MODERISKER: MODE-LEVEL RISK MINIMIZATION FOR SCENARIO DELIVERY IN PROBABILISTIC FORECASTING

Anonymous authors
under review

ABSTRACT

Consumers of probabilistic forecasts receive K representative scenarios rather than the full predictive distribution, and these scenarios are almost always chosen to optimize *average* coverage. We show that average-optimal selection undergoes a sharp phase transition: once a mode's probability π falls below a threshold set by K, the mode is omitted *entirely*, even though the sample pool still contains it — and rare futures are the reason scenario sets exist. We therefore formalize scenario delivery as mode-level risk minimization: CVaR_β over per-mode coverage losses. This objective is not submodular, and the natural greedy provably stalls; linearizing at the CVaR supergradient restores near-optimal selection at the same cost. We evaluate with Realized Tail Coverage (RTC_β), the CVaR over test instances of the best-of-K distance, under a protocol that stratifies by multimodality and matches β to the mode mass, on bifurcation processes and on three real domains with three generator families. Average-greedy delivery collapses from 0.83 to 0.07 as π drops from 0.02 to 0.01 while pool containment stays at 1.0; ModeRisker tracks the containment bound down to $\pi{=}2.5/N$ and improves stratified $\mathrm{RTC}_{0.01}$ by 39% at no average cost. The bottleneck is mode *discovery*, not selection: cluster-quality metrics mislead, and an outlier-rescue layer restores near-oracle delivery. Real data delimits the scope: selection pays only where a rare mode is in the pool and the generator is calibrated, and a modality screen keeps the machinery inert elsewhere, which makes the layer safe to deploy by default. Code is available at `https://anonymous.4open.science/r/moderisker-CD82`.

1 INTRODUCTION

Operational forecasting pipelines compress a predictive distribution into K scenarios: power-system operators schedule against scenario sets (Høyland & Wallace, 2001; Pinson et al., 2009), and trajectory-prediction modules hand K futures to a planner (Alahi et al., 2016; Chang et al., 2019). The selection step is almost universally average-optimal: choose S with $|S| = K$ to minimize the expected distance from a random future to its nearest scenario (facility location or K-medoids on a sample pool, or their soft variants).

The omission phase transition. Average optimality is the wrong objective for the tail. Consider a mixture of future modes with weights π_c. The marginal value of covering mode c scales with π_c. Whenever refining an *already-covered* mode gains more than π_c times the gain of covering mode c for the first time, the average-optimal set spends its entire budget refining large modes. The result is a sharp threshold in π, set by K and the mode geometry, below which the small mode is delivered with probability near zero *even though the pool contains it*: whenever $\pi \gtrsim 1/N$ for pool size N, the failure lies in selection, not sampling. Empirically the transition is abrupt (§4): delivery falls from ≈ 1 to ≈ 0 within a factor of 2–4 in π.

Mode-level risk. We replace the average with a coherent risk measure (Artzner et al., 1999) over *modes*, the conditional value-at-risk (Rockafellar & Uryasev, 2000):

$$\min_{|S|=K} \mathrm{CVaR}_\beta\big(\ell(S,1),\dots,\ell(S,C);\,\pi\big), \qquad \ell(S,c) = \mathbb{E}_{y\sim P_c} \min_{s\in S} d(y,s), \tag{1}$$

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

the mean of the worst β-mass of per-mode coverage losses under mode weights π. Setting $\beta{=}1$ recovers the average objective, and $\beta \to 0$ approaches max-min coverage. In between, β is the level at which "no β-mass of futures is left unrepresented" becomes a contract.

Optimization: the naive greedy provably stalls. CVaR aggregation destroys the submodularity (Nemhauser et al., 1978; Krause & Golovin, 2014) that justifies greedy facility location. Concretely, when two uncovered modes have near-tied worst losses, covering either one leaves the CVaR unchanged ($\mathrm{CVaR}_\beta = \max$ when $\beta \le \min_c \pi_c$). A greedy that scores candidates by the *resulting* risk therefore sees a flat landscape and stalls: in our ablation (§4) it keeps refining the middle mode while an uncovered small mode holds the max. Our fix linearizes the risk at the current loss vector. We compute the CVaR supergradient weights $q^\star$, the exact maximizer of the dual representation ($q_c^\star = \min(\pi_c, \text{remaining } \beta)/\beta$ down the sorted losses), and greedily add the candidate with the largest $q^\star$-weighted marginal gain. This keeps greedy's cost and restores delivery to the containment bound.

Evaluation: making omission visible. Standard probabilistic metrics such as CRPS and the energy score (Matheson & Winkler, 1976; Gneiting & Raftery, 2007) are average-type and provably insensitive to a missing $O(\pi)$ mode. We evaluate with Realized Tail Coverage, $\mathrm{RTC}_\beta@K$: the CVaR_β over test instances of the realized best-of-K distance. Two protocol elements are essential, and we argue they are of independent value: (i) *β must match the target mode mass*: if $\beta \gg \pi_{\text{small}}$, between-instance difficulty dominates the tail statistic and drowns the omission signal; (ii) *results must be stratified by multimodality*: on instances with a unimodal predictive distribution all reasonable selectors coincide, so pooling dilutes any comparison. A simple two-means separation-ratio screen isolates the instances where selection can matter.

Contributions. (1) The omission phase transition of average-optimal scenario selection, with its scaling in K and pool size. (2) ModeRisker: mode-level CVaR_β selection with a supergradient-linearized greedy, together with the negative result that motivates it. The naive CVaR greedy stalls; kept as an ablation, its stratified RTC is $1.5\times$ ModeRisker's at every β, and at the average end it falls behind even *random* selection. (3) The stratified RTC_β evaluation protocol with β–π matching. (4) The mode-discovery bottleneck: end-to-end delivery decouples from cluster-recovery metrics, because finer clustering fragments the mode weights that the CVaR budget operates on and can halve delivery while improving recovery. An outlier-rescue inference layer restores near-oracle delivery. (5) A scope taxonomy from three real domains — electricity (rare, in-support: selection pays), traffic (in-support, not rare: nothing to fix) and retail (rare, out-of-support: no selector can help) — with a modality screen that falls back to average-greedy on unimodal instances.

