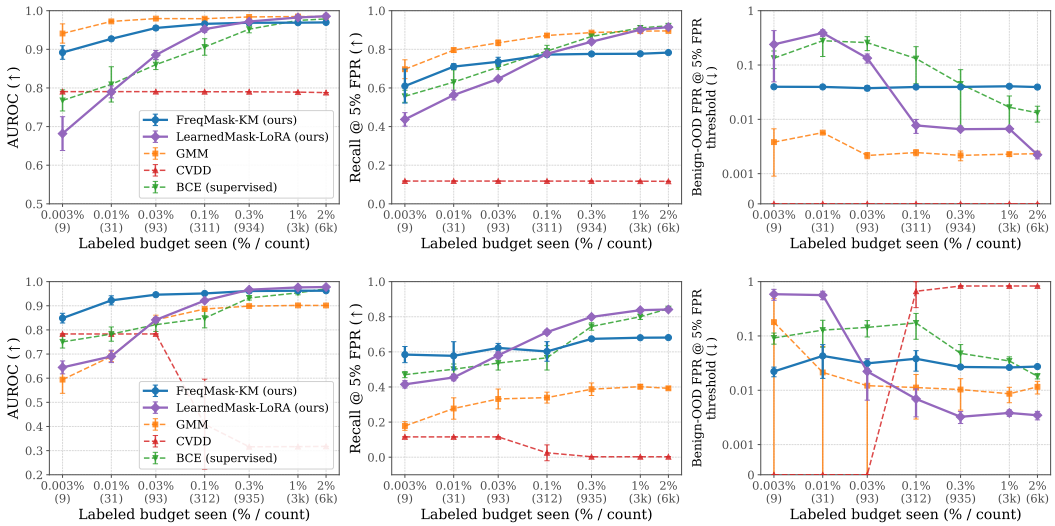

Figure 14: Calibration against the coupled budget for GPT-OSS-20B (top) and Gemma-4-26B (bottom). Axes, markers, and line styles are as in Figure 12. The CVDD curve on Gemma-4-26B falls below the AUROC range of the other methods once the budget exceeds 0.03%: its raw score is essentially uninformative on that model, so the calibration has no signal to sharpen.

2019]). Applied to LLM safety, this framing has a natural appeal: an unsafe input is one that is distributionally atypical with respect to safe behaviors, regardless of the harm category.

An existing line of work in LLMs proposes scoring functions without labeled negative data, but targets hallucination and reliability rather than safety, and it is not a priori clear that scores tuned for the former transfer to the latter. Examples include Gaussian models on embedding distances for OOD detection [Ren et al., 2022], semantic entropy over sampled outputs for uncertainty [Kuhn et al., 2023], intrinsic-dimension estimators on activation neighborhoods for truthfulness [Yin et al., 2024], and identifying hallucination subspaces from unlabeled generations [Du et al., 2024]. Feng et al. [2025] takes a first step toward bridging these literatures with safety by introducing an anomaly detection benchmark for LLM misalignment. However, it does not propose new algorithms or frameworks. By far, no existing work provides theoretical insight into what makes anomaly detection tractable in high-dimensional LLM representations, which is the question our work seeks to answer.

C.3 SAE Features for Downstream Tasks

Kantamneni et al. [2025] study whether SAE features are a useful representation for supervised binary probing. Their probe uses labels on two classes and selects a single global set of features from class-conditioned differences. They find that SAE probes rarely outperform probes trained on the dense activations, except in specific regimes such as data scarcity, class imbalance, label noise, and covariate shift. Our setting is complementary. We use no unsafe labels at any point, so features cannot be selected from differences between the two classes; the mask in FREQMASK-KM is chosen by activation frequency on safe data alone. Our claim is also different. We do not claim that SAE features are a good representation. We claim that the sparse support is *local*, in the sense that it varies across neighborhoods of the concept space, and that this locality is what makes anomaly detection feasible in a $16{,}384$-dimensional feature space, both in the algorithms of Section 3 and in the sample-complexity bound of Section 6, which scales only logarithmically in d_2. On the representation question, our unsupervised setting differs from their supervised one: moving to the SAE space provides a large gain, raising CVDD from 0.692 to 0.806 mean AUROC on the $\geq$8B models (Table 2). Masking provides substantial additional gains on the largest models, adding 0.119 AUROC on GPT-oss-20B and 0.090 on Gemma-4-26B, while changing AUROC by at most 0.014 on the other four models.

D Experimental Setup: Additional Details

This section provides extended details on the SAE training procedure (Appendix D.1), datasets (Appendix D.2), per-algorithm hyperparameters (Appendix D.4), and compute budget (Appendix D.5).

D.1 SAE Training

For each model, we train a Sparse Autoencoder on residual-stream activations extracted at the analysis layer specified in Section 4.1 (layer 20 for Qwen 1.5B and Ministral, layer 25 for LLaMA3, layer 30 for Qwen3, layer 20 for GPT-oss, and layer 25 for Gemma-4-26B). Training data is the same pooled safe distribution used by the downstream anomaly-detection methods, comprising approximately 206K last-token activations per model. SAEs are trained independently per model; no parameters or features are shared across models. For GPT-oss we train several SAEs at different sparsity weights (Table 7), as the SAE-space methods select different operating points along the sparsity spectrum.

The SAE follows a standard architecture mapping a residual-stream activation $a \in \mathbb{R}^{d_{\text{model}}}$ to a sparse feature vector $z = E(a) = \text{ReLU}(W_E a + b_E) \in \mathbb{R}^{d_{\text{SAE}}}$ and back via $\hat{a} = D(z) = W_D z + b_D$, with a single hidden layer of size $d_{\text{SAE}} = 16384$. Training minimizes a reconstruction loss with an ℓ_1 sparsity penalty on the hidden activations:

$$\mathcal{L}_{\text{rec}}(a) = \|a - D(E(a))\|_2^2 + \lambda_{\ell_1} \|E(a)\|_1. \tag{18}$$

The value of λ_{ℓ_1} is not comparable across models, because the penalty is applied to activations of different magnitude. We measure that magnitude by the root mean square (RMS) of a residual-stream activation, $\text{RMS}(a) = \left(\frac{1}{d_{\text{model}}} \sum_{j=1}^{d_{\text{model}}} a_j^2\right)^{1/2}$, which differs across the six models by up to a factor of about 34. Since the reconstruction term in $\mathcal{L}_{\text{rec}}$ is quadratic in the scale of a while the penalty term is only linear in it, the penalty carries relatively less weight the larger that scale is, so a model with larger RMS needs a larger λ_{ℓ_1} to reach the same feature density. Training hyperparameters are summarized in Table 7.

Table 7: SAE training hyperparameters. Input dimension is the residual-stream width of the corresponding base model; the SAE feature dimension is shared across all models. The sparsity weight λ_{ℓ_1} is shared across the original three models; GPT-oss and Gemma require larger values because their residual activations have larger RMS ($\approx 34\times$ for GPT-oss). For GPT-oss we train at two operating points ($\lambda_{\ell_1} = 300$ and 700), the $\lambda_{\ell_1} = 700$ encoder being the one used by FREQMASK-KM; Gemma uses a single intermediate value ($\lambda_{\ell_1} = 3$).

	Qwen 1.5B	Ministral 8B	LLaMA3 8B	Qwen3 8B	GPT-oss 20B	Gemma 4 26B
Input dimension d_{model}	1536	4096	4096	4096	2880	2816
SAE feature dimension d_{SAE}	16384	16384	16384	16384	16384	16384
Expansion factor	$\approx 10.7\times$	$4\times$	$4\times$	$4\times$	$\approx 5.7\times$	$\approx 5.8\times$
Sparsity weight λ_{ℓ_1}	0.01	0.01	0.01	0.01	300 / 700	3
Epochs	10	5	5	20	20	10
Batch size	256	256	256	256	256	256
Optimizer	Adam	Adam	Adam	Adam	Adam	Adam
Learning rate	1e-3	1e-3	1e-3	1e-3	1e-3	1e-3

D.2 Datasets

Safe distribution. The safe training set pools two groups of data: (i) general capability and instruction-following data (GSM8K, MathSFT40K, HumanEval, HumanEval+, MBPP, Alpaca, MT-Bench, MMLU, ARC), and (ii) the safe portions of the evaluation benchmarks (benign examples from BeaverTails and ToxiGen, and the model's correct refusals on HarmBench attacks). The total is approximately 206,000 last-token activations per model. Counts are approximately the same across models; minor variation arises from generation failures on individual prompts.

Unsafe evaluation. We evaluate on three categories: BeaverTails (harmful-content requests), ToxiGen (toxic and hateful language), and pooled HarmBench adversarial attacks generated by eight algorithms (AutoDAN [Liu et al., 2023b], AutoPrompt [Shin et al., 2020], DirectRequest [Mazeika et al., 2024], GBDA [Guo et al., 2021], GCG [Zou et al., 2023], HumanJailbreaks [Mazeika et al., 2024], PEZ [Wen et al., 2023], UAT [Wallace et al., 2019]). HarmBench suffixes are optimized per-target-model, so the adversarial inputs differ across models; the eight attacks are pooled into a single category for reporting. The total unsafe evaluation set comprises approximately 314,000 samples per model.

Splits. The safe set is split $0.8/0.2$: the 0.8 portion is used to fit each method (cluster centroids, mask selection, covariance estimation, etc.), and the remaining 0.2 is held out from fitting. Both AUROC and TPR@5%FPR are computed from the same ROC curve on the evaluation set, where

Table 8: Licenses for datasets used in this work. All datasets are publicly available and cited at first mention in Section 4.1. For the three benchmarks in the lower block, the safe portion contributes to the safe training pool and the unsafe portion is used only for evaluation.

Dataset	License	Use
GSM8K [Cobbe et al., 2021]	MIT	Safe training
MathSFT40K [ReasoningTransferability, 2025]	Not stated	Safe training
HumanEval [Chen et al., 2021]	MIT	Safe training
HumanEval+ (EvalPlus) [Liu et al., 2023a]	Apache 2.0	Safe training
MBPP [Austin et al., 2021]	CC-BY 4.0	Safe training
Alpaca [Taori et al., 2023]	CC-BY-NC 4.0	Safe training
MT-Bench [Zheng et al., 2023]	Apache 2.0	Safe training
MMLU [Hendrycks et al., 2020]	MIT	Safe training
ARC [Clark et al., 2018]	CC-BY-SA 4.0	Safe training
BeaverTails [Ji et al., 2023]	CC-BY-NC 4.0	Safe training & unsafe eval
ToxiGen [Hartvigsen et al., 2022]	MIT	Safe training & unsafe eval
HarmBench [Mazeika et al., 2024]	MIT	Safe training & unsafe eval

the safe evaluation samples supply the negatives and each unsafe category supplies the positives; TPR@5%FPR is the true-positive rate at the operating point whose false-positive rate equals 5%. No separate threshold-calibration step is performed.

D.3 Activation-space baselines

All activation-space baselines operate directly on the residual-stream activations extracted at the analysis layer (Section 4.1), without any SAE encoding. We use our own implementations to keep the training and scoring pipelines uniform across methods. The hyperparameters that were swept and the values selected for the main results are listed in Tables 9, 10, 11, 12, 13, and 14.

Mahalanobis. A single Gaussian is fit to the safe training set: the global mean μ is estimated on safe samples, and the covariance Σ is accumulated on the full training pool, regularized by adding λI before inversion. The anomaly score is the Mahalanobis distance to the global mean, $\sqrt{(x-\mu)^\top \Sigma^{-1}(x-\mu)}$, mapped to $[0,1]$ either by linear interpolation between the 5th and 95th percentiles of training distances (*percentile*) or by a sigmoid on standardized distances (*sigmoid*). The two free choices are the regularization strength λ and the score type.

Gaussian Mixture Model (GMM). A diagonal-covariance Gaussian mixture is fit on standardized safe features by EM, with components initialized via k-means++ or uniformly at random. The anomaly score is the negative log-likelihood $-\log p(x \mid \theta)$, normalized to $[0,1]$ by the same percentile or sigmoid mapping used for Mahalanobis. The free choices are the number of mixture components $n_{\text{components}}$, the initialization, and the score normalization.

One-Class SVM (OC-SVM). We use a PyTorch implementation that solves the standard one-class SVM dual on standardized safe features, with ν controlling the upper bound on the training-error fraction (and equivalently the lower bound on the support-vector fraction), and γ controlling the RBF bandwidth. The anomaly score is the negative decision function (distance from the separating hyperplane in feature space). The free choices are ν, γ, and the kernel (RBF or linear).

