Bandits via Additive Quantized Representations

Ami Tavory[1] **Noam Touitou**[1] **Tal Sarig**[1] **Frank Cheng**[1] **Ido Guy**[2]
[1]Meta Platforms
[2]Technion - Israel Institute of Technology

Abstract

Contextual bandits require balancing nonlinear reward modeling with online efficiency. Tree ensembles and neural methods capture nonlinearities but require periodic retraining and large replay buffers. Linear models update efficiently per observation with $O(1)$ memory, but are fundamentally restricted to linear reward structures. We propose Residual Quantization (RQ) as a representation layer to bridge this gap. An offline-trained RQ codebook maps continuous contexts into discrete centroid assignments across ℓ levels, set dynamically through a shadow mechanism. This enables a spectrum of additive bandit algorithms that achieve nonlinear expressivity with strictly bounded memory. Across 13 datasets, RQ variants beat their non-RQ counterparts on 11 of 13 datasets, often by wide margins, while matching doubling-retrain XGBoost and neural baselines at up to $1000\times$ less memory.

1 Introduction

In contextual bandits, an agent repeatedly observes a context, selects an action, and receives a reward, learning to improve its choices from each interaction. Every decision is made once and in real time, so the algorithm should ideally update from each observation as it arrives.

Tree ensembles (Nilsson et al., 2024) and neural methods (Zhou et al., 2020; Zhang et al., 2021) capture nonlinear reward structure, but they periodically retrain on a replay buffer that can grow linearly with the horizon. At five million rounds on a 28-feature dataset, a doubling-retrain bandit (retraining at exponentially spaced intervals on all past data) consumes over 1 GB of replay memory (Figure 3). Linear methods (LinTS (Agrawal and Goyal, 2013), LinUCB (Abbasi-Yadkori et al., 2011)) avoid memory and spiky retraining issues, but are structurally restricted to linear rewards.

This is not a smooth tradeoff. Practitioners face a binary choice: expressive models with unbounded memory, or bounded memory with linear rewards. The gap between the two regimes, nonlinear expressiveness at fixed memory cost, is the problem we address.

Our approach. We use Residual Quantization (RQ) (Chen et al., 2010) as a bandit representation layer to enable gradient-boosted reward modeling with fixed $O(1)$ memory. We train an RQ codebook of ℓ levels with b centroids each via unsupervised k-means on unlabeled contexts (assumed available and cheap) and freeze it before the bandit starts. RQ codebooks were originally developed for efficient retrieval; we repurpose them for gradient-boosted bandits, where we fit a bandit model at each level to predict the *pseudo-target*, the gap between the observed reward and the cumulative prediction of shallower levels. With ℓ levels and b centroids, this yields b^ℓ effective cells using only ℓb parameters, a potentially exponential compression. The online phase requires no replay buffer, no retraining, and fixed $O(1)$ memory at every horizon.

The algorithm adapts depth online via a *shadow* mechanism: a shadow model trains at the next depth but does not influence arm selection, so the active model's performance is unaffected during evaluation. A statistical test at pre-committed check times promotes the shadow if it improves prediction; otherwise the method defaults to its initial depth (Section 2).