2 Related work

Scenario generation and reduction. In the stochastic-programming tradition (Shapiro et al., 2021), scenario trees are built by moment matching (Høyland & Wallace, 2001) and thinned by Wasserstein reduction (Dupačová et al., 2003; Heitsch & Römisch, 2003); wind-power scenario pipelines are the canonical energy-sector instance (Pinson et al., 2009). All of these minimize an *average* distributional distance. Our objective is instead a risk measure over modes, and our phase-transition analysis explains when reduction drops rare regimes. Our average-greedy baseline *is* Heitsch–Römisch forward selection under the pool's empirical measure, so the comparison is built in. **Probabilistic forecasters as scenario sources.** Our pools come from autoregressive density heads (Salinas et al., 2020), mixture-density outputs (Bishop, 1994), conditioned normalizing flows (Rasul et al., 2021), quantile-regression heads (Koenker & Bassett, 1978; Wen et al., 2017) and zero-shot foundation forecasters (Ansari et al., 2024). All are trained and scored on distributional fidelity; none prescribes *which* K paths to hand downstream, which is the layer we study. The two concerns are not fully separable, however: §5(2) shows that a miscalibrated generator defeats any selector. **Diverse K-best / DPP sampling.** Determinantal selection (Kulesza & Taskar, 2012) promotes spread but has no notion of mode mass or risk level. Our k-DPP baseline quantifies the failure: it attains the delivery bound at a 57% penalty in average quality and a 40% penalty in RTC (Fig. 1), because diversity is a geometric proxy that spends budget on unrepresentative extremes. **Multiple-choice / winner-takes-all learning.** MCL/WTA (Guzmán-Rivera et al., 2012; Rupprecht et al., 2017) train K heads to partition the target distribution and inherit the average objective across heads. Our layer

Algorithm 1 ModeRisker selection

Input: pool P, labels, weights π, budget K, level β
$m_i \leftarrow$ large init for all pool points i (uncovered level)
for $t = 1..K$ **do**
  $\ell_c \leftarrow$ mean over mode-c points of m; $q^\star \leftarrow \arg\max_{q \in \Delta_\beta} q^\top \ell$ (CVaR dual maximizer)
  add s_j maximizing $\sum_c q_c^\star \big(\ell_c - \ell_c(m \wedge d_{\cdot j})\big)$; $m \leftarrow m \wedge d_{\cdot j}$
end for

is post-hoc over any generator's pool and carries an explicit risk contract. **Risk-averse optimization and submodularity.** Taking the CVaR of set functions is known to break the greedy guarantees of submodular maximization (Nemhauser et al., 1978; Krause & Golovin, 2014); SATURATE-style bisection (Krause et al., 2008) gives guarantees for max-min. Distributionally robust learning (Levy et al., 2020) optimizes a comparable tail objective over training *data*; ours is over the modes of a single predictive distribution, where coherence (Artzner et al., 1999) is what turns the level β into a deliverable contract. Our supergradient greedy is a practical middle ground: exact dual weights (Proposition 1), greedy cost, and an explicit perturbation bound under mode misestimation (Proposition 3). Empirically it is within 3% of exhaustive swap local search and beats the SATURATE endpoint on RTC in 25 of 27 budget–rarity cells (§4). **Probabilistic forecast evaluation.** The CRPS, energy and variogram scores (Matheson & Winkler, 1976; Gneiting & Raftery, 2007; Scheuerer & Hamill, 2015) are strictly proper for the full distribution but average-type over instances. RTC_β departs from them deliberately: it is a task-level tail statistic for the delivered set. Best-of-K (minADE/minFDE) is standard in trajectory prediction (Alahi et al., 2016; Gupta et al., 2018; Salzmann et al., 2020; Chang et al., 2019) but is reported as a mean; we add the CVaR-over-instances reading and the β–π matching requirement.

3 Mode-level risk minimization

3.1 Setting and objective

For each instance, a generator (any probabilistic forecaster) provides a pool $P = \{s_1, \ldots, s_N\}$ of sampled futures, and a selector returns $S \subset P$ with $|S| = K$. Mode labels and weights $(c(s), \pi)$ are oracle on synthetic data and inferred in deployment. Given them, the per-mode loss and the objective are those of equation 1, with d the trajectory RMSE.

3.2 Supergradient-linearized greedy

The dual feasible set is $\Delta_\beta = \{0 \le q \le \pi/\beta,\ \sum q = 1\}$. Its maximizer at the current losses is computed exactly in $O(C \log C)$: sort the modes by loss, then waterfill β mass.

Proposition 1 (Exact supergradient)**.** *For $\beta \in (0, 1]$ and loss vector ℓ, sort modes by decreasing ℓ and set $q_c = \min(\pi_c, \beta - \sum_{c' \prec c} \pi_{c'})_+ / \beta$. Then $q \in \arg\max_{\Delta_\beta} q^\top \ell$, i.e. $q^\top \ell = \mathrm{CVaR}_\beta(\ell; \pi)$, and q is a supergradient of the concave function $\ell \mapsto \mathrm{CVaR}_\beta(\ell; \pi)$ at ℓ.*

Proposition 2 (The naive greedy stalls)**.** *Let $\beta \le \min_c \pi_c$, so $\mathrm{CVaR}_\beta(\ell) = \max_c \ell_c$. Suppose two uncovered modes a, b have $\ell_a = \ell_b = \max_c \ell_c$ and no single candidate reduces both. Then every candidate leaves the* CVaR *unchanged, so the naive greedy's score is constant over candidates, and its (arbitrary) tie-break can spend the entire remaining budget without ever covering a or b. The linearized rule instead assigns $q_a^\star, q_b^\star > 0$ and strictly prefers candidates that cover either mode.*

Proposition 2 is not a corner case: near-tied uncovered modes arise systematically at initialization, when all uncovered modes sit at the same "uncovered" loss level. Empirically, the naive rule underperforms even *random* selection at the average end (§4).

Proposition 3 (Sensitivity to mode-weight error)**.** *For any losses ℓ and weight vectors $\pi, \hat{\pi}$ on the same modes,*

$$\big|\mathrm{CVaR}_\beta(\ell; \pi) - \mathrm{CVaR}_\beta(\ell; \hat{\pi})\big| \ \le\ \frac{\|\pi - \hat{\pi}\|_1}{\beta} \big(\max_c \ell_c - \min_c \ell_c\big).$$

The $1/\beta$ factor formalizes two empirical regularities. First, risk levels matched to small mode masses are the most sensitive to mode-weight misestimation. Second, *overclustering is harmful even when it finds the rare cluster*: fragmenting a large mode inflates $\|\pi - \hat{\pi}\|_1$ at the granularity the CVaR budget operates on, so fake fragments with $\hat{\pi} \approx \beta$ crowd the true rare mode out of the tail. We observe this as the recovery–delivery decoupling of §3.3. Proofs are in Appendix A.