Deep SVDD. Following Ruff et al. [2018], a small MLP encoder $\phi : \mathbb{R}^{d_{\text{model}}} \to \mathbb{R}^{d_{\text{rep}}}$ is autoencoder-pretrained and then fine-tuned to map safe inputs near a fixed center c in representation space, by minimizing the mean squared distance $\|\phi(x) - c\|_2^2$. The anomaly score is this squared distance. The free choices are the representation dimension d_{rep}, the deep one-class hyperparameter ν (used by the soft-boundary objective), and the optimizer learning rate.

Context Vector Data Description (CVDD). CVDD generalizes Deep SVDD to multiple centers: an MLP encoder is trained jointly with n_{contexts} context vectors $\{c_k\}_{k=1}^{n_{\text{contexts}}}$, with the per-sample loss given by a temperature-annealed soft-minimum over distances to all centers, $-T \log \sum_k \exp(-\|\phi(x) - c_k\|_2^2/T)$. An orthogonality regularizer $\|CC^\top - I\|_F^2$ encourages the centers to span distinct directions. The anomaly score is the distance to the nearest context vector. The free choices are the number of contexts n_{contexts} and the orthogonality weight; the temperature schedule and pretraining flag are fixed across runs.

D.4 Hyperparameters

For each method we list the hyperparameters that were swept and the values selected for the main results. Selection is performed by AUROC.

D.4.1 Activation-space baselines

Hyperparameters are tuned per model rather than shared, since the analysis layer, residual-stream geometry, and sample sizes vary across models. Tables 9, 10, 11, 12, 13, and 14 report swept ranges and selected values for each model.

Table 9: Hyperparameter sweeps and selected values for activation-space baselines on Qwen 1.5B.

Method	Hyperparameter	Sweep range	Selected
Mahalanobis	Regularization	$\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$	10^{-5}
	Score type	{percentile, sigmoid}	sigmoid
GMM	$n_{\text{components}}$	$\{3, 5, 10, 50, 100\}$	50
	Initialization	{k-means++, random}	k-means++
	Score normalization	{percentile, sigmoid}	percentile
OC-SVM	ν	$\{0.01, 0.05, 0.1, 0.2\}$	0.01
	γ	$\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$	10^{-2}
	Kernel	{RBF, linear}	RBF
Deep SVDD	Representation dim.	$\{16, 32, 64, 128\}$	128
	ν	$\{0.05, 0.1, 0.2\}$	0.2
	Learning rate	$\{10^{-4}, 10^{-3}, 10^{-2}\}$	10^{-3}
CVDD	n_{contexts}	$\{10, 50, 200, 500\}$	500
	Orthogonality weight	$\{10^{-4}, 10^{-3}, 10^{-2}\}$	10^{-3}

Table 10: Hyperparameter sweeps and selected values for activation-space baselines on Ministral 8B.

Method	Hyperparameter	Sweep range	Selected
Mahalanobis	Regularization	$\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$	10^{-5}
	Score type	{percentile, sigmoid}	percentile
GMM	$n_{\text{components}}$	$\{3, 5, 10, 50, 100\}$	10
	Initialization	{k-means++, random}	k-means++
	Score normalization	{percentile, sigmoid}	sigmoid
OC-SVM	ν	$\{0.01, 0.05, 0.1, 0.2\}$	0.1
	γ	$\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$	10^{-3}
	Kernel	{RBF, linear}	RBF
Deep SVDD	Representation dim.	$\{16, 32, 64, 128\}$	128
	ν	$\{0.05, 0.1, 0.2\}$	0.2
	Learning rate	$\{10^{-4}, 10^{-3}, 10^{-2}\}$	10^{-3}
CVDD	n_{contexts}	$\{10, 50, 200, 500\}$	50
	Orthogonality weight	$\{10^{-4}, 10^{-3}, 10^{-2}\}$	10^{-3}

D.4.2 Framework instantiations (SAE feature space)

All three framework instantiations share KMeans clustering and nearest-centroid cluster assignment; they differ in how the per-cluster sparse subspace is identified. The centroid is used for scoring only by the two FREQMASK-KM variants (L_1 distance to the assigned centroid); LEARNEDMASK-LORA uses it only to assign a test point to a cluster and scores by the SAE reconstruction residual under that cluster's learned mask.

FREQMASK-KM only has two hyperparameters: Cluster count N and mask size k. Selected values are listed in Table 15. The local version of it (Local FREQMASK-KM) uses the same hyperaprameters, but the top-k mask is applied locally by selecting the most frequently activated neurons.

LEARNEDMASK-LORA is trained in two stages, both operating under the SAE space.

Table 11: Hyperparameter sweeps and selected values for activation-space baselines on LLaMA3 8B.

Method	Hyperparameter	Sweep range	Selected
Mahalanobis	Regularization	$\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$	10^{-5}
	Score type	{percentile, sigmoid}	percentile
GMM	$n_{\text{components}}$	$\{3, 5, 10, 50, 100\}$	10
	Initialization	{k-means++, random}	k-means++
	Score normalization	{percentile, sigmoid}	percentile
OC-SVM	ν	$\{0.01, 0.05, 0.1, 0.2\}$	0.1
	γ	$\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$	10^{-2}
	Kernel	{RBF, linear}	RBF
Deep SVDD	Representation dim.	$\{16, 32, 64, 128\}$	128
	ν	$\{0.05, 0.1, 0.2\}$	0.2
	Learning rate	$\{10^{-4}, 10^{-3}, 10^{-2}\}$	10^{-3}
CVDD	n_{contexts}	$\{10, 50, 200, 500\}$	500
	Orthogonality weight	$\{10^{-4}, 10^{-3}, 10^{-2}\}$	10^{-3}

Table 12: Hyperparameter sweeps and selected values for activation-space baselines on Qwen3 8B.

Method	Hyperparameter	Sweep range	Selected
Mahalanobis	Regularization	$\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$	10^{-5}
	Score type	{percentile, sigmoid}	percentile
GMM	$n_{\text{components}}$	$\{3, 5, 10, 50, 100\}$	100
	Initialization	{k-means++, random}	k-means++
	Score normalization	{percentile, sigmoid}	sigmoid
OC-SVM	ν	$\{0.01, 0.05, 0.1, 0.2\}$	0.05
	γ	$\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$	10^{-2}
	Kernel	{RBF, linear}	RBF
Deep SVDD	Representation dim.	$\{16, 32, 64, 128\}$	32
	ν	$\{0.05, 0.1, 0.2\}$	0.1
	Learning rate	$\{10^{-4}, 10^{-3}, 10^{-2}\}$	10^{-2}
CVDD	n_{contexts}	$\{10, 50, 200, 500\}$	50
	Orthogonality weight	$\{10^{-4}, 10^{-3}, 10^{-2}\}$	10^{-4}

Stage 1: per-cluster mask training. A binary mask is learned per cluster using a straight-through estimator (`mask_type = binary_ste`). Concretely, each cluster's mask is parameterized by a vector of learnable logits $\ell \in \mathbb{R}^{d_2}$ (one per SAE feature, initialized to 0.5); the forward pass applies the hard mask $m = \mathbb{1}[\ell > 0]$ to the SAE code via $m \odot z$, while the backward pass replaces the non-differentiable indicator with the derivative of $\sigma(\ell)$, so the logits can be trained end-to-end with SGD despite m being discrete [Bengio et al., 2013]. In practice, the straight-through mask is $m_{\text{ST}} = \sigma(\ell) + \text{stopgrad}(\mathbb{1}[\ell > 0] - \sigma(\ell))$: its forward value is binary, while its gradient follows $\sigma(\ell)$. Sparsity is induced by adding $\lambda_{\text{sparsity}} \|\sigma(\ell)\|_1$ to the reconstruction loss, with the coefficient calibrated per cluster from the sweep $\{10^{-4}, 3\times10^{-4}, 10^{-3}, 3\times10^{-3}, 10^{-2}, 3\times10^{-2}, 10^{-1}\}$. We use $N = 100$ clusters with a minimum cluster size of 50. The clusters are created with an L_2 distance to centroids.

Stage 2: per-cluster LoRA adapter. A LoRA adapter is trained against an SAE-reconstruction objective on the cluster's masked subspace. LoRA rank scales with cluster size: $r = 2$ (small), 4 (medium), 8 (large). Training uses 30 epochs at learning rate 10^{-3} with sparsity weight 10^{-3} on the LoRA output. Stage 2 is gated by a model-specific minimum cluster size: 500 for LLaMA3, 20 for Ministral, 50 for Qwen3, 50 for Gemma, and disabled for Qwen 1.5B and GPT-oss (Stage 1 mask only, no LoRA adaptation), as we observe that LoRA hurts performance on those two models (Appendix G.1).

Table 13: Hyperparameter sweeps and selected values for activation-space baselines on GPT-oss 20B.

Method	Hyperparameter	Sweep range	Selected
Mahalanobis	Regularization	$\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$	10^{-5}
	Score type	{percentile, sigmoid}	sigmoid
GMM	$n_{\text{components}}$	$\{3, 5, 10, 50, 100\}$	10
	Initialization	{k-means++, random}	random
	Score normalization	{percentile, sigmoid}	sigmoid
OC-SVM	ν	$\{0.01, 0.05, 0.1, 0.2\}$	0.1
	γ	$\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$	10^{-4}
	Kernel	{RBF, linear}	RBF
Deep SVDD	Representation dim.	$\{16, 32, 64, 128\}$	16
	ν	$\{0.05, 0.1, 0.2\}$	0.05
	Learning rate	$\{10^{-4}, 10^{-3}, 10^{-2}\}$	10^{-4}
CVDD	n_{contexts}	$\{10, 50, 200, 500\}$	500
	Orthogonality weight	$\{10^{-4}, 10^{-3}, 10^{-2}\}$	10^{-4}

Table 14: Hyperparameter sweeps and selected values for activation-space baselines on Gemma 4 26B.

Method	Hyperparameter	Sweep range	Selected
Mahalanobis	Regularization	$\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$	10^{-5}
	Score type	{percentile, sigmoid}	sigmoid
GMM	$n_{\text{components}}$	$\{3, 5, 10, 50, 100\}$	100
	Initialization	{k-means++, random}	random
	Score normalization	{percentile, sigmoid}	sigmoid
OC-SVM	ν	$\{0.01, 0.05, 0.1, 0.2\}$	0.1
	γ	$\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$	10^{-2}
	Kernel	{RBF, linear}	RBF
Deep SVDD	Representation dim.	$\{16, 32, 64, 128\}$	128
	ν	$\{0.05, 0.1, 0.2\}$	0.1
	Learning rate	$\{10^{-4}, 10^{-3}, 10^{-2}\}$	10^{-4}
CVDD	n_{contexts}	$\{10, 50, 200, 500\}$	500
	Orthogonality weight	$\{10^{-4}, 10^{-3}, 10^{-2}\}$	10^{-4}

D.4.3 Supervised reference

The supervised reference reported in Section 4.2 is a linear probe (with bias) trained on SAE features against binary safe/unsafe labels. Several BCE variants were evaluated for completeness; their configurations are summarized in Table 16.

D.5 Compute

All experiments are run on a SLURM-managed cluster with single NVIDIA V100 or Titan-RTX GPU per job. Approximate wall-clock times per training run are listed in Table 17. We pre-cached the model activations from each dataset, and these methods load the cached activations for training.

E Effective Dimension: Definition and Per-Setting Specialization

Let $X \in \mathbb{R}^{n \times d}$ denote the data matrix in the feature space of interest, with n samples and d features. Let $\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$ be the empirical mean, computed independently within each setting's feature space. The empirical covariance is

$$\Sigma = \frac{1}{n} X^\top X - \bar{x}\, \bar{x}^\top \in \mathbb{R}^{d \times d}. \tag{19}$$

Table 15: Selected hyperparameters for FREQMASK-KM, swept over $N \in \{50, 100, 200\}$ and $k \in \{25, 50, 100, 200, 400\}$. GPT-oss and Gemma select small masks ($k = 50$ and $k = 25$) from the sweep extended to smaller masks (Appendix F.3).

Model	N (clusters)	k (mask size)
Qwen 1.5B	200	100
Ministral 8B	50	200
LLaMA3 8B	100	400
Qwen3 8B	100	400
GPT-oss 20B	50	50
Gemma 4 26B	50	25

Table 16: Supervised BCE variants and their 3-category performance across the six models. AUROC and TPR@5%FPR are each the mean of BeaverTails, ToxiGen, and pooled-HarmBench. All variants use the BCE loss. The SAE + linear (with bias) row is the *linear probe* reported in Table 1 and gives its mean over 5 random seeds (standard deviations are in Table 1). The other variants are single-seed runs (best of learning rate $\in \{10^{-3}, 10^{-4}\}$), reported here for completeness.