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

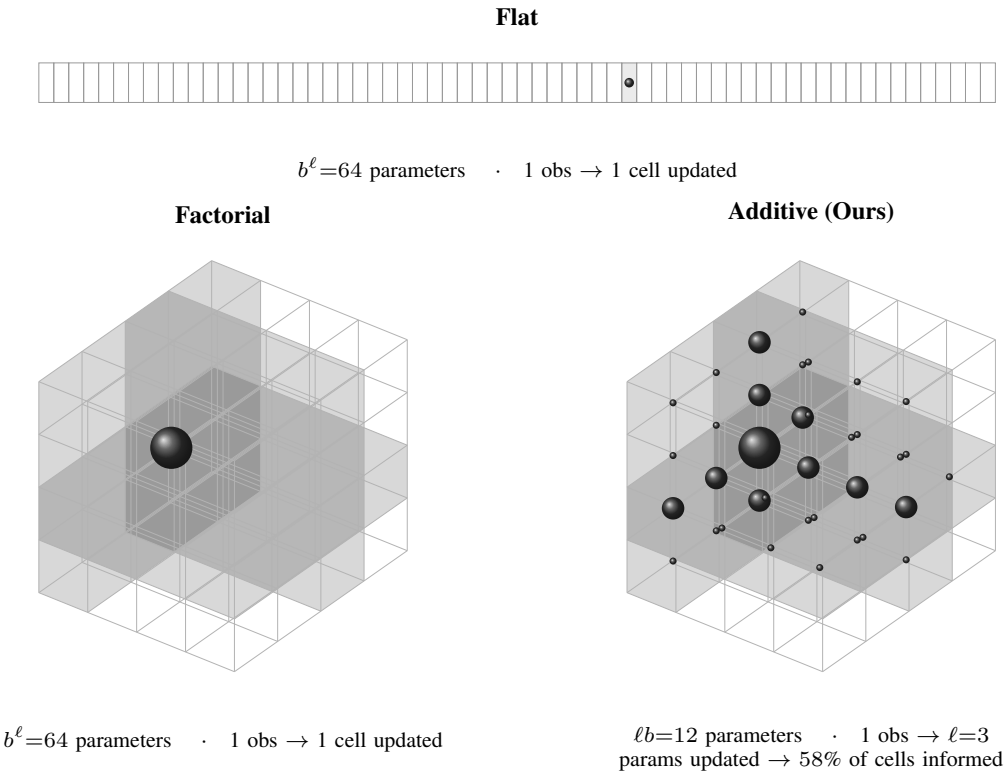


Figure 1: Partitioning the context space creates nonlinear reward models from linear per-cell learners, but finer partitions fragment the data: each cell sees fewer observations. All three schemes realize b^ℓ=64 cells (b=4, ℓ=3). **Top/Bottom left:** Flat and factorial parameterizations learn each cell from its own data alone. **Bottom right:** Additive RQ achieves the same 64-cell granularity with ℓb=12 shared parameters—each observation updates ℓ=3 centroids, pooling information across cells. Fine partition without data starvation.

We evaluate the architecture across three variants that span the efficiency–robustness spectrum: *TS-RQ* ($O(\ell K)$/round) uses per-centroid scalar counters with no features, robust and simple. *SGD-LinTS-RQ* ($O(\ell dK)$) fits diagonal linear models on the context residuals at each level. *LinTS-RQ* ($O(\ell d^2 K)$) fits full-covariance LinTS, the quality ceiling, competitive with XGBoost and NeuralCB at $1000\times$ less memory when data per parameter suffices (Section 4.2). All three use shadow promotion and maintain fixed memory.

Contributions.

1. *RQ as a bandit model*: we repurpose residual quantization from retrieval to bandits, using the additive codebook as a nonlinear reward model that maintains fixed $O(1)$ memory and requires no replay buffer.
2. *Analysis*: under three base-learner convergence properties (verified for all variants), the asymptotic regret rate is bounded by the additive misspecification Δ_ℓ (Theorem 3.1). We decompose depth-dependent approximation error exactly into saturated quantization and additive-interaction terms, and prove geometric convergence to the irreducible floor when reconstruction and additive-tail errors contract geometrically (Theorem 3.2 and Corollary 3.1). A shadow mechanism adapts depth online with false-promotion probability $\leq \alpha$ (Theorem 3.3).
3. *Scale experiments*: we evaluate on 13 datasets with up to 11M rounds and 30 seeds each, showing that RQ methods improve over their within-class baselines on 11 of 13 datasets and that LinTS-RQ is competitive with XGBoost (7W/4L/2T) at $1000\times$ less memory.