3.3 Mode discovery: the actual bottleneck

In deployment, labels and weights are inferred from the pool itself. The k-means variance objective (Lloyd, 1982; Arthur & Vassilvitskii, 2007) is structurally blind to clusters of a few distant samples. More damaging, *overclustering misallocates the risk budget*: splitting a large mode into fragments with $\hat{\pi} \approx \beta$ each fills the CVaR tail with fake modes and starves the true rare one. Label metrics, whether cluster recovery or silhouette-style validity indices (Rousseeuw, 1987), therefore decouple from delivery. On our benchmark, 8-means recovers the rare cluster more than twice as often as 5-means (0.41 vs 0.15 at π=0.005) yet delivers it substantially *less* often. The fragments (empirical $\hat{\pi} \in [0.10, 0.22]$) absorb ≈ 0.9 of the supergradient's β budget at every step, leaving the true rare cluster a residual share whose weighted marginal gain never wins a greedy step (cf. Proposition 3). We therefore infer modes by *outlier rescue*: run k-means on the main modes, then regroup samples whose residual exceeds a multiple of the within-cluster RMS into candidate rare modes. Rescue restores delivery to 0.99 at π=0.01 (vs 0.57 for 5-means) and to 0.70 at π=0.005, against a pool-containment ceiling of 0.91. The full ladder, including an EM-fitted GMM baseline (Dempster et al., 1977) that matches k-means, is in §4. An independent re-implementation reproduced all benchmark numbers.

3.4 Evaluation protocol: RTC_β with stratification

$\mathrm{RTC}_\beta@K$ is the CVaR_β over test instances of $\min_{s \in S} d(y, s)$, where y is the realized future. The protocol requires (i) β matched to the mode mass one claims to protect ($\beta \sim \pi_{\text{small}}$), and (ii) stratification by a modality screen (a two-means separation ratio with threshold τ), since selection can matter only on multimodal instances. We report the full β spectrum: improvements must appear for $\beta \lesssim \pi_{\text{small}}$ and vanish smoothly as $\beta \to 1$, which checks that the average cost is zero.

4 Synthetic experiments: the phase transition

Controlled bifurcation processes (C branches, GP noise, one late-bifurcating rare branch) provide oracle modes and a realized-future distribution; pools have N=500 samples.

Phase diagram (Fig. 1). With K=5, average-greedy delivery of the rare mode collapses from 0.83 to 0.07 as π drops from 0.02 to 0.01, while pool containment stays at 1.0: the failure is in selection, not sampling, and the threshold moves with K as predicted. ModeRisker with oracle modes instead tracks the pool-containment bound down to $\pi = 0.005 = 2.5/N$ (delivery 0.97, containment 0.97), well beyond the average-greedy cliff. Below that, two limits compound. Containment itself drops, and, by design, a $\mathrm{CVaR}_{0.1}$ objective stops demanding coverage of a mode whose supergradient weight π/β can no longer win a greedy step (at $\pi = 0.002$: containment 0.63, oracle delivery 0.13). Protecting a rare mode below the risk level β therefore requires changing the contract, not tuning the method. (Rescue inference delivers 0.27 there: its `min_size`=2 reclustering acts as an implicit floor of $2/N$ on $\hat{\pi}$, which inflates sub-β modes just enough to compete and thus marks the same β–π boundary.) The oracle top-density selector is the *worst* baseline throughout, because likelihood concentrates the budget on the core of the dominant mode.

Pool-size scaling. The cliff is a property of the selection *rule*, not of sampling. We repeat the sweep at $N \in \{125, 250, 500, 1000, 2000\}$ (30 seeds each), a $16\times$ range over which the containment ceiling changes by a factor of 2. The average-greedy transition stays at $\pi \approx 0.02 \to 0.01$ for every N: delivery is 0.63–0.97 at π=0.02 and collapses to 0.07–0.20 at π=0.01. ModeRisker's delivery instead equals the N-dependent containment ceiling *point for point* down to $\pi = 0.005$ at every pool size (N=125: 0.73 at π=0.01 and 0.40 at π=0.005, both exactly the containment values; N=2000: 1.00 vs 1.00 at π=0.005). A gap that does not depend on containment opens only at $\pi = 0.002 \ll \beta$ (0.40 vs a ceiling of 0.90 at N=1000; 0.13 vs 0.93 at N=2000). That gap is governed by the risk

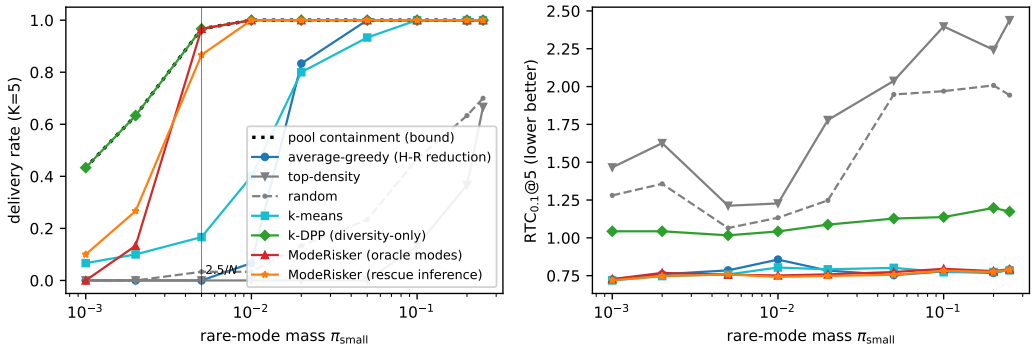

Figure 1: **Left:** rare-mode delivery vs. π_{small} (K=5, 30 seeds, N=500). Average-greedy collapses at $\pi \approx 0.01$ while the pool still contains the mode; ModeRisker tracks the containment bound to $2.5/N$; diversity-only k-DPP sits *on* the bound at every π. **Right:** the price of that diversity, as stratum $\text{RTC}_{0.1}$: DPP is 35–45% worse than ModeRisker throughout. Only the mode-risk selectors occupy both frontiers.

Table 1: **The selection-rule ladder.** Selectors of §4 on identical pools (N=500, 30 seeds, $\pi \in [0.001, 0.25]$, $K \in \{3, 5, 10\}$). Delivery is rare-mode delivery at K=5; $\text{RTC}_{0.5}$ is on the multimodal stratum, where ModeRisker (rescue) attains $0.81/0.75/0.62/0.55$ at β=0.05/0.1/0.3/0.5. "—" marks a quantity not reported for that selector.