		AUROC (↑)						TPR@5%FPR (↑)					
Input	Classifier	Ministral	LLaMA3	Qwen3	GPT-oss	Gemma	Qwen 1.5B	Ministral	LLaMA3	Qwen3	GPT-oss	Gemma	Qwen 1.5B
SAE features	2-layer MLP	0.993	0.959	0.992	0.987	0.978	0.971	0.966	0.902	0.958	0.938	0.937	0.887
Raw activations	2-layer MLP	0.993	0.979	0.992	0.988	0.969	0.975	0.966	0.913	0.962	0.939	0.915	0.802
Raw activations	Linear	0.988	0.972	0.985	0.985	0.961	0.959	0.928	0.748	0.905	0.910	0.751	0.688
SAE features	Linear (with bias)	0.993	0.972	0.993	0.990	0.983	0.982	0.964	0.905	0.957	0.948	0.942	0.876
SAE features	Linear (no bias)	0.985	0.965	0.979	0.973	0.947	0.954	0.898	0.693	0.847	0.858	0.635	0.664

Let $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d \geq 0$ denote its eigenvalues in decreasing order; negative eigenvalues arising from finite-precision arithmetic are clamped to zero. Define the cumulative explained-variance ratio

$$\rho(k) = \frac{\sum_{j=1}^{k} \lambda_j}{\sum_{j=1}^{d} \lambda_j}. \tag{20}$$

The local effective dimension at the 90% variance threshold is

$$d_{90} = \min\{k \in \{1, \ldots, d\} : \rho(k) \geq 0.90\}. \tag{21}$$

Equivalently, d_{90} is the smallest number of principal components required to retain at least 90% of the total variance in X.

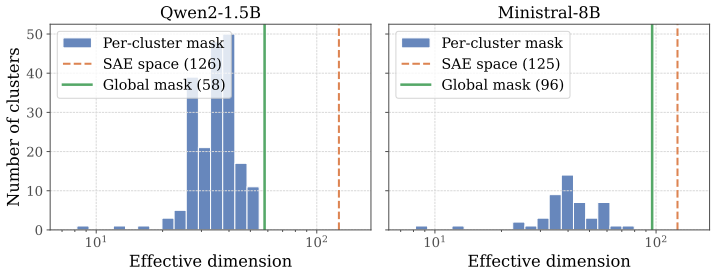

Figure 15: Effective dimension (d_{90}) for two model families not shown in the main text (Figure 3): Qwen2-1.5B and Ministral-8B. We omit Gemma-4-26B because its selected mask has only $k = 25$ features, which already bounds its masked d_{90} values by 25. As in the main text, blue bars show the per-cluster d_{90} distribution, computed on each cluster's members within its own top-k mask $\mathcal{M}_c$; vertical lines mark d_{90} in the raw SAE feature space (dashed orange) and the global-mask subspace (solid green). Per-cluster d_{90} again sits well below the global-mask and raw-SAE values, confirming the low local effective dimension across the five model families shown.

The settings differ only in the choice of X (and hence d):

- **Raw SAE:** $X \in \mathbb{R}^{n \times d_{\text{SAE}}}$, rows are $z_i = \text{ReLU}(W_{\text{enc}} h_i + b_{\text{enc}})$.
- **Global mask:** $X = Z_{:,\mathcal{M}} \in \mathbb{R}^{n \times k}$, where $\mathcal{M} \subset \{1, \ldots, d_{\text{SAE}}\}$, $|\mathcal{M}| = k$, is the global mask's active feature index set.

Table 17: Approximate wall-clock per run on a single V100 or Titan-RTX.

Method	Wall-clock per run
Training SAE	< 5 min
Mahalanobis, GMM, OC-SVM	5–20 min
Deep SVDD, CVDD	≈ 30 min
FREQMASK-KM, Local FREQMASK-KM	≈ 10 min
LEARNEDMASK-LORA	≈ 60 min

- **Per-cluster local mask:** for cluster c with members $\mathcal{I}_c = \{i : \ell_i = c\}$, define the local active set

$$\mathcal{M}_c = \text{top-}k\left(\sum_{i \in \mathcal{I}_c} \mathbf{1}[z_i > 0]\right), \tag{22}$$

  i.e. the k SAE features that fire most often within cluster c. Then $X^{(c)} = Z_{\mathcal{I}_c, \mathcal{M}_c} \in \mathbb{R}^{n_c \times k}$, and we report the distribution of $d_{90}^{(c)}$ over clusters with $n_c \geq \max(20, 2k)$ to avoid rank-limited estimates.

In all settings $d_{90} \leq \min(n-1, d)$, with the per-cluster bound $d_{90}^{(c)} \leq \min(n_c - 1, k)$. Centering is performed independently within each setting's feature space; in particular, for the per-cluster analysis, $\bar{x}^{(c)}$ is the mean over $\mathcal{I}_c$ within the local masked subspace $\mathcal{M}_c$.

F Ablations: Stage-by-Stage Variations on FREQMASK-KM

This section reports six ablations on FREQMASK-KM (Section 3.5, Appendix D.4) targeting the design choices left open by the framework definition, ordered to follow the stage sequence of Section 3: the clustering metric (Appendix F.1), the number of clusters $\mathcal{N}_c$ (Appendix F.2), the mask size k (Appendix F.3), the scoring metric (Appendix F.4), the within-cluster overlap of safe and unsafe active supports (Appendix F.5), which gives a mechanistic reading of the clustering-metric and scoring-metric results, and finally the choice of extraction layer (Appendix F.6).

Goal of these ablations. The framework of Section 3 decomposes the pipeline into four stages (embedding, clustering, local subspace, scoring), each of which admits several principled choices. The ablations below are intended as a worked example of how those choices can be varied one stage at a time within a single instantiation, and what the resulting qualitative orderings look like. They are not a hyperparameter optimization study: the goal is to compare design choices under a fixed setting, not to identify a per-cell optimum. Two consequences follow.

First, all runs in this section use a single seed. The only source of randomness in FREQMASK-KM is the KMeans cluster assignment under uniform random centroid initialization (Algorithm 1); the global frequency mask, centroid estimation, and ℓ_1 scoring are deterministic given the clustering. The seed-to-seed standard deviation of FREQMASK-KM in Table 1 is empirically zero at the precision reported, and we do not expect multi-seed sweeps to change the qualitative orderings reported below.

Second, the sweep grids are narrower than the per-model selection in Appendix D.4: the clustering and scoring ablations use $\mathcal{N}_c \in \{50, 100\}$ and $k \in \{100, 200, 400\}$, the scaling ablations sweep one of the two over a wider range. The grid omits $\mathcal{N}_c = 200$, which is the value selected for Qwen 1.5B in the main results, so the Qwen entries below peak at a slightly lower AUROC than the main-table figure. We accept this gap because the goal is the cross-method ordering at a fixed setting, not the optimum for any one method.

Performance is reported as mean AUROC across the three evaluation slices (BeaverTails, ToxiGen, pooled HarmBench adversarial attacks).

F.1 Clustering Metric

Setup. The clustering stage in Section 3.2 permits magnitude-aware and binarized metrics. We compare three KMeans variants in SAE feature space: $\rho_{L_2}(x, x') = \|z(x) - z(x')\|_2$ (the default in Section 3.5), $\rho_{L_1}(x, x') = \|z(x) - z(x')\|_1$, and $\rho_{L_0}(x, x') = \|b(x) - b(x')\|_1$ (Hamming distance on the binarized active-feature indicator $b(x)$ from Section 3.2). All other components are held fixed: global frequency mask, ℓ_1 centroid scoring, and a sweep over $(\mathcal{N}_c, k) \in \{50, 100\} \times \{100, 200, 400\}$

with the best cell reported per model. The ρ_{L_2} row is each model's default FREQMASK-KM configuration (Appendix D.4); for GPT-oss this configuration uses a smaller mask ($k = 50$) than the binarized-metric grid above.

Results. Table 18 reports the best 3-slice mean AUROC per clustering metric and per model. Two observations.

First, ρ_{L_2} is the best metric on four of the five $\geq$8B models. The margin over the next-best metric ranges from 0.10 AUROC on Ministral down to 0.013 on Qwen3 and 0.003 on GPT-oss. The exception is Gemma, where ρ_{L_1} edges ahead (0.776 vs. 0.762 for ρ_{L_2}, a 0.014 gap). On Qwen 1.5B, ρ_{L_2} places third, behind ρ_{L_0} (gap 0.014) and ρ_{L_1} (gap 0.003).

Second, the binarized-clustering penalty is model-dependent. ρ_{L_0} depends on $z(x)$ only through $b(x)$ and so discards the magnitude information in $z(x)$; on Ministral, LLaMA3, and Gemma it is the worst metric, consistent with adversarial inputs being separable from safe inputs primarily by activation magnitude on shared active features rather than by the active support itself (Appendix F.5 presents direct evidence). On Qwen3 and GPT-oss, by contrast, ρ_{L_0} is the second-best metric (within 0.013 and 0.003 of ρ_{L_2}): the magnitude information that ρ_{L_2} clustering exploits is less pivotal for these two models' cluster assignments, even though magnitude still carries the discriminative signal at scoring time.

Table 18: Clustering metric for FREQMASK-KM, holding the global frequency mask and ℓ_1 scoring fixed. AUROC is the 3-slice mean; the ρ_{L_2} row is each model's default configuration and the other rows report the best of a $(\mathcal{N}_c, k)$ sweep over $\{50, 100\} \times \{100, 200, 400\}$. Single seed; bold marks the best metric per column.

Metric	Qwen 1.5B	Ministral 8B	LLaMA3 8B	Qwen3 8B	GPT-oss 20B	Gemma 4 26B
ρ_{L_2} (default)	0.745	**0.914**	**0.937**	**0.915**	**0.900**	0.762
ρ_{L_1}	0.748	0.817	0.914	0.882	0.839	**0.776**
ρ_{L_0}	**0.759**	0.785	0.860	0.902	0.897	0.731

F.2 Number of Clusters

Setup. We sweep $\mathcal{N}_c \in \{10, 25, 50, 100, 200, 500\}$ at *each model's default mask size* k (Qwen 1.5B 100, Ministral 200, LLaMA3 400, Qwen3 400, GPT-oss 50, Gemma 25; Table 15), holding ρ_{L_2} clustering, global mask, and ℓ_1 scoring fixed. Fixing k per model (rather than a single shared value) keeps each model in the mask regime it was selected for: a shared mask far above the small-mask models' optima collapses their adversarial slice at every $\mathcal{N}_c$, so a shared setting would understate those rows.

Results. Table 19 reports the resulting 3-slice mean AUROC. With each model held at its own mask size, all six rows peak inside the sweep and fall off toward both ends, with $\mathcal{N}_c = 500$ uniformly the worst or near-worst entry. The peak $\mathcal{N}_c$ tracks the per-model selection in Appendix D.4: Qwen 1.5B at 25, Ministral and Gemma at 50, LLaMA3 and Qwen3 at 100, and GPT-oss at the small end (10–25, 0.927). The two small-mask models (GPT-oss, Gemma) are no longer depressed—at their own k they reach 0.927 and 0.856, in line with their main-table FREQMASK-KM values, rather than the much lower figures a shared large mask would produce. The Ministral row still shows large fluctuation outside its $\mathcal{N}_c = 50$ peak, driven by the adversarial slice collapsing at $\mathcal{N}_c \in \{25, 100, 200, 500\}$; we do not have a mechanistic explanation and read it as a sensitivity of Ministral's representation geometry to $\mathcal{N}_c$ rather than as noise.

Does the mask do real work? A best-case no-mask comparison. Table 19 varies $\mathcal{N}_c$ with the global frequency mask *on*. To test whether the mask is load-bearing rather than cosmetic, we give the no-mask condition its own best case: we disable the mask entirely (`use_masking=false`, so ℓ_1 scoring runs over *all* SAE features with no top-k restriction) and re-sweep $\mathcal{N}_c \in \{10, 25, 50, 100, 200, 500\}$, holding ρ_{L_2} clustering and ℓ_1 scoring fixed, so the comparison does not depend on a cluster count chosen for the masked configuration. Table 20 gives the full sweep; the best no-mask entry per model, against the deployed masked configuration, is the table discussed in Section 4.3.

Table 19: Number of clusters for FREQMASK-KM, sweeping $\mathcal{N}_c$ at *each model's default mask size* k (Qwen 1.5B 100, Ministral 200, LLaMA3 400, Qwen3 400, GPT-oss 50, Gemma 25; Table 15), with ρ_{L_2} clustering, global mask, and ℓ_1 scoring. AUROC is the 3-slice mean. Single seed; bold marks the best $\mathcal{N}_c$ per row. The complementary no-mask analysis (mask disabled, $\mathcal{N}_c$ swept) is in Table 20.