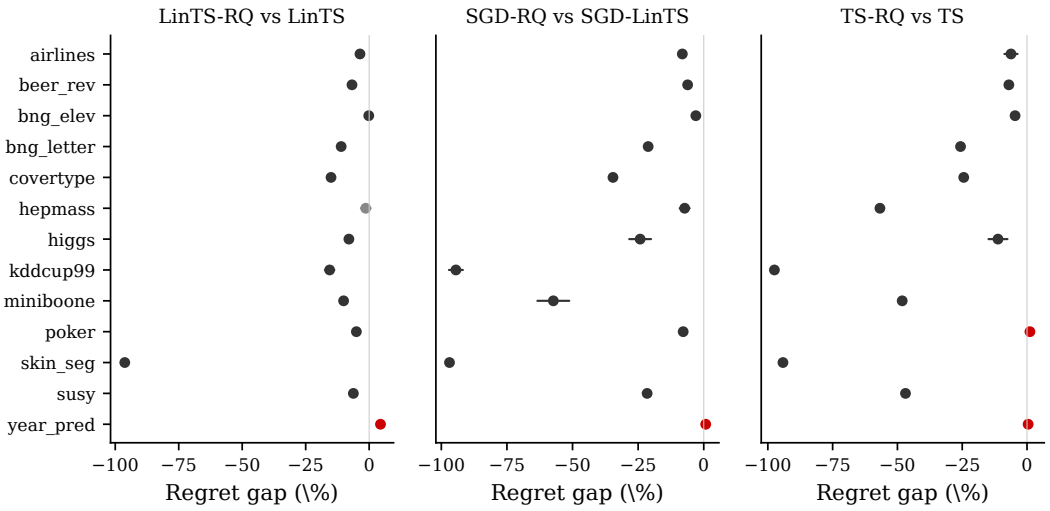


Figure 2: Regret gap (%) of each RQ method relative to its within-class baseline on 13 datasets (30 seeds, 95% CIs). Negative = RQ method has lower regret. Black: significant win (CI excludes zero). Gray: not significant. Red: significant loss. **Left:** LinTS-RQ vs LinTS wins on 11/13 datasets; the sole loss (year_prediction, d=90, K=84) is explained by the data sufficiency condition $N/(bKd^2) \ll 1$ (Section 4.2). **Center:** SGD-LinTS-RQ vs SGD-LinTS wins on 12/13. **Right:** TS-RQ vs TS wins on 11/13.

2 Method

Setting. We consider the standard stochastic contextual bandit setting (Lattimore and Szepesvári, 2020): at each round t, the agent observes a context $x_t \in \mathbb{R}^d$, selects an arm $a_t \in [K]$, and receives a reward $r_t \in \{0, 1\}$. The goal is to minimize cumulative regret $R(T) = \sum_{t=1}^{T}[r^*(x_t) - r_t]$, where r^* is the oracle reward.

We assume access to a pool of n_{rq} *unlabeled* contexts—separate from the bandit evaluation pool—on which we train the RQ codebook via sequential k-means. We train the codebook once and freeze it before the bandit starts. This is a standard assumption in settings where unlabeled contexts are abundant (user profiles, item catalogs, sensor readings) but rewards require costly sequential interaction. The codebook encodes each context via greedy k-means assignment:

$$k_i = \arg\min_{k \in [b]} \|x^{(i-1)} - c_k^{(i)}\|^2, \qquad x^{(i)} = x^{(i-1)} - c_{k_i}^{(i)}, \qquad x^{(0)} = x, \tag{1}$$

producing a code sequence $z = [k_1, \ldots, k_\ell]$ that partitions the context space, and context residuals $x^{(0)}, x^{(1)}, \ldots, x^{(\ell-1)}$ that serve as per-level feature inputs to the bandit model below.

Gradient-boosted bandit over RQ levels. All methods share the architecture in Algorithm 1. Each RQ level operates on the context residual left by shallower levels, and boosts (Friedman, 2001) the reward prediction: level 0 predicts the raw reward; level $i > 0$ predicts the *pseudo-target*—the gap between the observed reward and the cumulative prediction of levels $0, \ldots, i-1$. A damping factor $\eta \in (0, 1]$ controls each level's contribution.