Selector	Rare-mode delivery	$\text{RTC}_{0.5}$	Relation to ModeRisker
Random	—	0.79	dominated at every β
Oracle top-density	—	—	worst baseline throughout
Kernel herding	collapses at π=0.1	—	fails delivery outright
Average-greedy	$0.83 \to 0.07$ as π crosses $0.02 \to 0.01$	—	ModeRisker +39% at β=0.01
k-DPP (diversity)	= containment at every π	0.87	1.05 vs 0.75 at β=0.1; 57% worse average quality
Proportional quota	0 seats for $\pi \leq 0.05$	—	fails delivery outright
Round-robin quota	0.98 (stratum)	—	1.5–2.0× RTC at every $\beta \leq 0.2$
SATURATE (max-min)	= containment at every π	—	+0.4% to +4.1% RTC; worse in 25 of 27 cells
Swap local search	—	—	brackets the optimum; ModeRisker within 3.5%
Naive CVaR greedy	—	0.86	1.5× RTC at every β; stalls (Prop. 2)
ModeRisker, oracle modes	0.97 at π=0.005 (= containment)	—	—
k-means modes	0.57 / 0.24 at π=0.01 / 0.005	—	overclustering crowds the CVaR budget
outlier rescue	0.99 / 0.70 at π=0.01 / 0.005	0.55	best RTC at every β, and needs no labels

level and the budget, not by N: more sampling cannot buy back a mode that the CVaR_β contract has stopped pricing.

The selection-rule ladder. Thirteen selectors on identical pools ($K \in \{3, 5, 10\}$, N=500, 30 seeds, $\pi \in [0.001, 0.25]$; delivery reported at K=5) separate four hypotheses about what delivery requires (Table 1). (i) *Mode-blind average-type rules* fail early. Kernel herding (Chen et al., 2010) collapses at π=0.1 and average-greedy at π=0.01: greedy energy-distance matching is an average objective and inherits the phase transition. (ii) *Mode-aware but risk-blind quotas* fail in one direction or the other. A proportional (largest-remainder) quota gives the rare mode zero seats for $\pi \leq 0.05$.

A rarest-first round-robin quota holds the delivery frontier down to π=0.01 but pays in representativeness. Its $\mathrm{RTC}_{0.1}$ is up to 2.0× worse than ModeRisker's at K=5 (1.55 vs 0.76 at π=0.02, exactly where average-greedy collapses) and uniformly 2.3–4.3× worse at the tight budget K=3 (2.2–3.8× against the label-matched rescue variant). Counting modes without weighing risk either starves the tail or sacrifices representativeness; the two quotas are the mode-aware analogue of the diversity counterpoint below. (iii) *The max-min endpoint.* SATURATE-style bisection (Krause et al., 2008) delivers exactly the containment bound at every π, including sub-β modes that the $\mathrm{CVaR}_{0.1}$ contract does not price, and pays for it in $\mathrm{RTC}_{0.1}$. The two exceptions among the 27 (K, π) cells are ties within 0.2%, one of them at the sub-contract rarity π=0.002. The risk level is a contract dial, and ignoring it costs quality even when it buys delivery. (iv) *How much does greedy leave on the table?* Best-improvement swap local search (multi-start, including from ModeRisker's solution) brackets the discrete optimum. ModeRisker's RTC is within 3.0% of it in every (K, π) cell, strictly better in 8 cells, and indistinguishable at K=3 (within 1.4%, with greedy ahead at six of nine π values). The supergradient linearization is thus more than a fix for the stall: at most a few percent of tail quality separate it from the bracketed optimum.

Delivery is purchasable; representativeness is not. Two rules reach the delivery bound and still lose, for the same reason: both spend seats without regard to where the risk is. The k-DPP's delivery equals pool containment at *every* π, because an isolated rare-branch sample is maximally diverse and is always picked when present. Yet its average scenario quality is 57% worse than ModeRisker's (0.689 vs 0.438), and its RTC is dominated at every β. The naive resulting-risk greedy stalls exactly as Proposition 2 predicts, refining the middle mode indefinitely while near-tied uncovered modes hold the max. On the multimodal stratum its RTC is 1.5× ModeRisker's at every β (1.45 vs 0.98 at β=0.01; 1.05 vs 0.68 at β=0.2). At the average end it falls behind even *random* selection (0.67 vs 0.59 at β=1), a failure it shares only with k-DPP (0.71). Mode-risk selection reaches the containment bound *without* that sacrifice: ModeRisker with rescue inference attains the best RTC at every β *and* near-bound delivery, and it is insensitive to the internal β (RTC varies by $< 1\%$ over $\beta \in [0.05, 0.5]$).

β-matched stratified RTC. On the multimodal stratum ($\pi_{\text{small}} \in \{0.01, 0.02\}$; n=4029 of 8000 instances), ModeRisker improves RTC over average-greedy by 39% at β=0.01 (bootstrap 95% CI [+29%, +48%]), 27% at 0.02 ([+18%, +34%]), 14% at 0.05, 8% at 0.1 and 4% at 0.2, converging to parity for $\beta \geq 0.5$. The dose-response is monotone, every CI up to β=0.2 excludes zero, and the average cost at β=1 is zero (+0.1%, n.s.). The rescue variant matches the curve within one point at every β and requires no mode labels. ModeRisker's tail estimate is also $\approx 8.5\times$ tighter at β=0.01 (bootstrap CI width 0.058 vs 0.498): the improvement comes from removing catastrophic misses, which make the average-greedy tail both bad and noisy. Re-running the full ladder at this scale (n=8000; screen TPR 1.00, FPR 0.00) reproduces the grid ordering under β-matched stratified evaluation; these runs supply the stratum entries of Table 1. The ladder is therefore not an artifact of aggregate (whole-grid) evaluation. On the unimodal stratum all reasonable selectors coincide (random and diversity-only DPP remain visibly worse), and the mode-level machinery reduces exactly to average-greedy. Pooling the strata would dilute the signal; hence stratified reporting.

Mode-inference benchmark. At $\pi = 0.005/0.01/0.02/0.05$ (pool containment 0.91/1.0/1.0/1.0), delivery rates are 0.24/0.57/0.94/1.0 for 5-means, 0.29/0.57/0.94/1.0 for GMM, 0.06/0.36/0.89/1.0 for 8-means (the recovery–delivery decoupling of §3.3) and **0.70/0.99/1.00/1.00** for outlier rescue. Oracle labels reach 0.98 at the hardest setting, so rescue closes most of the inference gap.

Spurious-mode contamination. Hallucinated modes are the standing objection to mode-aware selection, so we inject them deliberately. We replace a fraction $\varepsilon \in \{0, 0.02, 0.05, 0.1, 0.2, 0.4\}$ of each pool with samples from a spurious mid-horizon branch that is absent from the realized distribution (K=5, π=0.02, 100 seeds). Rescue-ModeRisker is fooled at the slot level: 23–29% of selected scenarios lie on the fake branch for $\varepsilon \geq 0.05$ (already 14% at ε=0.02). Yet its true-rare delivery holds at 0.94–0.99 throughout, and its RTC advantage over average-greedy *never reverses*: +0.036/ + 0.008/ + 0.048/ + 0.063/ + 0.058/ + 0.140 across the ε grid. The advantage widens at ε=0.4 because average-greedy's entire budget is pulled toward the contaminated mixture. The vulnerable component is the *screen*, not the selection rule: its fire rate on genuinely multimodal

Table 2: **The holiday depth dose-response across three generator families.** Entries are the per-date client-mean paired advantage $\mathrm{adv}(d) = \overline{\mathrm{bok}}_{\mathrm{avg}} - \overline{\mathrm{bok}}_{\mathrm{rescue}}$ over best-of-K distances (positive favors ModeRisker), on the calibrated half (20 of 40 electricity clients); depths are 2014 values. Screen fire rates are over all 40 clients. "—" marks a quantity not reported for that family.