$\mathcal{N}_c$	10	25	50	100	200	500
Qwen 1.5B	0.745	**0.770**	0.729	0.745	0.762	0.720
Ministral 8B	0.879	0.767	**0.914**	0.768	0.774	0.768
LLaMA3 8B	0.884	0.916	0.933	**0.936**	0.898	0.876
Qwen3 8B	0.880	0.903	0.848	**0.914**	0.885	0.880
GPT-oss 20B	**0.927**	0.926	0.900	0.875	0.835	0.832
Gemma 4 26B	0.841	0.841	**0.856**	0.801	0.802	0.783

Table 20: FREQMASK-KM with the global mask *disabled* (`use_masking=false`, ℓ_1 over all SAE features), sweeping $\mathcal{N}_c$ with ρ_{L_2} clustering and ℓ_1 scoring. AUROC is the 3-slice mean; single seed; bold marks the best $\mathcal{N}_c$ per row. "Masked best" is each model's main-table FREQMASK-KM AUROC (Table 1), where the mask is on. These are the two columns compared in Table 3.

$\mathcal{N}_c$	10	25	50	100	200	500	Masked best
Qwen 1.5B	0.769	0.766	0.747	0.759	**0.776**	0.736	0.762
Ministral 8B	0.875	0.843	**0.904**	0.864	0.865	0.859	0.914
LLaMA3 8B	0.895	0.919	0.935	**0.937**	0.902	0.880	0.937
Qwen3 8B	0.883	0.904	0.865	**0.914**	0.896	0.898	0.915
GPT-oss 20B	**0.781**	0.780	0.752	0.748	0.735	0.745	0.900
Gemma 4 26B	0.736	0.718	**0.766**	0.707	0.737	0.733	0.856

F.3 Mask Size

Setup. We sweep $k \in \{25, 50, 100, 200, 400, 800, 1600\}$ at fixed $\mathcal{N}_c = 100$, holding ρ_{L_2} clustering, global mask, and ℓ_1 scoring fixed.

Results. In Table 21, Three of the five $\geq$8B models (Ministral, LLaMA3, Qwen3) gain as k grows and then saturate. Ministral rises monotonically: $0.654 \to 0.817 \to 0.858$ across $k = 25, 400, 1600$. LLaMA3 has a non-monotone region at small k (0.805 at $k = 25$, dropping to 0.701–0.702 at $k \in \{50, 100\}$), then recovers and saturates at $k \geq 200$ with AUROC $\approx$ 0.93–0.94. Qwen3 collapses at small k (0.38 at $k \in \{25, 50\}$) but jumps to 0.793 at $k = 100$ and saturates at 0.915 for $k \geq 200$. GPT-oss and Gemma run in the opposite direction: both peak at the smallest masks (0.875 at $k = 50$ for GPT-oss, 0.820 at $k = 50$ for Gemma) and decay as the mask grows—GPT-oss to a plateau of 0.748 for $k \geq 200$ and Gemma to 0.55–0.70, the two models for which a larger mask hurts. These small-mask peaks are taken at the table's fixed $\mathcal{N}_c = 100$; each model's main-table value (0.900 for GPT-oss, 0.856 for Gemma; Table 1) is reached only when the small mask is paired with $\mathcal{N}_c = 50$, the joint corner this $\mathcal{N}_c = 100$ slice does not reach. Qwen 1.5B is essentially flat: AUROC fluctuates within 0.717–0.759 across the full range and shows no monotone trend. For the models that benefit from a larger mask, the saturation point ($k \approx 400$ on LLaMA3 and Qwen3, $k \approx 800$ on Ministral) sets a soft upper bound on how much the global mask helps before adding features stops contributing discriminative signal.

Table 21: Mask size for FREQMASK-KM at $\mathcal{N}_c = 100$, ρ_{L_2} clustering, global mask, ℓ_1 scoring. AUROC is the 3-slice mean. Single seed; bold marks the best k per row. Most $\geq$8B models favor a larger mask and then plateau; GPT-oss is the exception, peaking at a small mask.

k	25	50	100	200	400	800	1600
Qwen 1.5B	0.717	0.741	0.745	0.734	0.736	**0.759**	**0.759**
Ministral 8B	0.654	0.667	0.729	0.768	0.817	0.845	**0.858**
LLaMA3 8B	0.805	0.701	0.702	0.929	**0.937**	**0.937**	**0.937**
Qwen3 8B	0.383	0.378	0.793	**0.915**	**0.915**	**0.915**	**0.915**
GPT-oss 20B	0.863	**0.875**	0.832	0.748	0.748	0.748	0.748
Gemma 4 26B	0.801	**0.820**	0.729	0.549	0.684	0.697	0.701

F.4 Scoring Metric: ℓ_1 vs. ℓ_2 vs. Mahalanobis

Setup. FREQMASK-KM as defined in Section 3.5 uses ℓ_1 centroid distance on the masked SAE features. We compare ℓ_1, ℓ_2, and a per-cluster Mahalanobis distance on the same masked features, holding ρ_{L_2} KMeans clustering and the global frequency mask fixed. The Mahalanobis variant fits a per-cluster covariance on safe samples and adds a regularizer λI before inversion, swept over $\lambda \in \{10^{-3}, 10^{-2}, 10^{-1}\}$. For each scoring rule we sweep $(\mathcal{N}_c, k) \in \{50, 100\} \times \{100, 200, 400\}$ and report the best cell per model.

Results. Table 22 reports best 3-slice mean AUROC per scoring rule on each model. ℓ_1 and ℓ_2 are within 0.012 AUROC of each other on every model except GPT-oss and Gemma, where the gap widens to 0.032 and 0.035 respectively; ℓ_1 wins on Ministral, LLaMA3, and Gemma, while ℓ_2 wins on Qwen3, GPT-oss, and Qwen 1.5B. The two magnitude-aware rules thus remain close, with the choice between them model-dependent. Mahalanobis trails both magnitude-aware rules on every $\geq$8B model except Gemma, where it is in fact the best scoring rule (0.785, against ℓ_1's 0.762). Across models it benefits from a large swept regularizer—rising monotonically to the largest value $\lambda = 10^{-1}$ on the other models, and peaking one step earlier at $\lambda = 10^{-2}$ on Gemma (0.785 vs. 0.783 at 10^{-1})—while at $\lambda = 10^{-3}$ AUROC drops to 0.56–0.81 depending on model. This is consistent with the per-cluster covariance being poorly conditioned on the masked SAE features at the cluster sample sizes available on most models, and substantiates the choice of a magnitude-aware ℓ_1 scoring in Algorithm 1 on those models; Gemma is the lone exception, with a better-conditioned per-cluster covariance whose origin we do not fully diagnose.

Table 22: Scoring metric for FREQMASK-KM, holding ρ_{L_2} KMeans clustering and the global frequency mask fixed. AUROC is the 3-slice mean; each cell is the best of a $(\mathcal{N}_c, k)$ sweep over $\{50, 100\} \times \{100, 200, 400\}$. The Mahalanobis row additionally sweeps $\lambda \in \{10^{-3}, 10^{-2}, 10^{-1}\}$ and selects the best λ (the largest, 10^{-1}, except Gemma, which selects 10^{-2}). Bold marks the best scoring rule per column.

Scoring	Qwen 1.5B	Ministral 8B	LLaMA3 8B	Qwen3 8B	GPT-oss 20B	Gemma 4 26B
ℓ_1	0.745	**0.914**	**0.937**	0.915	0.851	0.762
ℓ_2	**0.749**	0.902	0.931	**0.917**	**0.883**	0.727
Mahalanobis	0.743	0.885	0.848	0.770	0.860	**0.785**

F.5 Within-Cluster Active-Support Overlap of Safe and Unsafe Points

Setup. The framework assumes that within a cluster C_c, safe samples occupy a small active-feature support, and the scoring rule operates on the projection onto that support. The remaining question is whether unsafe samples assigned to C_c activate a *different* support from safe samples, or the *same* support at different magnitudes. We measure this directly. Let C_c^{safe} and C_c^{unsafe} denote the safe and unsafe samples assigned to cluster c at evaluation time. For each cluster, we define the population active set

$$S_c^{\text{safe}} = \Big\{ j \in [d_2] : \tfrac{1}{|C_c^{\text{safe}}|} \sum_{x \in C_c^{\text{safe}}} \mathbf{1}[z(x)_j > 0] > 0.5 \Big\}, \qquad S_c^{\text{unsafe}} \text{ analogously.} \tag{23}$$

The inner expression is the within-cluster firing frequency of SAE feature j on the safe population: the fraction of safe samples in C_c for which feature j is active. The set S_c^{safe} collects the features that fire on a strict majority ($> 50\%$) of safe samples in the cluster; we treat these as the typical active features for the safe population in that cluster. S_c^{unsafe} is the analogous set for unsafe samples assigned to the same cluster. The framework's local-sparsity assumption is that $|S_c^{\text{safe}}|$ is small, by construction of the clustering; the question this section asks is whether S_c^{unsafe} differs from S_c^{safe}.

Results. Table 23 reports the restricted Jaccard distribution. The median is ≥ 0.94 on all six models, and reaches 0.98 on Qwen3: within a cluster, safe and unsafe points populate nearly identical masked active supports. The anomaly score therefore is not driven by which features are active in unsafe inputs, but by how strongly they are active. This is the structural reason that magnitude-aware scoring (ℓ_1, ℓ_2) beats the binarized alternative in Appendix F.4: discarding the magnitude information loses the discriminative signal. At the clustering stage the magnitude penalty is more model-dependent—ρ_{L_0} is competitive on Qwen3 and GPT-oss—as discussed in Appendix F.1.

Table 23: Within-cluster active-support overlap of safe and unsafe points under the FREQMASK-KM global mask. Counts are masked features that fire in $> 50\%$ of the corresponding population within each cluster, summed across clusters; "Jaccard median" is the per-cluster restricted Jaccard, taken over clusters with at least one safe and one unsafe sample assigned.

Model	k	Both	Safe only	Unsafe only	Neither	Jaccard median
Qwen 1.5B	100	98	0	0	1	1.00
Ministral 8B	200	139	20	2	40	0.94
LLaMA3 8B	400	67	3	2	329	0.95
Qwen3 8B	400	53	1	1	345	0.98
GPT-oss 20B	50	36	1	2	10	0.95
Gemma 4 26B	25	15	1	1	8	0.95

Summary. Reading the four ablations in stage order: ρ_{L_2} is the best clustering metric on four of the five $\geq$8B models (Gemma is the exception, narrowly preferring ρ_{L_1}), though its margin over the binarized ρ_{L_0} shrinks to 0.003–0.013 on Qwen3 and GPT-oss; the best $\mathcal{N}_c$ and k vary widely across models, within the ranges $\mathcal{N}_c \in \{10, 25, 50, 100\}$ and $k \in \{25, \ldots, 1600\}$, with no single setting dominating and GPT-oss and Gemma preferring the small-$\mathcal{N}_c$, small-k corner that the others avoid; ℓ_1 and ℓ_2 scoring perform within 0.012 AUROC of each other on every model except GPT-oss (0.032) and Gemma (0.035), with the better of the two model-dependent, while per-cluster Mahalanobis trails both on every $\geq$8B model except Gemma, and is best at the largest swept regularizer on every model but Gemma (which selects 10^{-2}), broadly consistent with ill-conditioned per-cluster covariance estimates on most models. The within-cluster Jaccard analysis in Appendix F.5 gives a direct mechanistic reason for why magnitude-aware scoring (ℓ_1, ℓ_2) outperforms its binary counterpart: safe and unsafe points within a cluster activate the same masked features, and the discriminative signal is on magnitudes.

These orderings are specific to FREQMASK-KM, and several of them follow from choices made elsewhere in the pipeline rather than from the stage in question alone: the ℓ_1 scoring rule is paired with a global frequency mask on SAE features whose entries are nonnegative, the per-cluster covariance estimates that disadvantage Mahalanobis are computed on a k-dimensional masked subspace with $|C_c|/k$ in the low-tens range, and the $\mathcal{N}_c$ peak depends on the safe sample size N and the global sparsity s (Section 3.2). Other instantiations of the framework, with different embedding spaces, mask constructions, or learned per-cluster subspaces, may legitimately prefer different choices at the same stage. Table 2 already shows that LEARNEDMASK-LORA reaches the same performance band under a different scoring rule (reconstruction residual rather than ℓ_1 centroid distance), and this section is intended as an ablation study of how to read the framework's design space, not as a prescription that transfers to those instantiations unchanged.

F.6 Extraction Layer

The main results read activations from a single mid-to-late layer per model, chosen by the heuristic described in Section 4.1. To test that choice, we sweep the extraction layer on three model families, training a fresh SAE at each layer and running FREQMASK-KM with every other hyperparameter held at the model's published setting. AUROC in this subsection is computed against pooled HarmBench.