The three instantiations differ only in the per-level model class $\mathcal{M}$ (Table 1).

For LinTS-RQ, pseudo-targets at deeper levels have smaller magnitude and fewer effective samples; we describe the per-level regularization and numerical scaling in Section 4.1.

Shadow promotion. Algorithm 1 adapts depth ℓ online via a shadow level that trains alongside the active model but does not influence arm selection (line 13). At pre-committed check times (line 16), a statistical test compares the shadow's prediction accuracy against the active model over a trailing window. If the shadow wins, it is promoted and a new shadow spawned; otherwise the active depth is unchanged. We analyze the false-promotion guarantee in Section 3.3.

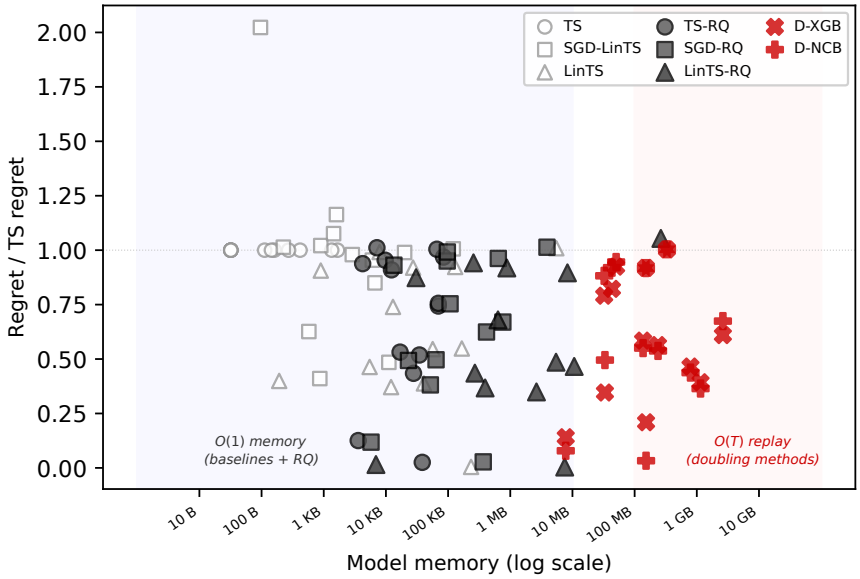


Figure 3: Memory vs. quality across 13 datasets. Each point is one (method, dataset) pair. Hollow gray: linear baselines ($O(1)$ memory, high regret). Solid gray: RQ methods ($O(1)$ memory, low regret). Red: doubling methods ($O(T)$ replay buffers, low regret). RQ methods match tree/neural regret at $1000\times$ less memory. The vertical drop from hollow to solid at the same x-position shows the regret gain from the RQ codebook.

Table 1: Model variants sharing Algorithm 1, differing only in per-cell learner $\mathcal{M}$ (full pseudocode in Appendix A).

Method	$\mathcal{M}$	**Features used**	**Cost/round**
TS-RQ	Count/sum	Index k_i only	$O(\ell K)$
SGD-LinTS-RQ	Diagonal linear	Index k_i + $[x^{(i-1)}; 1] \in \mathbb{R}^{d+1}$	$O(\ell d K)$
LinTS-RQ	Full-covariance LinTS	Index k_i + $[x^{(i-1)}; 1] \in \mathbb{R}^{d+1}$	$O(\ell d^2 K)$

3 Analysis

We analyze Algorithm 1 along three axes: asymptotic regret under base-learner convergence properties, the effect of depth on misspecification, and false-promotion control for the shadow mechanism.