	Gaussian head trained, 2014	Gaussian head out-of-year, 2013	Chronos-t5-small zero-shot
New Year (depth 0.39)	$+0.090$	$+0.048$	$+0.020$
cross-client 95% CI	$[+0.009, +0.194]$	$[+0.007, +0.097]$	$[-0.015, +0.077]$
Christmas (depth 0.49)	$+0.046$	$+0.060$	$+0.135$
cross-client 95% CI	$[-0.009, +0.110]$	$[+0.008, +0.134]$	$[+0.040, +0.260]$
Deep-holiday mean	$+0.068$	$+0.054$	—
Shallow-holiday cost	-0.028	-0.009	-0.003
date-level 95% CI (7 dates)	$[-0.050, -0.012]$	—	—
$\mathrm{corr}(\mathrm{depth}, \mathrm{adv})$	0.78	0.95	0.87
Miscalibrated-half advantage	-0.49	—	$+0.012$
95% CI	$[-0.76, -0.24]$	—	$[+0.002, +0.028]$
Screen fires, Christmas / other	12.5% / 7.4%	—	47.5% / 24.8%
Gated retention of deep adv.	$+0.017$ of $+0.068$	—	$+0.071$ of $+0.135$

instances degrades with contamination, and the gated variant's delivery drops to 0.83 at $\varepsilon{=}0.4$ while ungated rescue holds 0.98. Spurious modes alone, even at 40% of the pool, cannot produce the real-data reversal of §5(2); that requires distribution-level miscalibration.

5 Real data: where selection pays

We first use DeepAR-style generators (Salinas et al., 2020) on UCI electricity (aggregate and client-level), forecasting 24h from a 168h window with pools of 500. Findings (1)–(3) build on one another; (4)–(5) add two domains that delimit the scope.

(1) Aggregate load is unimodal, and the gate is safe. Aggregate load with a Gaussian AR head yields genuinely unimodal pools (3% screened multimodal; the screen statistic never exceeds 2.1). All selectors tie, and the screen-gated ModeRisker reduces *exactly* to average-greedy, which is the property that makes it safe to deploy. The same holds at client level: on calibrated pools the screen fires on 3 of 1696 instances (0.2%, all normal days), and on every other instance the gated choice is identical to average-greedy's.

(2) A miscalibrated generator defeats any selector. Client-level mixture-density heads (Bishop, 1994) produce 60% screened-multimodal pools but a $1.6\times$ worse pool CRPS than Gaussian heads. On that generator the comparison reverses: rescue-ModeRisker is consistently worse than average-greedy on the holiday stratum of (3) (paired per-date CI $[-0.050, -0.009]$, nominal, under the same $n_{\mathrm{eff}} \approx 4$ caveat as the pilot holiday analysis). The contamination experiment (§4) pins down what breaks. Spurious modes alone do not: even 40% fake pool mass leaves the mode-level advantage positive. The MDN failure is therefore *distribution-level* miscalibration — the whole predictive law is misplaced, as the CRPS gap shows — which mode-aware selection amplifies by committing scenarios to it. Pool calibration is a precondition for mode-level selection; no selection layer repairs a broken generator.

(3) Calendar-blind generators meet real rare events, at scale. We train Gaussian-head generators on load history only (168h window, no calendar features), so national holidays are structurally invisible to the pool. A pilot with 8 hand-picked clients and 4 test-window holidays showed the predicted pattern but was power-limited: $n_{\mathrm{eff}} \approx 4$ dates, an effect concentrated in Christmas, and a date-level CI that crosses zero. We therefore scaled the design along the two axes that carry the power, *dates and clients*, and made every choice outcome-blind. The scaled design has 40 clients (the 8 plus a pre-registered top-up by training-period mean load among clients with $<1\%$ zero hours), a full evaluation year (2014; 9 recorded holiday dates) and one instance per midnight; it reports the per-date client-mean paired advantage $\mathrm{adv}(d)$ of Table 2. Because (2) establishes calibration as a

precondition, we stratify clients by an outcome-independent calibration statistic: a median split on normal-day pool CRPS in per-client standardized units (threshold 0.245; CRPS in the miscalibrated half ranges up to 22.6).

Depth predicts the advantage. Where the precondition holds, both deep holidays show a positive client-mean advantage (mean $+0.068$), and $\mathrm{corr}(\mathrm{depth}, \mathrm{adv}) = 0.78$ across the 9 dates (Table 2). With only two deep dates, uncertainty can be quantified only across the 20 clients, so we report per-date client-level intervals rather than a date-level one. The advantage is spread across clients rather than driven by one: the largest single-client share is 0.45 of the total on both dates, against ≈ 1 in the pilot, and 7–8 of 20 clients are positive per date. Large wins against small losses are the expected signature when only some generators place usable mass near the holiday level. Shallow holidays carry a small but real *cost*: forcing mode-level selection on genuinely unimodal pools buys nothing and costs a little. Where the precondition fails, the pilot's MDN reversal reappears in full.

The gate erases the harms but keeps little of the gain. The gated selector's advantage over average-greedy is -0.0005 on shallow dates and -0.001 on miscalibrated clients, so both harms are erased. However, it retains only $+0.017$ of the $+0.068$ deep advantage, because calendar-blind pools *look unimodal on holiday mornings*: a pool-shape screen cannot fire when the mode is missing from the pool itself. Capturing the rest requires the ungated selector, at the shallow-date cost, or a calendar-aware generator: mode suppression in the generator is upstream of selection (§6).