On LLaMA3-8B and GPT-oss-20B the layer used in the main results gives the highest AUROC of those swept (0.956 and 0.966), with the remaining layers spanning 0.785–0.924 and 0.842–0.932 respectively. On Gemma-4-26B the shallow layers are the strongest (0.940 at layer 8 and 0.963 at layer 12) and the layer used in the main results reaches 0.828.

Takeaway. The mid-to-late heuristic is a reasonable default on all three models: it selects the best available layer on two of them, and a workable one on the third. Which layer is best is model-dependent, and it does not correspond to a fixed depth or a fixed fraction of depth, so it appears to follow each model's architecture rather than a rule that transfers across families. A per-model sweep can therefore recover a small amount of additional AUROC, at the cost of extra computations.

F.7 SAE Sparsity Penalty

The framework operates in the feature space of a pretrained SAE, so its performance could in principle depend on how that SAE was trained. We test this by sweeping the SAE's main training hyperparameter, the sparsity penalty λ_{ℓ_1} of Appendix D.1, over three orders of magnitude on four

Table 24: FREQMASK-KM AUROC against pooled HarmBench as a function of the extraction layer. A fresh SAE is trained at each layer; all other hyperparameters are held at each model's published setting. Bold marks the layer used in the main results. The layer used in the main results is the strongest of those swept on LLaMA3-8B and GPT-oss-20B; on Gemma-4-26B the shallow layers score higher.

LLaMA3-8B								
Layer	8	12	16	20	24	**25**	28	30
AUROC	0.924	0.913	0.800	0.802	0.799	**0.956**	0.785	0.921

GPT-oss-20B								
Layer	4	8	12	16	18	**20**	22	23
AUROC	0.924	0.932	0.927	0.896	0.924	**0.966**	0.842	0.861

Gemma-4-26B						
Layer	8	12	16	20	**25**	29
AUROC	0.940	0.963	0.934	0.807	**0.828**	0.930

model families, and rerunning FREQMASK-KM unchanged on each resulting encoder: only the SAE checkpoint changes, while $\mathcal{N}_c$, k, the extraction layer and every other setting stay at the model's published value. Table 25 reports, for each encoder, the density it reaches (the measured average number of active features per input, L_0, out of $d_{\text{SAE}} = 16384$) together with the AUROC it yields. AUROC here is the three-category mean of Table 1, and the deployed rows reproduce that table.

Detection is stable over a wide range of penalties rather than depending on one setting. On LLaMA3-8B, AUROC stays between 0.858 and 0.936 while λ_{ℓ_1} moves by a factor of 1000 and L_0 by a factor of seven, and Qwen2-1.5B stays between 0.643 and 0.762 at every setting, consistent with its representation quality rather than its SAE bounding what unsupervised detection can recover on that model (Section 4.2). The clear losses occur where the encoder fails to sparsify: the two Gemma-4-26B encoders trained for twenty epochs reach $L_0 = 81$ and 103 and fall to 0.627 and 0.665, well below the other three settings on that model.

The penalty λ_{ℓ_1} does not determine sparsity on its own, since training length also affects the density reached: the same $\lambda_{\ell_1} = 3$ on Gemma-4-26B gives $L_0 = 54$ after ten epochs and $L_0 = 81$ after twenty. The measured L_0, not λ_{ℓ_1}, is therefore the quantity on which SAE sparsity is comparable, which is also why λ_{ℓ_1} is not shared across models: the two largest models need a much larger penalty to reach a comparable density because their residual-stream activations have a larger RMS (Appendix D.1).

G Ablations for LEARNEDMASK-LORA

This section reports three ablations on LEARNEDMASK-LORA (Section 3.5, Appendix D.4) targeting choices that are not pinned down by the framework definition: whether the per-cluster LoRA stage is necessary (Appendix G.1), whether Mahalanobis scoring on the LoRA-adapted reconstruction outperforms the ℓ_2 whitened residual used in the main results (Appendix G.2), and whether the binary straight-through mask in Stage 1 can be replaced by a continuous mask without loss (Appendix G.3).

Unless otherwise noted, all runs use the configuration of Appendix D.4: cosine binarized clustering with $\mathcal{N}_c = 50$ clusters for Ministral 8B and LLaMA3 8B and $\mathcal{N}_c = 100$ for the other four models, whitened residual normalization with shrinkage $\alpha = 0.3$, Stage 1 logit initialization at 0.5, LoRA rank schedule $r \in \{2, 4, 8\}$ keyed to cluster size, 30 LoRA epochs at learning rate 10^{-3}, and LoRA sparsity weight 10^{-3}. The minimum cluster size for LoRA Stage 2 is 20 for Qwen 1.5B and Ministral 8B, 500 for LLaMA3 8B, and 50 for Qwen3 8B and Gemma 4 26B. Performance is reported as mean AUROC across the three evaluation slices (BeaverTails, ToxiGen, pooled HarmBench adversarial attacks).

Relation to Table 1. The Stage 1+Stage 2 row of Table 26 matches the main table for every model except Qwen3 8B, where it reads 0.914 against 0.917. The ablation applies the rank schedule above, whereas the Qwen3 main-table run used a flat $r = 8$. The gap is 0.003 AUROC and does not change the conclusion that Stage 2 helps on Qwen3: $+0.046$ with the schedule, $+0.049$ without.

Table 25: FREQMASK-KM AUROC as a function of the SAE sparsity penalty λ_{ℓ_1}. Only the SAE checkpoint changes; all other hyperparameters are held at each model's published setting. L_0 is the measured average number of active SAE features per input, out of 16384. Epochs are listed because λ_{ℓ_1} and training length jointly determine the density reached. AUROC is the three-category mean of Table 1. Bold marks the encoder used in the main results.

Model	λ_{ℓ_1}	Epochs	L_0	AUROC
LLaMA3-8B	**0.01**	**5**	**66**	**0.936**
	1	10	58	0.861
	3	10	22	0.858
	10	10	9	0.910
Qwen2-1.5B	**0.01**	**10**	**227**	**0.762**
	0.1	10	213	0.758
	1	10	153	0.719
	3	10	91	0.643
	10	10	53	0.696
GPT-oss-20B	0.01	20	393	0.846
	300	20	103	0.832
	700	**20**	**48**	**0.900**
Gemma-4-26B	1	10	55	0.823
	1	20	103	0.665
	3	**10**	**54**	**0.856**
	3	20	81	0.627
	10	10	31	0.835

G.1 LoRA Stage Contribution

Setup. We compare the Stage 1 binary-mask checkpoint against the full two-stage pipeline that adds a per-cluster LoRA adapter on top. Both rows reuse the same Stage 1 checkpoints and clustering; the only difference is whether Stage 2 is applied. Five seeds per row.

Results. Table 26 reports mean and standard deviation of 3-slice AUROC across 5 seeds. The LoRA stage helps on four of the five $\geq$8B models—$+0.062$ AUROC on Ministral 8B, $+0.096$ on LLaMA3 8B, $+0.046$ on Qwen3 8B, and $+0.051$ on Gemma 4 26B—but degrades two models: Qwen 1.5B (-0.033) and GPT-oss 20B (-0.101). GPT-oss thus joins Qwen 1.5B as a model on which the Stage 2 adapter hurts, so we report it Stage 1-only in the main results and disable Stage 2 on both.

Table 26: Effect of the per-cluster LoRA stage on LEARNEDMASK-LORA. AUROC is the 3-slice mean (BeaverTails, ToxiGen, pooled HarmBench), reported as mean $\pm$ standard deviation across 5 random seeds.

Method	Qwen 1.5B	Ministral 8B	LLaMA3 8B	Qwen3 8B	GPT-oss 20B	Gemma 4 26B
LEARNEDMASK (Stage 1 only)	0.794 ± 0.001	0.834 ± 0.001	0.816 ± 0.001	0.868 ± 0.000	0.919 ± 0.001	0.853 ± 0.000
LEARNEDMASK + LORA (Stage 1 + Stage 2)	0.761 ± 0.001	0.896 ± 0.001	0.912 ± 0.001	0.914 ± 0.001	0.818 ± 0.002	0.904 ± 0.000
Δ from LoRA	-0.033	$+0.062$	$+0.096$	$+0.046$	-0.101	$+0.051$

G.2 Scoring Function: Mahalanobis vs. Whitened ℓ_2

Setup. The main LEARNEDMASK-LORA results in Table 2 use the ℓ_2 norm of the whitened (masked) SAE-reconstruction residual under the assigned cluster's adapted encoder/decoder pair. We test whether replacing this with a per-cluster Mahalanobis distance on the same residual improves performance. The Mahalanobis variant fits a per-cluster covariance on the safe residuals and adds a regularizer λI before inversion; we sweep $\lambda \in \{10^{-4}, 10^{-2}, 10^{-1}, 1.0\}$, single seed per cell, on Stage 1 checkpoints with hyperparameters matched to the multi-seed configuration of Appendix G.1.

Results. Table 27 reports 3-slice mean AUROC for the full λ sweep, the best Mahalanobis cell per model, and the whitened ℓ_2 reference from Table 26. Best λ is large for the $\geq$8B models ($\lambda = 1.0$) and intermediate for Qwen 1.5B ($\lambda = 10^{-1}$), with the exception of Gemma, whose best λ is the smallest (10^{-4}); at small λ (10^{-4}) the per-cluster covariance is otherwise ill-conditioned and

AUROC collapses. At its best λ, Mahalanobis reaches $0.793/0.890/0.904/0.879/0.577/0.785$ on the six models, against the whitened ℓ_2 reference of $0.794/0.896/0.912/0.914/0.818/0.904$. The two scoring rules are within 0.008 AUROC on the original three models, but the gap widens on the newer models (0.035 on Qwen3, 0.119 on Gemma, and 0.241 on GPT-oss); on no model does Mahalanobis exceed the whitened ℓ_2 reference. We retain whitened ℓ_2 as the default, since it requires no per-cluster covariance fit and no regularization sweep.

Table 27: Scoring function comparison on LEARNEDMASK-LORA. Mahalanobis scoring on the per-cluster reconstruction residual is swept over the regularizer λ (1 seed per cell); the "best Mahal." column reports the best λ per model. The "whitened ℓ_2" column is the multi-seed reference from Table 26. AUROC is the 3-slice mean; bold marks the best λ within each model.

	Mahalanobis, λ				best Mahal.	whitened ℓ_2
Model	10^{-4}	10^{-2}	10^{-1}	1.0	(selected λ)	(reference)
Qwen 1.5B	0.602	0.738	**0.793**	0.786	0.793 (λ=10^{-1})	0.794
Ministral 8B	0.426	0.653	0.885	**0.890**	0.890 (λ=1.0)	0.896
LLaMA3 8B	0.576	0.785	0.897	**0.904**	0.904 (λ=1.0)	0.912
Qwen3 8B	0.691	0.703	0.710	**0.879**	0.879 (λ=1.0)	0.914
GPT-oss 20B	0.544	0.546	0.552	**0.577**	0.577 (λ=1.0)	0.818
Gemma 4 26B	**0.785**	0.782	0.776	0.749	0.785 (λ=10^{-4})	0.904

G.3 Mask Parameterization: Continuous vs. Binary STE

Setup. Stage 1 of LEARNEDMASK-LORA learns a binary mask $m_c \in \{0,1\}^{d_2}$ via a straight-through estimator (Appendix D.4). We replace this with a continuous mask $m_c \in [0,1]^{d_2}$ trained against the same Stage 1 reconstruction objective, then apply the standard Stage 2 LoRA on top. Hyperparameters match Appendix G.1 otherwise; single seed.

Results. Table 28 reports 3-slice mean AUROC and the continuous-variant FPR@95. On all five $\geq$8B models the two parameterizations are within 0.007 AUROC of each other ($+0.004$ on Ministral, -0.007 on LLaMA3, -0.003 on Qwen3, $+0.005$ on GPT-oss, -0.001 on Gemma); the continuous-mask row is a single seed, so we cannot resolve whether these gaps reflect a real difference or seed variation, and we treat the two parameterizations as indistinguishable at this granularity. On Qwen 1.5B the continuous variant underperforms the Stage 1 binary baseline by 0.046 AUROC. Note that for GPT-oss both rows sit near 0.82 rather than near its main-table AUROC: this ablation holds the Stage 2 LoRA adapter *on* while varying the mask type, and LoRA degrades GPT-oss (Table 26), so both mask types fall below the Stage 1-only configuration GPT-oss actually deploys (0.919, Table 1); the within-row binary-vs-continuous gap ($+0.005$) is unaffected. Binary STE is retained as the default since it does not meaningfully underperform the continuous variant on any model and yields a sparser mask.