3.1 Asymptotic regret

Definition 3.1 (Additive misspecification)**.** *The* additive misspecification *at depth ℓ is*

$$\Delta_\ell(\mathcal{M}) = \inf_{f_1,\ldots,f_\ell \in \mathcal{M}} \mathbb{E}_x \left[\max_a \left| r^*(x,a) - \sigma\left(\sum_{i=1}^{\ell} f_i(k_i(x), a) \right) \right| \right]. \tag{2}$$

We require three properties of the per-cell base learner $\mathcal{M}$, stated informally here (formal definitions and verification for all three variants in Appendix B):

1. *Eventual boundedness*: score means do not diverge.
2. *Exploration*: if an arm rarely visits a centroid, its score variance remains positive, ensuring continued exploration.
3. *Consistency*: when shallower levels stabilize, each level's predictions converge to the best predictor in its function class.

Algorithm 1 RQ Bandit with Shadow Promotion

Require: Codebook $\mathcal{C}^{(1)}, \ldots, \mathcal{C}^{(\ell_{\max})}$; model array $\mathcal{M}[i, c, a]$ for each level $i \in [\ell_{\max}]$, centroid $c \in [b]$, arm $a \in [K]$; clipping function $\sigma(z) = \mathrm{clip}(z + 0.5, 0, 1)$; damping η; initial depth ℓ_0; significance level α

$\ell_{\mathrm{act}} \leftarrow \ell_0$; initialize shadow at level $\ell_0{+}1$
for $t = 1, \ldots, T$ **do**
  Encode: observe x_t; compute codes $(k_1, \ldots, k_{\ell_{\max}})$ and context residuals $(x_t^{(0)}, x_t^{(1)}, \ldots)$ via (1)
  Score: $s_a \leftarrow \sum_{i=1}^{\ell_{\mathrm{act}}} \eta^{i-1} \cdot \mathcal{M}[i,\, k_i,\, a].\mathrm{sample}(x_t^{(i-1)})$ for each arm a
  Select: $a_t \leftarrow \arg\max_a s_a$; observe reward $r_t \in \{0, 1\}$
  Targets (residual cascade): $\hat{y}_0 \leftarrow 0$
  for $i = 1, \ldots, \ell_{\mathrm{act}}$ **do**
    $\hat{\mu}_i \leftarrow \mathcal{M}[i,\, k_i,\, a_t].\mathrm{mean}(x_t^{(i-1)})$ ▷ level-i prediction
    $\tilde{r}_i \leftarrow r_t - \sigma(\hat{y}_{i-1})$ ▷ residual target
    $\hat{y}_i \leftarrow \hat{y}_{i-1} + \eta^{i-1}\hat{\mu}_i$ ▷ accumulate
  end for
  Update: $\mathcal{M}[i,\, k_i,\, a_t].\mathrm{update}(x_t^{(i-1)}, \tilde{r}_i)$ for $i = 1, \ldots, \ell_{\mathrm{act}}$
  Shadow: $\mathcal{M}[\ell_{\mathrm{act}}{+}1,\, k_{\ell_{\mathrm{act}}+1},\, a_t].\mathrm{update}(x_t^{(\ell_{\mathrm{act}})}, \tilde{r}_{\ell_{\mathrm{act}}+1})$
  if $t \in \{T_0, 2T_0, 4T_0, \ldots\}$ **and** $\ell_{\mathrm{act}} < \ell_{\max}$ **then**
    if Freedman test (Theorem 3.3, Appendix D): shadow significantly better at level $\alpha/(n_c \cdot (\ell_{\max}{-}1))$, where n_c is the number of checks **then**
      ℓ_{act} += 1; spawn new shadow
    end if
  end if
end for

Theorem 3.1 (Asymptotic regret)**.** *For any base learner $\mathcal{M}$ satisfying Properties 1–3, let $\hat{r}_\infty$ denote the converged prediction of the sequential residual-fitting procedure (Appendix C). Then*

$$\limsup_{T\to\infty} \frac{R(T,\ell)}{T} \;\le\; 2\,\mathbb{E}_x\Big[\max_a |r^*(x,a) - \hat{r}_\infty(x,a)|\Big]. \tag{3}$$

Since $\hat{r}_\infty$ is one feasible additive predictor, the right-hand side equals $2\,\Delta_\ell(\mathcal{M})$ when the stagewise and joint optima coincide; in general it is controlled by the stagewise approximation error.