Two replications: another year, another generator family. An out-of-year replication (train through 2012, evaluate on 2013; 9 further holiday dates; the *same* absolute calibration threshold) reproduces the pattern on calibrated clients. Across both years all four deep holiday dates are positive ($+0.046$ to $+0.090$), and three of the four cross-client CIs exclude zero (2014 Christmas is marginal at $[-0.009, +0.110]$). The one-year training window breaks a few clients' generators outright, with normalized CRPS up to 10^{12}; the calibration screen removes them by construction. The remaining above-threshold clients in 2013 are only mildly worse-calibrated and show an attenuated, not reversed, effect, which suggests that the precondition in (2) acts as a dose, not a cliff. A third generator family, the zero-shot foundation forecaster Chronos-t5-small (Ansari et al., 2024) (500 sampled paths per instance, calendar-blind like the trained heads), replicates the design end to end and relaxes both of its weak points. The deep-holiday advantage survives the generator swap: Christmas, at $+0.135$, is the largest single-date advantage in the study. The shallow cost shrinks to -0.003, so mode-splitting on well-calibrated unimodal pools is nearly free. *Both* calibration halves stay positive, because zero-shot pools are uniformly calibrated: normal-day CRPS is 0.06–0.55 across all 40 clients, whereas the trained heads reach 22.6. This is the dose reading of (2) seen from the other side. Most important for deployment, the foundation pool *contains* the holiday mode well enough for the screen to see it. The screen fires on 47.5% of Christmas instances against a non-holiday rate of 24.8% (trained heads: 12.5% against 7.4%), and the gated selector retains $+0.071$ of the $+0.135$. These rates are over all 40 clients of the scaled design; the 0.2% quoted in (1) comes from the eight-client pilot, a smaller and differently chosen client set. The gate's blind spot is therefore a property of the generator's support, not of the gate; the M5 closure in (5) is the same statement taken to zero containment. The diversity-only DPP baseline repeats its synthetic signature: it delivers low-load scenarios indiscriminately (91% on *all* days in the pilot) and pays for it in coverage every day.

(4) A second domain delineates the method's scope. Traffic occupancy (PEMS, Lai et al., 2018; 20 sensors chosen outcome-blind by training-split rush-hour dispersion; the congested stratum is each sensor's top realized quintile of rush-hour means) reverses every electricity precondition. Pools are genuinely multimodal, with the screen firing on 18–22% of instances, and *both* generators are calibrated (pool CRPS 0.35 for the Gaussian head and 0.29 for the MDN, which is the better generator here). Yet mode-level selection finds almost nothing to fix. Rescue doubles the delivery of high-congestion scenarios (0.61–0.67 vs 0.33–0.44), but congested-stratum coverage is a tie (0.725 vs 0.719). The only realized gain is at the extreme instance tail of the better generator ($\mathrm{RTC}_{0.01}$ of 1.54 vs 1.64, -5.6%), and the mean pays for it ($+8\%$ at β=1). The gate is nearly free on the better-calibrated MDN generator (within 0.4% of average-greedy at every $\beta \leq 0.5$) but costs 1.5–3.7% on the Gaussian one. With no rare mode to find, every one of the screen's firings is a false positive; this is the price of a gate in a domain that needs none. The contrast with holidays states the scope of this paper. Rush-hour congestion is not a rare mode but the *high-mass regime* of a continuous severity

Table 3: **The three-domain scope taxonomy.** Each real domain occupies a different cell of the (rarity, containment) plane, and the modality gate is required to behave differently in each. Numbers are those reported in §5(3)–(5).

Domain	Rare mode?	In pool support?	Generator calibrated?	Does selection pay?	What the gate does
Electricity (deep holidays)	yes	yes	yes on half the clients	yes: deep-holiday advantage $+0.068$	fires too rarely; keeps only $+0.017$ of $+0.068$
Traffic (rush-hour congestion)	no: high-mass regime	yes	yes, both heads (CRPS 0.35 / 0.29)	no: congested coverage ties, 0.725 vs 0.719	fires on 18–22%, all false positive; costs $\leq 0.4\%$ (MDN), 1.5–3.7% (Gaussian)
M5 retail (Christmas closure)	yes	no: containment ≈ 0	no on the event: CRPS 1.15 vs 0.57	no: all selectors blind, best-of-K 1.68–1.74	fires on 0.06% of instances; stands down exactly (adv $\equiv 0$)

spectrum that history-trained generators sample daily, so average-optimal selection already covers it, and a well-behaved gate should keep the machinery inert.

(5) A third domain closes the scope taxonomy: the out-of-support mode. M5 retail demand (Makridakis et al., 2022) instantiates the failure that, by the containment bound, no selector can address. We use 70 store×department daily series aggregated outcome-blind from the item level, per-series calendar-blind Gaussian generators trained on sales through 2014, and the 30 recorded calendar events of 2015 for evaluation. The deepest event is Christmas, a store closure with realized sales at or near zero on the day itself: 87% of the 70 series realize exactly 0 units (median 0, maximum 4). The best-of-K and CRPS figures below cover the full 7-day horizon that starts on that day. Pool containment is ≈ 0, so every selector is equally blind: Christmas best-of-K is 1.68–1.74 for random, DPP, average-greedy and rescue alike, against 0.81 for average-greedy on normal days, and pool CRPS doubles (1.15 vs 0.57). Rescue finds no low path because none exists. At this aggregation level the pools are unimodal *everywhere*: the screen fires on 15 of 25 130 instances (0.06%) and on none of the 2100 event instances. The ungated selector therefore pays the familiar unimodal spread cost on all dates (-0.012 over all 70 series, CI $[-0.018, -0.006]$; $\mathrm{corr}(|\mathrm{depth}|, \mathrm{adv}) = -0.49$), while the gated selector coincides with average-greedy *identically* (adv $\equiv 0$ on all 30 event dates). This is the one regime the gate handles perfectly. The three real domains now tile the (rarity, containment) plane that parameterizes the theory (Table 3). Selection pays where a future of probability π is rare *and* present in a calibrated pool. Everywhere else, including M5, where containment is zero and no selection rule can help, the correct behavior is to stand down, which the gate does exactly. The fix for M5 is generator-side (calendar covariates, or an output law that can place mass at closure), upstream of any selection layer.

6 Discussion and limitations

Mode granularity. Mode-level CVaR requires a choice of mode granularity. The rescue layer reduces the sensitivity to this choice but is not parameter-free (Appendix B): its outlier threshold has little effect, but it needs a coarse count of the main regimes, and the budget must exceed the inferred mode count. At $K{=}3$ with ≈ 5 inferred modes, rescue delivery collapses to 0.07 at $\pi{=}0.005$. **The screen is a gate, not a guarantee.** Unimodal-looking pools can hide modes that the generator itself failed to sample, as the M5 Christmas closure shows, and such generator-side mode suppression is upstream of any selection layer. **Estimation cost.** RTC_β at small β is expensive to estimate. For average-greedy at $\beta{=}0.01$, the bootstrap (Efron, 1979) 95% CI width is 0.50 (31% of the point estimate) even with $n{\approx}4000$ multimodal-stratum instances. The width grows by the usual $\approx 2\times$ per $4\times$ reduction in n wherever the tail stays populated (1.8–2.1× per step for $\beta \geq 0.05$). At smaller β the width stops growing only because the estimate degenerates: at $\beta{=}0.01$ it widens by 1.7× and then 1.2×, by which point the tail holds three instances. In practice, $\beta n \gtrsim 40$ tail instances is the floor for a stable comparison at a single β, which is why we report the monotone dose-response over β rather than a single value.