Table 28: Mask parameterization in Stage 1 of LEARNEDMASK-LORA, with Stage 2 LoRA applied on both rows. "Binary STE + LoRA" reproduces the multi-seed reference from Table 26 (mean $\pm$ std over 5 seeds). "Continuous + LoRA" is a single seed.

Method	Qwen 1.5B	Ministral 8B	LLaMA3 8B	Qwen3 8B	GPT-oss 20B	Gemma 4 26B
3-slice mean AUROC						
Binary STE + LoRA (multi-seed)	0.794 ± 0.001	0.896 ± 0.001	0.912 ± 0.001	0.914 ± 0.001	0.818 ± 0.002	0.904 ± 0.000
Continuous + LoRA (1 seed)	0.764	0.900	0.905	0.911	0.823	0.903
Δ	-0.046	$+0.004$	-0.007	-0.003	$+0.005$	-0.001
Continuous + LoRA FPR@95 ($\downarrow$)	0.594	0.253	0.250	0.433	0.456	0.229

Summary. The LoRA stage improves the performance of LEARNEDMASK-LORA on Ministral, LLaMA3, Qwen3, and Gemma, and is correctly disabled on the two models it hurts (Qwen 1.5B and GPT-oss). The whitened-ℓ_2 scoring rule used in the main results is not improved upon by per-cluster Mahalanobis at any swept regularizer on any model. The binary STE Stage 1 mask is retained over a continuous-mask variant since the binary method yields a sparser mask and does not perform worse than a continuous one.

H Adaptive Attacks Against the Deployed Detector

A static detector exposes a single fixed decision rule, a score and one operating threshold, and a white-box attacker who optimizes against that rule can almost always find an input that stays on the safe side of the threshold yet remains harmful. Detection-based defenses for LLMs have repeatedly been broken this way [Andriushchenko et al., 2024, Nasr et al., 2025], and we expected the same here. We tested it directly: with FREQMASK-KM used as an adversarial defense detector at its deployment threshold, adaptive attack [Andriushchenko et al., 2024] succeeds on 98% of 50 AdvBench behaviors on LLaMA3-8B and 94% on Qwen2-1.5B, and the detector flags none of them. That attacker optimizes only the model's refusal signal and never the anomaly score, so a detector-aware attacker would be at least as strong. We therefore do not claim robustness to detector-aware adaptive attacks; what we claim is unsupervised detection of new, non-adaptive unsafe inputs, which is a different threat model.

Table 29: A white-box adversarial-suffix jailbreak against the guarded model, with FREQMASK-KM detecting harmful generation at its deployment threshold.

Model	Attack success rate	Detector block rate
LLaMA3-8B	98% $(49/50)$	0% $(0/50)$
Qwen2-1.5B	94% $(47/50)$	0% $(0/50)$

The outcome is a property of static defenses rather than of this detector in particular: the defense is fixed first and the attacker moves second. It also does not contradict the detection results of Section 4.2. AUROC measures how well the score ranks unsafe inputs above safe ones on a fixed evaluation distribution, whereas the attack here moves the input distribution until individual points fall below one chosen threshold, so a detector can rank well and still be evaded at a fixed operating point. Designing a defense robust to a sufficiently strong white-box adaptive attacker remains an open problem and is beyond the scope of this paper.

I Proofs for the Theory Section

We use the notation of Section 6. Throughout, the representation z, the population distribution P_0, the selected population centers, and K, k are fixed before drawing $\mathcal{D}_n \sim P_0^n$. An SAE fitted on an independent pilot sample is allowed by conditioning on that sample. Merely freezing an SAE after fitting it on $\mathcal{D}_n$ does not give this independence. The results below do not analyze that additional reuse of data or adaptive hyperparameter selection.

All probabilities conditional on $\mathcal{D}_n$ concern a fresh test point. After a permutation of the empirical center labels, we transport the assignment labels by the same permutation; we do not recompute a different tie-breaking rule. Empty oracle clusters are assigned mean zero by convention and are excluded on the count event used below.

I.1 Population centers and oracle sample means

We first justify the conditional-mean identity used in the main text. Let $V_c = \{z : c^\star(z) = c\}$. For any $u \in \mathbb{R}^{d_2}$,

$$\mathbb{E}[\|Z - u\|_2^2 \mid Z \in V_c] = \mathbb{E}[\|Z - \mathbb{E}[Z \mid Z \in V_c]\|_2^2 \mid Z \in V_c] + \|u - \mathbb{E}[Z \mid Z \in V_c]\|_2^2.$$

If a center with $\pi_c > 0$ differed from this conditional mean, replacing it would strictly reduce the objective with the old assignments held fixed. Reassigning to the nearest new centers cannot increase that objective. This contradicts population optimality and proves $\mu_c^\star = \mathbb{E}[Z \mid c^\star(Z) = c]$.

Lemma 1. *If $n\pi_{\min} \geq 8\log(4K/\delta)$, then with probability at least $1 - \delta/4$,*

$$N_c \geq n\pi_c/2 \geq n\pi_{\min}/2 \qquad (c \in [K]). \tag{24}$$

Proof. For each c, $N_c \sim \mathrm{Bin}(n, \pi_c)$. The multiplicative Chernoff bound gives $\Pr(N_c < n\pi_c/2) \leq \exp(-n\pi_c/8)$. Summing over K clusters proves the claim. For completeness, this Chernoff bound follows by applying Markov's inequality to e^{-tN_c}, using $\mathbb{E}e^{-tN_c} \leq \exp(n\pi_c(e^{-t} - 1))$, and taking $t = \log 2$: the exponent is $-n\pi_c(1 - \log 2)/2 \leq -n\pi_c/8$. □

Lemma 2. *Under Assumption 1 and the count condition of Lemma 1, with probability at least* $1-\delta/2$,

$$\max_c \|M^\star(\overline{\mu}_c - \mu_c^\star)\|_2 \le \sqrt{\frac{2}{n\pi_{\min}}}\left(v + B\sqrt{2\log\frac{4K}{\delta}}\right), \tag{25}$$

$$\max_c \|M^\star(\overline{\mu}_c - \mu_c^\star)\|_1 \le \sqrt{\frac{2s_{\max}}{n\pi_{\min}}}\left(v + B\sqrt{2\log\frac{4K}{\delta}}\right). \tag{26}$$

Proof. Fix c and condition on the indicators $(\mathbf{1}\{c^\star(Z_i)=c\})_{i=1}^n$, with $N_c = r \ge 1$. The observations in I_c are then independent with distribution $P_0(\cdot \mid c^\star(Z)=c)$. Index them as $Z_{c,1},\dots,Z_{c,r}$ and set $Y_j = M^\star(Z_{c,j} - \mu_c^\star)$. They satisfy

$$\mathbb{E}Y_j = 0, \qquad \mathbb{E}\|Y_j\|_2^2 \le v^2, \qquad \|Y_j\|_2 \le B \quad \text{almost surely.}$$

For $F = \|r^{-1}\sum_j Y_j\|_2$, independence and centering give

$$\mathbb{E}F \le (\mathbb{E}F^2)^{1/2} = \left(\frac{1}{r^2}\sum_j \mathbb{E}\|Y_j\|_2^2\right)^{1/2} \le \frac{v}{\sqrt{r}}.$$

Replacing one Y_j changes F by at most $2B/r$. The bounded-differences (Mcdiarmid's) inequality therefore implies

$$\Pr\left(F > \frac{v}{\sqrt{r}} + t \,\middle|\, \text{membership indicators}\right) \le \exp\left(-\frac{rt^2}{2B^2}\right).$$

Take $t = B\sqrt{2\log(4K/\delta)/r}$. The bound holds for every membership pattern with $r \ge 1$, so it may be averaged over these patterns without conditioning on any empirical clustering event. A union bound over c and Lemma 1 cost at most $\delta/4 + \delta/4$. On their intersection, $r \ge n\pi_{\min}/2$, which proves (25).

If $j \in S^\star \setminus J_c$, then $Z_j = (\mu_c^\star)_j$ almost surely in cluster c. Hence $M^\star(\overline{\mu}_c - \mu_c^\star)$ is supported on J_c, simultaneously for every nonempty oracle cluster, with probability one. Cauchy–Schwarz gives

$$\|M^\star(\overline{\mu}_c - \mu_c^\star)\|_1 \le \sqrt{|J_c|}\,\|M^\star(\overline{\mu}_c - \mu_c^\star)\|_2,$$

and proves (26). □

I.2 The global frequency mask and the main theorem

Lemma 3. *Let* $\widehat{p}_j = n^{-1}\sum_i \mathbf{1}\{(Z_i)_j > 0\}$ *and suppose* $\gamma = p_{(k)} - p_{(k+1)} > 0$. *If* $n \ge 8\gamma^{-2}\log(4d_2/\delta)$, *then* $\widehat{M} = M^\star$ *with probability at least* $1-\delta/2$.

Proof. Hoeffding's inequality and a union bound give

$$\Pr\left(\max_{j\in[d_2]} |\widehat{p}_j - p_j| > \gamma/4\right) \le 2d_2 e^{-n\gamma^2/8} \le \delta/2.$$

On the complementary event, for every $j \in S^\star$ and $\ell \notin S^\star$, $\widehat{p}_j - \widehat{p}_\ell \ge p_j - p_\ell - \gamma/2 \ge \gamma/2 > 0$. Thus the selected supports agree. Independence between different coordinates is unnecessary; independence across observations suffices. □

Proof of Theorem 1. Intersect the events in Lemmas 2 and 3. Their joint probability is at least $1-\delta$. No independence between these events or the fitted K-means solution is required. For every c, the definition of ρ_n gives

$$\|M^\star(\widehat{\mu}_c - \mu_c^\star)\|_1 \le \|M^\star(\widehat{\mu}_c - \overline{\mu}_c)\|_1 + \|M^\star(\overline{\mu}_c - \mu_c^\star)\|_1 \le \varepsilon_n. \tag{27}$$

Consequently, for every z whose empirical and population assignments agree, the reverse triangle inequality implies

$$|\widehat{s}_n(z) - s^\star(z)| \le \|M^\star(\widehat{\mu}_{c^\star(z)} - \mu^\star_{c^\star(z)})\|_1 \le \varepsilon_n. \tag{28}$$

This statement is pointwise on the whole representation space, not only on the safe support. The use of local sparsity concerned the *centroid estimation error*, not the support of the test point.

Write $A_n = \{z : \widehat{c}_n(z) \neq c^\star(z)\}$. Outside A_n, disagreement between the two strict-threshold detectors implies $|s^\star(z) - \tau_q| \leq \varepsilon_n$. Conditional on the training sample, ε_n is fixed and therefore

$$P_0(\widehat{f}_{n,q} \neq f_q^\star \mid \mathcal{D}_n) \leq P_0(A_n \mid \mathcal{D}_n) + P_0(|s^\star(Z) - \tau_q| \leq \varepsilon_n) \leq \eta_{0,n} + L_0\varepsilon_n.$$

This proves the theorem. The last step remains valid for random ρ_n because anti-concentration holds for every $t \geq 0$. □

Meaning of the rate. The theorem is an oracle inequality with explicit clustering remainders. It applies to any measurable fitted centers and assignments. To conclude consistency, one must separately show or assume that ρ_n and $\eta_{0,n}$ vanish. In a sequence of problems, γ, $\pi_{\min}$, v, B, and L_0 must also be tracked. In particular, a small frequency gap can make mask identification expensive even when $\log d_2$ is small. For balanced clusters and bounded $L_0 v, L_0 B$, obtaining sampling error at most e requires n of order $K s_{\max}/e^2$, up to logarithms, in addition to mask recovery and clustering control. Multiplying all scores by a constant changes L_0 inversely, so score normalization alone cannot improve this rate.