The key technical step is an inductive convergence proof across boosting levels: each level's targets depend on all shallower levels' estimates, creating a coupled system not covered by single-model analysis. We establish Properties 1–3 for TS-RQ, SGD-LinTS-RQ, and LinTS-RQ in Appendix B; the full proof appears in Appendix C. The verification follows the standard approach in the TS literature (Agrawal and Goyal, 2013); a fully rigorous treatment of the strong law under adaptive Thompson Sampling with misspecification remains open in general, as noted by Agrawal and Goyal (2012). The bound says that average regret per round depends on the stagewise approximation error, not directly on the horizon T. When that error is zero, the method achieves sublinear regret; under misspecification, regret may grow linearly at the residual error rate. This is not the optimal $O(\sqrt{T})$ rate that LinTS achieves under correct specification; the tradeoff is a richer non-linear reward model at fixed memory.

3.2 Misspecification and depth

The additive parameterization resolves b^ℓ effective cells with only ℓb parameters per arm, and each observation updates ℓ centroid models—so exploration cost grows linearly in ℓ while resolution grows exponentially. Since deeper models contain shallower ones ($f_{\ell+1} \equiv 0$), $\Delta_{\ell+1} \le \Delta_\ell$ trivially; the question is the *rate* of decay.

We can now separate the two mechanisms that determine this rate. For one arm and the identity-link readout, let S_ℓ be the saturated space of square-integrable functions of the prefix code $z_{1:\ell}$, and let $A_\ell = \{\sum_{j\le\ell} g_j(z_j)\}$ be the compact additive space used by RQ. Write $\delta_\ell = \mathrm{dist}(r^*, A_\ell)$,

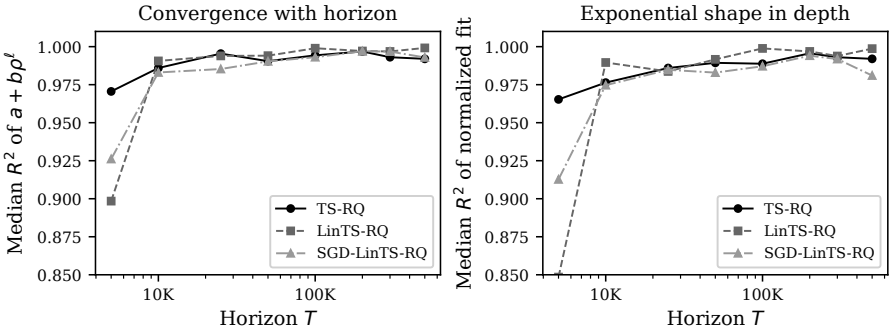


Figure 4: Empirical behavior consistent with the conditional geometric depth-rate law. **Left:** Median R^2 of the exponential fit $a + b\rho^\ell$ improves with horizon T. **Right:** Normalized regret decay $(\Delta_\ell - \Delta_5)/(\Delta_1 - \Delta_5)$ at terminal T for each dataset; curves cluster along exponential trajectories.

$\delta_\ell^{\mathrm{sat}} = \mathrm{dist}(r^*, S_\ell)$, and $\iota_\ell = \mathrm{dist}(P_{S_\ell} r^*, A_\ell)$. Let $A_\infty = \overline{\bigcup_\ell A_\ell}$, $\delta_\infty = \mathrm{dist}(r^*, A_\infty)$, and $\tau_\ell = \mathrm{dist}(P_{A_\infty} r^*, A_\ell)$.