REPRODUCIBILITY STATEMENT

§3.4 states the RTC_{β} evaluation protocol with its two stratifications, by oracle modality and by the modality screen; §5(3) states the holiday protocol's windows, outcome-blind client selection and calibration split. Proposition assumptions and proofs are in Appendix A. The supplement at `https://anonymous.4open.science/r/moderisker-CD82` ships per-instance rows rather than aggregates alone, so every bootstrap interval we report can be recomputed on a CPU without retraining a generator; `REPRODUCTION.md` pairs each quantity with its script and run directory. Data preparation is scripted; the raw benchmark datasets are not redistributed.

AI USE STATEMENT

We have not used generative AI tools for any task with required disclosure: they generated no data, formulated or proved no claim, designed no method or experiment, implemented no component, and neither analyzed nor interpreted any result; translation is not applicable to this work. We used generative AI tools for two tasks with recommended disclosure: polishing the wording of text drafted by the authors, and reviewing our experiment code for bugs. We have reviewed all AI-assisted work. Each suggested rewording was checked by an author against the intended technical meaning before it was adopted; each issue raised in code review was confirmed by an author, who wrote and tested the fix, and every number reported here comes from the released code, run by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social LSTM: Human trajectory prediction in crowded spaces. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 961–971, 2016.

Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Yuyang Wang. Chronos: Learning the language of time series. *Transactions on Machine Learning Research*, 2024.

David Arthur and Sergei Vassilvitskii. k-means++: The advantages of careful seeding. In *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pp. 1027–1035, 2007.

Philippe Artzner, Freddy Delbaen, Jean-Marc Eber, and David Heath. Coherent measures of risk. *Mathematical Finance*, 9(3):203–228, 1999.

Christopher M. Bishop. Mixture density networks. Technical Report NCRG/94/004, Aston University, 1994.

Ming-Fang Chang, John Lambert, Patsorn Sangkloy, Jagjeet Singh, Slawomir Bak, Andrew Hartnett, De Wang, Peter Carr, Simon Lucey, Deva Ramanan, and James Hays. Argoverse: 3D tracking and forecasting with rich maps. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Yutian Chen, Max Welling, and Alex Smola. Super-samples from kernel herding. In *Conference on Uncertainty in Artificial Intelligence*, 2010.

Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B*, 39(1):1–38, 1977.

Jitka Dupačová, Nicole Gröwe-Kuska, and Werner Römisch. Scenario reduction in stochastic programming: An approach using probability metrics. *Mathematical Programming*, 95(3):493–511, 2003.

Bradley Efron. Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, 7(1): 1–26, 1979.

Tilmann Gneiting and Adrian E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.

Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social GAN: Socially acceptable trajectories with generative adversarial networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2255–2264, 2018.

Abner Guzmán-Rivera, Dhruv Batra, and Pushmeet Kohli. Multiple choice learning: Learning to produce multiple structured outputs. In *Advances in Neural Information Processing Systems*, 2012.

Holger Heitsch and Werner Römisch. Scenario reduction algorithms in stochastic programming. *Computational Optimization and Applications*, 24(2-3):187–206, 2003.

Kjetil Høyland and Stein W. Wallace. Generating scenario trees for multistage decision problems. *Management Science*, 47(2):295–307, 2001.

Roger Koenker and Gilbert Bassett. Regression quantiles. *Econometrica*, 46(1):33–50, 1978.

Andreas Krause and Daniel Golovin. Submodular function maximization. In *Tractability: Practical Approaches to Hard Problems*. Cambridge University Press, 2014.

Andreas Krause, H. Brendan McMahan, Carlos Guestrin, and Anupam Gupta. Robust submodular observation selection. *Journal of Machine Learning Research*, 9:2761–2801, 2008.

Alex Kulesza and Ben Taskar. Determinantal point processes for machine learning. *Foundations and Trends in Machine Learning*, 5(2-3):123–286, 2012.

Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. Modeling long- and short-term temporal patterns with deep neural networks. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 95–104, 2018.

Daniel Levy, Yair Carmon, John C. Duchi, and Aaron Sidford. Large-scale methods for distributionally robust optimization. In *Advances in Neural Information Processing Systems*, 2020.

Stuart P. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28 (2):129–137, 1982.

Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. M5 accuracy competition: Results, findings, and conclusions. *International Journal of Forecasting*, 38(4):1346–1364, 2022.

James E. Matheson and Robert L. Winkler. Scoring rules for continuous probability distributions. *Management Science*, 22(10):1087–1096, 1976.

George L. Nemhauser, Laurence A. Wolsey, and Marshall L. Fisher. An analysis of approximations for maximizing submodular set functions—I. *Mathematical Programming*, 14(1):265–294, 1978.

Pierre Pinson, Henrik Madsen, Henrik Aa. Nielsen, George Papaefthymiou, and Bernd Klöckl. From probabilistic forecasts to statistical scenarios of short-term wind power production. *Wind Energy*, 12(1):51–62, 2009.

Kashif Rasul, Abdul-Saboor Sheikh, Ingmar Schuster, Urs Bergmann, and Roland Vollgraf. Multivariate probabilistic time series forecasting via conditioned normalizing flows. In *International Conference on Learning Representations*, 2021.

R. Tyrrell Rockafellar and Stanislav Uryasev. Optimization of conditional value-at-risk. *Journal of Risk*, 2(3):21–41, 2000.

Peter J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20(1):53–65, 1987.

Christian Rupprecht, Iro Laina, Robert DiPietro, Maximilian Baust, Federico Tombari, Nassir Navab, and Gregory D. Hager. Learning in an uncertain world: Representing ambiguity through multiple hypotheses. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, 36(3): 1181–1191, 2020.

Tim Salzmann, Boris Ivanovic, Punarjay Chakravarty, and Marco Pavone. Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data. In *European Conference on Computer Vision (ECCV)*, 2020.

Michael Scheuerer and Thomas M. Hamill. Variogram-based proper scoring rules for probabilistic forecasts of multivariate quantities. *Monthly Weather Review*, 143(4):1321–1334, 2015.

Alexander Shapiro, Darinka Dentcheva, and Andrzej Ruszczyński. *Lectures on Stochastic Programming: Modeling and Theory*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 3rd edition, 2021.

Ruofeng Wen, Kari Torkkola, Balakrishnan Narayanaswamy, and Dhruv Madeka. A multi-horizon quantile recurrent forecaster. *arXiv preprint arXiv:1711.11053*, 2017.