I.3 Two elementary controls on the clustering remainders

These controls clarify the terms left explicit by the main theorem; they do not claim that local sparsity alone guarantees K-means recovery. Let

$$\eta_{\mathrm{tr},n} = \frac{1}{n}\sum_i \mathbf{1}\{\widehat{c}_n(Z_i) \neq c^\star(Z_i)\}.$$

Lemma 4. *Suppose each fitted center is the mean of the points assigned to it,* $\max_i \|M^\star Z_i\|_1 \leq H$, *and the count event* (24) *holds. If* $\eta_{\mathrm{tr},n} \leq \pi_{\min}/4$, *then*

$$\rho_n \leq \frac{8H\eta_{\mathrm{tr},n}}{\pi_{\min}}. \tag{29}$$

Moreover, if $a_n = \max_c \|\widehat{\mu}_c - \mu_c^\star\|_2$, *then for* $y \in \{0,1\}$,

$$P_y(\widehat{c}_n(Z) \neq c^\star(Z) \mid \mathcal{D}_n) \leq P_y(D_2(Z) - D_1(Z) \leq 2a_n), \tag{30}$$

where D_1, D_2 *are the smallest and second-smallest population-center distances. For* $K = 1$, *the assignment error is zero.*

Proof. Put $\widehat{I}_c = \{i : \widehat{c}_n(Z_i) = c\}$. For each c, $|I_c \triangle \widehat{I}_c| \leq n\eta_{\mathrm{tr},n}$, and hence $|\widehat{I}_c| \geq n\pi_{\min}/4$. For two index sets A, B of sizes $a, b > 0$ and vectors u_i of norm at most H, subtraction of their means gives

$$\left\| \frac{1}{b}\sum_{i\in B} u_i - \frac{1}{a}\sum_{i\in A} u_i \right\|_1 \leq \frac{H|A \triangle B|}{b} + \frac{H|a-b|}{b} \leq \frac{2H|A \triangle B|}{b}.$$

Apply this with $u_i = M^\star Z_i$ to obtain (29). For (30), each center distance changes by at most a_n. A gap greater than $2a_n$ therefore preserves the unique nearest center. Taking probabilities proves the claim. □

The quantity a_n is a *full-space* center error. The masked error in Theorem 1 does not control it. Thus a dimension dependence in full-space clustering cannot be removed merely by masking the score.

I.4 Masking: estimation complexity and population separation

Corollary 1. *Let* M *be a deterministic binary coordinate mask, fixed before drawing* $\mathcal{D}_n$, *and keep the same full-space centers and assignments. Define*

$$J_c(M) = \{j : M_{jj} = 1,\ \mathrm{Var}(Z_j \mid c^\star(Z) = c) > 0\}, \qquad s(M) = \max_c |J_c(M)|,$$
$$v(M)^2 = \max_c \mathbb{E}[\|M(Z - \mu_c^\star)\|_2^2 \mid c^\star(Z) = c],$$
$$B(M) = \max_c \operatorname*{ess\,sup}_{Z \mid c^\star(Z) = c} \|M(Z - \mu_c^\star)\|_2,$$

where $B(M) < \infty$, and put $\rho_n(M) = \max_c \|M(\widehat{\mu}_c - \overline{\mu}_c)\|_1$. Under the count condition, with probability at least $1 - \delta/2$,

$$\max_c \|M(\widehat{\mu}_c - \mu_c^\star)\|_1 \le \varepsilon_n(M) := \rho_n(M) + \sqrt{\frac{2s(M)}{n\pi_{\min}}}\left(v(M) + B(M)\sqrt{2\log\frac{4K}{\delta}}\right). \quad (31)$$

Let $s_M^\star, \widehat{s}_{n,M}$ be the corresponding scores and let $\tau_{M,q}$ be the population safe quantile. If $P_0(|s_M^\star(Z) - \tau_{M,q}| \le t) \le L_M t$, their detector disagreement is at most $\eta_{0,n} + L_M \varepsilon_n(M)$ on this event.

Proof. Apply the proof of Lemma 2 to M and use $\|u\|_1 \le \sqrt{s(M)}\|u\|_2$ on each $J_c(M)$. The triangle inequality adds $\rho_n(M)$. The decision bound follows from (28) and the stated margin condition for this particular score. No frequency gap or mask recovery is needed because the mask is fixed in advance. □

If M_1 retains a subset of the coordinates retained by M_2, then $s(M_1) \le s(M_2)$, $v(M_1) \le v(M_2)$, $B(M_1) \le B(M_2)$, and $\rho_n(M_1) \le \rho_n(M_2)$. Thus the envelope in (31) is monotone under removing coordinates, for fixed centers and matching. This is not a classification-dominance statement: the population score, threshold, and margin constant change with the mask, and useful information may be discarded. The unmasked case is $M = I_{d_2}$. The result holds for each fixed mask, not simultaneously for all data-selected masks without an additional argument.

What effective dimension does and does not imply. For $\Sigma_{c,M} = M\,\mathrm{Cov}(Z \mid c^\star(Z) = c)M$,

$$v(M)^2 = \max_c \mathrm{tr}(\Sigma_{c,M}).$$

For a nonzero positive-semidefinite matrix Σ, the PCA definition of d_{90} implies $\mathrm{tr}(\Sigma)/\|\Sigma\|_{\mathrm{op}} \le d_{90}(\Sigma)/0.9$. Indeed, the sum of the largest d_{90} eigenvalues is at least 90% of the trace and at most $d_{90}\|\Sigma\|_{\mathrm{op}}$. But d_{90} does not bound the absolute trace without an eigenvalue scale, and a rank-one covariance can involve every coordinate. Furthermore, Figure 3 reports empirical quantities, not a population concentration bound. We therefore use it as evidence about relative variance concentration, not as a verification of every hypothesis in Theorem 1.

The next example separates population-score quality from finite-sample estimation. It is a mathematical illustration, not a fitted model of the reported activations.

Proposition 2 (Weak sparse noise can obscure an unmasked score)**.** *There exist safe and unsafe distributions on $\mathbb{R}_+^{D+1}$ with $1 + O(\sqrt{D})$ nonzero coordinates per input with high probability such that the following holds for $K = k = 1$. The global population frequency mask retains a single coordinate and its centroid-distance score has AUROC one. The unmasked distance to the* known population safe centroid *has AUROC tending to $1/2$ as $D \to \infty$.*

Proof. Take $D \ge 9$, $p = D^{-1/2}$ and $b = D^{-1/8}$. Let $U_0 \sim \mathrm{Unif}[1,2]$, $U_1 \sim \mathrm{Unif}[3,4]$, and let $B_{y,j} \sim \mathrm{Bern}(p)$ be independent of these variables and of one another. Define, for $y \in \{0,1\}$,

$$Z_y = (U_y, bB_{y,1}, \ldots, bB_{y,D}).$$

The expected support size is $1 + Dp = 1 + \sqrt{D}$; a binomial Chernoff bound gives $\|Z_y\|_0 \le 1 + 2\sqrt{D}$ with probability at least $1 - e^{-\sqrt{D}/3}$. Each nuisance activation has amplitude $b \to 0$. The first safe coordinate has activation frequency one, whereas every other coordinate has frequency p. The global top-one mask therefore selects the first coordinate, with gap $1 - p$.

The population safe center is $\mu = (3/2, bp, \ldots, bp)$. The masked scores are $A_0 = |U_0 - 3/2| \in [0, 1/2]$ and $A_1 = |U_1 - 3/2| \in [3/2, 5/2]$, so the masked ranking is perfect. For $T_y = \sum_j B_{y,j} \sim \mathrm{Bin}(D, p)$, the unmasked scores are

$$S_y = A_y + b\{Dp + (1-2p)T_y\}.$$

Let $r_D = \sqrt{2Dp(1-p)}$. The standardized difference $(T_1 - T_0)/r_D$ converges to $N(0,1)$. This can be seen directly from its characteristic function:

$$\mathbb{E}\exp\left(it\frac{T_1 - T_0}{r_D}\right) = [1 + 2p(1-p)\{\cos(t/r_D) - 1\}]^D \longrightarrow e^{-t^2/2},$$

using $Dp(1-p) \to \infty$ and the expansion of cosine at zero. Since $b(1-2p)r_D \asymp D^{1/8} \to \infty$ and $A_1 - A_0$ is bounded,

$$\frac{S_1 - S_0}{b(1-2p)r_D} \implies N(0,1).$$

The score difference has no atom at zero because U_1 is continuous and independent of the other variables. Hence $\mathrm{AUROC}(S) = \Pr(S_1 > S_0) \to 1/2$. The class label has not changed the nuisance distribution; its increasing fluctuation alone obscures the bounded signal in the unmasked score. □

I.5 Independent safe-threshold calibration

Appendix D.2 reports a held-out ROC evaluation and states that no separate threshold-calibration step was performed. The following results concern a possible deployment procedure. They do not attribute an additional split to the experiments.

Proposition 3. *Condition on $\mathcal{D}_n$ and let $Z'_1, \dots, Z'_m \sim P_0^m$ be independent safe calibration points. Let $\widehat{F}_m$ be the empirical distribution function of $\widehat{s}_n(Z'_i)$, and define $\widehat{\tau}_q = \inf\{t : \widehat{F}_m(t) \geq 1-q\}$. For $u_m = \sqrt{\log(2/\delta_q)/(2m)}$, with conditional probability at least $1-\delta_q$,*

$$P_0(\widehat{s}_n(Z) > \widehat{\tau}_q \mid \mathcal{D}_n, Z'_1, \dots, Z'_m) \leq q + u_m. \tag{32}$$

The assertion allows atoms and uses the strict inequality in the detector.

Proof. Let $\widehat{F}(t) = P_0(\widehat{s}_n(Z) \leq t \mid \mathcal{D}_n)$. The Dvoretzky–Kiefer–Wolfowitz inequality gives $\sup_t |\widehat{F}_m(t) - \widehat{F}(t)| \leq u_m$ with the stated probability. Since $\widehat{F}_m(\widehat{\tau}_q) \geq 1-q$, $\widehat{F}(\widehat{\tau}_q) \geq 1-q-u_m$. Taking complements proves the result. □

False-positive control alone does not bound the displacement of the threshold or the unsafe false-negative rate. The next result states the additional quantile regularity needed for such a comparison.

Proposition 4. *Let $F^\star(t) = P_0(s^\star(Z) \leq t)$ and suppose, for some $\lambda_0, r_0 > 0$ and all $0 \leq u \leq r_0$,*

$$F^\star(\tau_q + u) \geq 1 - q + \lambda_0 u, \qquad F^\star(\tau_q - u) \leq 1 - q - \lambda_0 u. \tag{33}$$

On the training event of Theorem 1, put $a_m = 2(\eta_{0,n} + u_m)/\lambda_0$ and assume $a_m \leq r_0$. With conditional calibration probability at least $1-\delta_q$,

$$|\widehat{\tau}_q - \tau_q| \leq \varepsilon_n + a_m, \tag{34}$$

$$P_0(\mathbf{1}\{\widehat{s}_n(Z) > \widehat{\tau}_q\} \neq f_q^\star(Z) \mid \mathcal{D}_n, Z'_1, \dots, Z'_m) \leq \eta_{0,n} + L_0(2\varepsilon_n + a_m). \tag{35}$$

Proof. The deterministic score comparison (28) shows, for every t,

$$F^\star(t - \varepsilon_n) - \eta_{0,n} \leq \widehat{F}(t) \leq F^\star(t + \varepsilon_n) + \eta_{0,n}.$$

On the DKW event, set $w = \eta_{0,n} + u_m > 0$. Applying (33) at $u = a_m$ yields

$$\widehat{F}_m(\tau_q + \varepsilon_n + a_m) \geq 1 - q + w, \qquad \widehat{F}_m(\tau_q - \varepsilon_n - a_m) \leq 1 - q - w.$$

The empirical quantile lies between these two arguments, proving (34). Outside changed assignments,

$$|(\widehat{s}_n(z) - \widehat{\tau}_q) - (s^\star(z) - \tau_q)| \leq \varepsilon_n + |\widehat{\tau}_q - \tau_q| \leq 2\varepsilon_n + a_m.$$

The same threshold-band argument as in Theorem 1 proves (35). □

I.6 Training and held-out cluster occupancy

Lemma 5. *For r independent observations in $[K]$ with common law p and empirical law $\widehat{p}_r$, with probability at least $1-\delta$,*

$$\|\widehat{p}_r - p\|_1 \leq b_r(\delta) := \sqrt{K/r} + \sqrt{2\log(1/\delta)/r}. \tag{36}$$

Proof. Cauchy–Schwarz and the multinomial variances give

$$\mathbb{E}\|\widehat{p}_r - p\|_1 \leq \sqrt{K\,\mathbb{E}\|\widehat{p}_r - p\|_2^2} = \sqrt{\frac{K}{r}\left(1 - \sum_c p_c^2\right)} \leq \sqrt{K/r}.$$

Changing one observation changes the ℓ_1 deviation by at most $2/r$. The bounded-differences inequality therefore bounds its upper tail above the mean by $\exp(-rt^2/2)$, proving the claim. □

Proposition 5. *Let $\widehat{\pi}_{\mathrm{tr}}$ be the fitted-cluster histogram of $\mathcal{D}_n$ and let $\widehat{\pi}_{\mathrm{ho}}$ be the corresponding histogram of m independent held-out safe observations. With probability at least $1-\delta$ over both samples,*

$$\|\widehat{\pi}_{\mathrm{ho}} - \widehat{\pi}_{\mathrm{tr}}\|_1 \le 2\eta_{\mathrm{tr},n} + 2\eta_{0,n} + b_n(\delta/2) + b_m(\delta/2). \tag{37}$$

Here the assignment-error quantities are their actual sample-dependent values. The result requires neither mask recovery nor small centroid error.