Theorem 3.2 (Depth-rate decomposition)**.** *If r^* is L-Lipschitz and $\mathrm{MSE}_\ell = \mathbb{E}\|x - \hat{x}_\ell\|^2$ is the depth-ℓ RQ reconstruction error, then*

$$\delta_\ell^2 = (\delta_\ell^{\mathrm{sat}})^2 + \iota_\ell^2, \qquad (\delta_\ell^{\mathrm{sat}})^2 \le L^2 \mathrm{MSE}_\ell, \qquad \iota_\ell \le \delta_\infty + \tau_\ell. \tag{4}$$

Consequently,

$$\delta_\ell^2 - \delta_\infty^2 \le L^2 \mathrm{MSE}_\ell + 2\delta_\infty \tau_\ell + \tau_\ell^2. \tag{5}$$

Corollary 3.1 (Conditional geometric decay)**.** *If $\mathrm{MSE}_\ell \le M_0 \rho_x^\ell$ and $\tau_\ell \le C_\tau \rho_\tau^\ell$, with $\rho_x, \rho_\tau \in (0, 1)$, then*

$$\delta_\ell \le \delta_\infty + L\sqrt{M_0}\, \rho_x^{\ell/2} + C_\tau \rho_\tau^\ell = \delta_\infty + O(\rho^\ell), \tag{6}$$

where $\rho = \max\{\sqrt{\rho_x}, \rho_\tau\} < 1$.

The proof is in Appendix C. Reconstruction controls the saturated b^ℓ-cell benchmark, while τ_ℓ measures the additional interaction tail imposed by the compact $O(\ell b)$ additive readout. Thus geometric reconstruction decay alone is insufficient; geometric additive-tail decay is the explicit additional condition. For multiple arms, $\mathbb{E}\max_a |e_a| \le (\sum_a \mathbb{E} e_a^2)^{1/2}$, so the per-arm result transfers to Definition 3.1 up to a $\sqrt{K}$ factor. A Lipschitz output link similarly transfers a pre-link score bound after multiplication by its Lipschitz constant.

Figure 4 is consistent with the conditional geometric law: fitting $a + b\rho^\ell$ to regret rates at depths 1–5 yields median $R^2 = 0.99$ across 39 (dataset, method) pairs. This finite-depth evidence does not establish the assumptions of Corollary 3.1 universally. Within-cell reward variance contracts with median per-level ratio 0.83 (Appendix E).

3.3 Shadow false-promotion control

Theorem 3.3 (Shadow false-promotion control)**.** *The probability that the shadow is promoted despite having no lower expected squared prediction error than the active model, under the active policy's data distribution over the tested window, is at most α over the full horizon.*

We test the paired squared-error differences via Freedman's martingale inequality (Freedman, 1975), which applies without requiring i.i.d. observations. Given a finite planning horizon T (or an upper bound on the deployment length), the number of check times per depth is $n_c = \lfloor \log_2(T/T_0) \rfloor$, known before the first observation. The per-test significance level is Bonferroni-corrected over all $n_c \cdot (\ell_{\max} - 1)$ comparisons, giving total false-promotion probability $\le \alpha$. Proof in Appendix D.

Table 2: Model memory at shadow-promoted depths (b=16). XGB replay buffer grows $O(T)$; all RQ methods are fixed $O(1)$.

	TS-RQ	**LinTS-RQ**	**XGB replay**
higgs (T=5M)	12 KB	620 KB	1.1 GB
susy (T=5M)	22 KB	176 KB	800 MB
covertype (T=530K)	60 KB	10 MB	51 MB