A PROOFS

Proposition 1. The problem $\max_{q\in\Delta_\beta} q^\top \ell$ is a linear program over a box intersected with the simplex constraint $\sum_c q_c = 1$; equivalently, distribute a budget of β probability mass across modes, mode c holding at most π_c, to maximize $\sum_c (\text{mass}_c/\beta)\,\ell_c$. A standard exchange argument shows greedy waterfilling by decreasing ℓ is optimal: if an optimal solution put mass on mode j while mode i with $\ell_i > \ell_j$ had slack, moving ε of mass from j to i increases the objective by $\varepsilon(\ell_i - \ell_j)/\beta > 0$. The dual (Rockafellar–Uryasev) representation $\mathrm{CVaR}_\beta(\ell;\pi) = \max_{\Delta_\beta} q^\top \ell$ makes CVaR a pointwise maximum of linear functions of ℓ; any maximizing q is therefore a supergradient at ℓ. □

Proposition 2. With $\beta \le \min_c \pi_c$ the box constraint allows q concentrated on a single mode, so $\mathrm{CVaR}_\beta(\ell) = \max_c \ell_c$. By assumption every candidate leaves at least one of ℓ_a, ℓ_b at the shared maximum value, hence the resulting CVaR equals that value for every candidate: the naive score is constant, and any tie-break rule (e.g. first index) is decisive. If the tie-break systematically selects candidates refining covered modes (as observed with index-ordered pools), modes a, b are never covered. For the linearized rule, the supergradient at the current losses puts $q^\star_a, q^\star_b > 0$ (both are in the top-β mass), so any candidate reducing ℓ_a or ℓ_b has strictly positive score while refinements of covered modes score 0. □

Proposition 3. Let $q^\star$ attain $\mathrm{CVaR}_\beta(\ell;\hat\pi)$. Define $\tilde q_c = \min(q^\star_c, \pi_c/\beta)$; the total deficit is $D = \sum_c \big(q^\star_c - \tilde q_c\big) \le \sum_c (\hat\pi_c - \pi_c)_+/\beta \le \|\pi - \hat\pi\|_1/\beta$. Redistribute D onto coordinates with slack under π/β (possible since both boxes contain total mass ≥ 1); each redistributed unit changes the objective by at least $-(\max_c \ell_c - \min_c \ell_c)$. The resulting q is feasible for π, so $\mathrm{CVaR}_\beta(\ell;\pi) \ge \mathrm{CVaR}_\beta(\ell;\hat\pi) - \|\pi - \hat\pi\|_1(\max \ell - \min \ell)/\beta$; exchanging roles of $\pi, \hat\pi$ gives the other direction. □

B ADDITIONAL EXPERIMENTS

Sensitivity to the number of inferred modes C. With k-means inference on the stratified benchmark (multimodal stratum), delivery *decreases* in C: 0.72 at C=5, 0.63 at C=8 and 0.59 at C=12, against 0.98 with oracle modes. Overclustering is not a safe direction: each extra fragment carries a $\hat\pi$ comparable to β and crowds the CVaR budget (the mechanism of §3.3 and Proposition 3). This motivates outlier rescue, which adds candidate rare modes only where residual structure demands them.

Rescue hyperparameters: what binds and what does not. We run a full grid over the rescue layer's three hyperparameters: `min_size`$\in \{1,2,3,4\}$ × `flag_mult`$\in \{2.0, 2.5, 3.0\}$ × $n_{\text{main}} \in \{3,4,5,6\}$ (48 combinations, 100 seeds each, K=5, $\pi \in \{0.002, 0.005, 0.01, 0.02\}$). We rank the three by how much delivery moves across each one's range with the other two held fixed. n_{main} dominates (mean spread 0.31, up to 0.64), `min_size` is second (0.12), and the outlier threshold

flag_mult is by far the weakest (mean 0.07, median 0.045). flag_mult is not strictly free: its spread exceeds 0.06 in 10 of the 16 (min_size, n_{main}) cells and reaches 0.34. But this happens almost only at settings where the other two have already destroyed the low-π regime; it never decides *whether* the rare mode is delivered. The two parameters that encode a *prior on mode structure* do decide it, each through a clear mechanism. (i) min_size: a rescue cluster must contain min_size samples, and the expected rare count is $N\pi$ (1–2.5 at $\pi \leq 0.005$, N=500). Raising it therefore bites exactly where the rare mode is thinnest, and only there. Pooled over the other two parameters, delivery falls from 0.215 to 0.089 at π=0.002 and from 0.462 to 0.305 at π=0.005 as min_size goes from 1 to 4, while π=0.02 is unaffected ($0.962 \rightarrow 0.967$). (ii) n_{main} binds hardest: raising it from 3 to 6 overclusters the three true main branches, and the fragments reproduce in miniature the CVaR budget-crowding of Proposition 3. Pooled delivery collapses from 0.59 to 0.13 at π=0.005 and from 0.91 to 0.50 at π=0.01, again largely sparing π=0.02 ($1.00 \rightarrow 0.91$). The defaults (min_size=2, any flag_mult, n_{main}=3–4) sit on a wide plateau: the layer needs a coarse count of the main regimes, not tuning.

The budget must exceed the inferred mode count. With rescue inference the pipeline infers 4.3–5.1 modes on average. At K=3 delivery collapses ($0.00/0.07/0.18/0.67$ at π=$0.002/0.005/0.01/0.02$), while oracle labels still deliver 0.90 at π=0.005. With fewer slots than inferred modes, fragments and rare candidates compete for a budget that cannot cover them all, and the true rare mode loses to whichever inferred mode holds more estimated mass. K=10 recovers ($0.38/0.78/0.98/1.00$ vs oracle $0.68/0.91/1.00/1.00$). This is a deployment constraint, not a tuning failure: a mode-level risk contract requires $K \gtrsim$ the number of modes the inference layer reports. Since the layer reports that number, the condition can be checked at selection time.

Screen-threshold sensitivity. The holiday runs record every instance's screen statistic together with both branch outcomes, so the gated selector can be reconstructed at any threshold τ without re-running selection. On the calibrated Gaussian generator the statistic is tightly concentrated (median 1.23, 90th percentile 1.35, maximum 7.7). Any $\tau \in [1.5, 5]$ fires on at most 0.4% of instances, and the gated selector coincides with average-greedy to three decimals. The default τ=2 thus sits on a wide plateau; only pushing τ below the bulk of the distribution (τ=1, firing on 83%) switches the system to rescue behavior. On the miscalibrated MDN generator the screened fraction declines smoothly with τ ($72\% \rightarrow 7\%$ over $\tau \in [1, 5]$), and holiday coverage improves *monotonically* toward the average-greedy endpoint ($0.453 \rightarrow 0.418$ vs 0.417). When modes are hallucinated, every marginal firing hurts, so raising τ always fails safe. The asymmetry is itself diagnostic: a screen statistic whose upper tail is heavy relative to a calibrated reference flags generator miscalibration before any selection is run.