Proof. Let $\pi_c = P_0(c^\star(Z) = c)$, let $\widehat{\pi}^\star_{\mathrm{tr},c} = n^{-1}\sum_i \mathbf{1}\{c^\star(Z_i) = c\}$, and let $\widetilde{\pi}_{0,c} = P_0(\widehat{c}_n(Z) = c \mid \mathcal{D}_n)$. Changing one assignment moves empirical mass between at most two cells, so

$$\|\widehat{\pi}_{\mathrm{tr}} - \widehat{\pi}^\star_{\mathrm{tr}}\|_1 \le 2\eta_{\mathrm{tr},n}.$$

Coupling $c^\star(Z)$ and $\widehat{c}_n(Z)$ with the same fresh Z gives $\|\widetilde{\pi}_0 - \pi\|_1 \le 2\eta_{0,n}$. Since the population partition is fixed, Lemma 5 applies to the oracle training histogram. Conditional on $\mathcal{D}_n$, it also applies to the held-out histogram with law $\widetilde{\pi}_0$. These two concentration events cost $\delta/2$ each; their intersection satisfies

$$\|\widehat{\pi}^\star_{\mathrm{tr}} - \pi\|_1 \le b_n(\delta/2), \qquad \|\widehat{\pi}_{\mathrm{ho}} - \widetilde{\pi}_0\|_1 \le b_m(\delta/2).$$

The triangle inequality proves the result. In particular, we have not incorrectly treated the fitted training histogram as multinomial conditional on a partition learned from those same observations. □

Similar occupancy does not imply similar partitions: distinct partitions may have identical cell probabilities. Figure 4 is therefore an empirical check of one aspect of generalization, not a converse to Proposition 5 or a measurement of $\eta_{0,n}$.

I.7 Optimal readout and the information retained by the detector

Condition on the fitted detector throughout this subsection. Let $\mathcal{R}_\alpha(f) = \alpha P_0(f(Z) = 1) + (1-\alpha)P_1(f(Z) = 0)$ for $\alpha \in (0,1)$. Define $\mathcal{R}_{\mathrm{Bayes}}$ as the infimum over measurable classifiers of Z, $\mathcal{R}^\star_{\mathrm{cs}}$ as the infimum over measurable functions of $(\widehat{c}_n(Z), \widehat{s}_n(Z))$, and $\mathcal{R}^\star_{\mathrm{cl}}$ as the infimum over functions of $\widehat{c}_n(Z)$ alone. These are population oracle risks, not risks of fitted calibration models.

Proposition 6. *With $\widetilde{\pi}_{y,c} = P_y(\widehat{c}_n(Z) = c \mid \mathcal{D}_n)$,*

$$\mathcal{R}^\star_{\mathrm{cl}} = \sum_c \min\{\alpha\widetilde{\pi}_{0,c}, (1-\alpha)\widetilde{\pi}_{1,c}\} \tag{38}$$

$$= \frac{1 - \|\alpha\widetilde{\pi}_0 - (1-\alpha)\widetilde{\pi}_1\|_1}{2}. \tag{39}$$

For equal priors this becomes

$$\mathcal{R}^\star_{\mathrm{cl}} = \frac{1}{2}(1 - \mathrm{TV}(\widetilde{\pi}_0, \widetilde{\pi}_1)), \qquad \mathrm{TV}(p,q) = \tfrac{1}{2}\|p-q\|_1. \tag{40}$$

Let $\mathcal{R}^\star_{\mathrm{aff}}$ be the infimum over classifiers $\mathbf{1}\{a_{\widehat{c}_n(Z)}\widehat{s}_n(Z) + b_{\widehat{c}_n(Z)} > 0\}$ with arbitrary real coefficients. Then

$$\mathcal{R}_{\mathrm{Bayes}} \le \mathcal{R}^\star_{\mathrm{cs}} \le \mathcal{R}^\star_{\mathrm{aff}} \le \mathcal{R}^\star_{\mathrm{cl}}. \tag{41}$$

In particular, $0 \le \mathcal{R}^\star_{\mathrm{cs}} - \mathcal{R}_{\mathrm{Bayes}} \le \mathcal{R}^\star_{\mathrm{cl}}$.

Proof. Declaring cluster c unsafe incurs error $\alpha\widetilde{\pi}_{0,c}$, whereas declaring it safe incurs error $(1-\alpha)\widetilde{\pi}_{1,c}$. Choosing the better label separately in each cluster gives (38). The identity $\min(a,b) = (a+b-|a-b|)/2$ and the normalization of the two occupancy vectors give (39) and (40).

Every cluster-and-score classifier is a measurable classifier of Z. Every cluster-affine classifier is a cluster-and-score classifier. Finally, setting $a_c = 0$ and $b_c \in \{-1, 1\}$ realizes any constant label in each cluster. These inclusions prove the risk ordering. □

For a fitted affine-threshold classifier $\widehat{g}$, the excess risk is exactly

$$\mathcal{R}_\alpha(\widehat{g}) - \mathcal{R}_{\mathrm{Bayes}} = [\mathcal{R}_\alpha(\widehat{g}) - \mathcal{R}^\star_{\mathrm{aff}}] + [\mathcal{R}^\star_{\mathrm{aff}} - \mathcal{R}^\star_{\mathrm{cs}}] + [\mathcal{R}^\star_{\mathrm{cs}} - \mathcal{R}_{\mathrm{Bayes}}]. \tag{42}$$

The three terms are nonnegative: fitting the readout, restricting it to cluster-affine maps, and losing information in the cluster-and-score representation. The occupancy formula bounds the last term, not the first two. The supervised probe is another fitted classifier, not the Bayes rule. Similar measured AUROCs therefore provide empirical evidence but not a bound on these classification-risk gaps.

Proposition 7. *Condition on $\mathcal{D}_n$. For $y \in \{0,1\}$, let $Z_{y,1}, \dots, Z_{y,m_y} \sim P_y^{m_y}$ be independent evaluation samples not used to fit or select the detector. Let $\widehat{\pi}_y^{\mathrm{ev}}$ be their fitted-cluster histograms and put*

$$\widehat{\mathcal{R}}_{\mathrm{cl}} = \sum_c \min\{\alpha \widehat{\pi}^{\mathrm{ev}}_{0,c}, (1-\alpha)\widehat{\pi}^{\mathrm{ev}}_{1,c}\}.$$

With conditional probability at least $1-\delta$,

$$|\widehat{\mathcal{R}}_{\mathrm{cl}} - \mathcal{R}^\star_{\mathrm{cl}}| \le \tfrac{1}{2}\{\alpha b_{m_0}(\delta/2) + (1-\alpha) b_{m_1}(\delta/2)\}. \tag{43}$$

Proof. Use (39) for both normalized histogram pairs. The reverse triangle inequality bounds the difference by

$$\tfrac{1}{2}\{\alpha \|\widehat{\pi}_0^{\mathrm{ev}} - \widetilde{\pi}_0\|_1 + (1-\alpha)\|\widehat{\pi}_1^{\mathrm{ev}} - \widetilde{\pi}_1\|_1\}.$$

Apply Lemma 5 to each independent evaluation sample with failure probability $\delta/2$. □

Figure 4 overlays each evaluation histogram with the *training* histogram. The overlap formula instead compares safe and unsafe *population* occupancies under the same fitted partition; the preceding proposition estimates them with independent safe and unsafe evaluations. The training overlay must not be substituted without accounting for its data dependence. For the benign-OOD evaluation, the same formulas apply after replacing P_0 in the risk by the evaluation's mixture of in-distribution safe and benign-OOD data. A conclusion for one negative distribution does not automatically transfer to the other.

I.8 Generalization on a safe–unsafe mixture

Corollary 2. *On the event of Theorem 1, define $\eta_{1,n} = P_1(\widehat{c}_n(Z) \neq c^\star(Z) \mid \mathcal{D}_n)$. Suppose $P_1(|s^\star(Z) - \tau_q| \le t) \le L_1 t$ for all $t \ge 0$. Then*

$$|\mathcal{R}_\alpha(\widehat{f}_{n,q}) - \mathcal{R}_\alpha(f^\star_q)| \le \alpha\eta_{0,n} + (1-\alpha)\eta_{1,n} + \{\alpha L_0 + (1-\alpha)L_1\}\varepsilon_n. \tag{44}$$

For the independently calibrated detector of Proposition 4, the same bound holds with ε_n replaced by $2\varepsilon_n + a_m$ on its calibration event.

Proof. The score comparison (28) holds for any test point. The threshold-band argument therefore bounds disagreement under P_y by $\eta_{y,n} + L_y\varepsilon_n$, for $y = 0, 1$. For binary classifiers f, g, the absolute difference in either class-conditional error is at most $P_y(f \neq g)$. Weight by the class priors and add. The calibrated statement follows from the last display in the proof of Proposition 4. □

Consequently, total excess risk is bounded by the estimation term in (44) plus $\mathcal{R}_\alpha(f^\star_q) - \mathcal{R}_{\mathrm{Bayes}}$. The latter includes the choice of a global threshold and the false-positive budget q; it is not solely representation loss. In fact, (10) excludes an atom at τ_q, so $P_0(s^\star(Z) > \tau_q) = q$ and $\mathcal{R}_\alpha(f^\star_q) \ge \alpha q$. For fixed $q > 0$, vanishing estimation error therefore need not imply vanishing safety-classification error.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and Section 1 state three contributions: a one-class framing for LLM safety, a theory and modular framework based on local low-dimensional structure, and empirical evidence across six model families. These are supported by Sections 2, 6, 3, and 4. The introduction notes that unsupervised methods are not proposed as replacements for supervised ones.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 5 discusses two main limitations: (i) the AUROC gap to the supervised reference, framed as the cost of dropping the assumption that the unsafe distribution is known in advance, and (ii) the sensitivity of one-class detectors to the safe training distribution, which can cause benign inputs from a shifted distribution to be flagged as atypical regardless of whether they are unsafe. For the second limitation the section also reports a measured mitigation, per-cluster calibration from a small coupled labeled budget, together with the operating point at which it is read and an explicit statement that its AUROC is not comparable to Table 1; Appendix B gives the estimator and the full budget sweep. Appendix A states that we do not advocate unsupervised methods as standalone replacements for supervised safety systems.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.

- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Answer: [Yes]

   Justification: All theoretical results (the generalization bound underlying the framework's tractability claim) are stated with their assumptions in Section 6, and full proofs are provided in Appendix I.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Section 4.1 specifies models, layers, and the safe/unsafe data composition. Appendix D provides SAE training details (Appendix D.1), dataset composition and splits (Appendix D.2), per-method hyperparameter sweeps and selected values (Appendix D.4), and compute budget (Appendix D.5). Algorithms 1 and 2 give pseudocode for the two main framework instantiations.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.

(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The implementation and evaluation scripts are publicly available at `https://github.com/lasgroup/unsupervised-llm-safety`. The repository includes installation and pipeline instructions, configuration files, and per-sample reference scores with a script to recompute reported metrics. Public datasets and open-weight models are cited in the paper; experimental details are given in Appendix D.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Data splits, optimizer choices, hyperparameter sweep ranges, selected values, and threshold-calibration procedure are specified in Appendix D, with per-method details in Appendix D.4. The main paper (Section 4.1) summarizes the setup at the level needed to interpret the results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.

- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The main results in Table 1 report mean and standard deviation across five random seeds per configuration.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix D.5 reports the compute setup and approximate per-run wall-clock times for the experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conforms with the NeurIPS Code of Ethics. We use only publicly released datasets and open-weight base models, do not collect new data from human subjects, and the work's stated goal is improved detection of unsafe model inputs.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Broader impacts are discussed in Appendix A.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release new pre-trained models, generative systems, or scraped datasets. All base models and evaluation datasets are existing public assets, and the contribution is a detection algorithm that is not itself a high-misuse-risk artifact.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All datasets and models used are publicly released and are cited at first mention in Section 4.1. We include the license of these datasets in Table 8.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The released code and per-sample reference scores are documented at `https://github.com/lasgroup/unsupervised-llm-safety`, with setup instructions, evaluation scripts, and an MIT license. The README notes that upstream corpora used to compute the scores retain their own terms. The paper does not release a new dataset or pre-trained model.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or research with human subjects. All evaluations are automated on existing public datasets and model outputs.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve research with human subjects, so IRB review does not apply.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs are the object of study (we evaluate detection methods on LLM activations) but are not used as a non-standard component of the core methodology beyond standard inference. Any LLM use for writing or editing assistance falls outside the declaration requirement per the question's stated exception.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.