4 Experiments

4.1 Setup

Following the supervised-to-bandit protocol (Riquelme et al., 2018), we convert datasets by assigning reward 1 for the correct class and 0 otherwise. We evaluate on 13 tabular datasets from OpenML and UCI with N ranging from 130K to 11M, feature dimension $d \in [3, 90]$, and $K \in [2, 104]$ arms (Table 6 in Appendix G). Following the codebook training protocol (Section 2), n_{rq}=50K contexts are held out for codebook training; the bandit runs on the remaining pool with contexts drawn uniformly at random without replacement, each seen once, so the horizon $T = N - n_{\text{rq}}$. All methods use b=16 centroids per level, initial depth ℓ_0=2, $\ell_{\max}$=8, ν=0.1, η=0.5, λ=1.0, shadow α=0.05, W=1000 (trailing window), $T_0 = 50 \cdot b \cdot K$, and 30 bandit seeds per configuration. For LinTS-RQ, we scale the per-level regularization as $\lambda_i = \lambda \cdot 0.25^{i-1}$ and add numerical jitter $10^{i-1} \times 10^{-6}$ for Cholesky stability. We fix all hyperparameters across all datasets—no per-dataset tuning. Code is available at `https://github.com/atavory/rqb`. Each seed initializes all random number generators (arm selection, posterior sampling) deterministically.

We compare six methods: three baselines (TS, SGD-LinTS, LinTS) and their RQ variants with adaptive depth (TS-RQ, SGD-LinTS-RQ, LinTS-RQ). We additionally compare against three nonlinear baselines: Doubling-XGB (XGBoost: 100 trees, max depth 6, learning rate 0.1, subsample 0.8, colsample_bytree 0.8, ϵ=0.1 greedy, doubling retrain schedule), Doubling-NeuralCB (2-layer MLP, 128 hidden units, ReLU activations, Adam optimizer with learning rate 10^{-3}, batch size 256, doubling retrain schedule, LinTS head), and TabNet-LinTS (Arik and Pfister, 2021) (unsupervised TabNet pre-training with default hyperparameters: N_{steps}=3, N_a=N_d=8, γ=1.3, λ_{sparse}=10^{-3}; encoder frozen on the same 50K holdout, LinTS on learned features). All representation-based methods (RQ, TabNet, Neural-Linear) use the same 50K unlabeled holdout; the baselines (TS, LinTS, SGD-LinTS) operate without any representation layer by design—they have no mechanism to consume unlabeled data—and evaluate on the same T-context pool, ensuring identical evaluation conditions. We fix all hyperparameters—both for RQ methods and for the nonlinear baselines (XGBoost, NeuralCB, TabNet)—across all 13 datasets with no per-dataset tuning.

We report regret *rate* (cumulative regret divided by T) throughout, making results length-invariant across datasets. Win/loss calls use 95% CIs from 30 seeds; a method "wins" only when the CI excludes zero.

4.2 Results

Memory–quality tradeoff. Figure 3 places all methods on a memory–regret scatter, and Table 2 gives the details. Regret is normalized by the context-free oracle rate ($1 - \max_a P(Y{=}a)$), so 1.0 corresponds to majority-class guessing. Values above 1.0 occur when squared-loss exploration produces overconfident wrong predictions on skewed datasets. Doubling-XGB and Doubling-NeuralCB match LinTS-RQ regret but require $O(T)$ replay memory (Table 2).

Within-family comparisons. Figure 2 shows the regret gap of each RQ method relative to its within-class baseline, and Table 3 gives the details. LinTS-RQ wins significantly on 10 of 13 datasets, with gaps up to 96% (skin_segmentation); SGD-LinTS-RQ wins on 12 of 13; TS-RQ wins on 11 of 13 versus context-free TS. The common failure case is year_prediction (d=90, K=84), where all three RQ variants show small losses (+0.4% to +4.5%); we discuss this in the data sufficiency analysis below.

Table 3: Win/loss/tie (W/L/T) summary across 13 datasets (30 seeds, 95% CIs). Negative gap = RQ method has lower regret.

Comparison	W/L/T	Best win	Worst loss
LinTS-RQ vs LinTS	10W/1L/2T	skin_seg -96%	year_pred $+4.5\%$
SGD-RQ vs SGD-LinTS	12W/1L	skin_seg -97%	year_pred $+0.7\%$
TS-RQ vs TS	11W/2L	kddcup99 -97%	year_pred $+0.4\%$

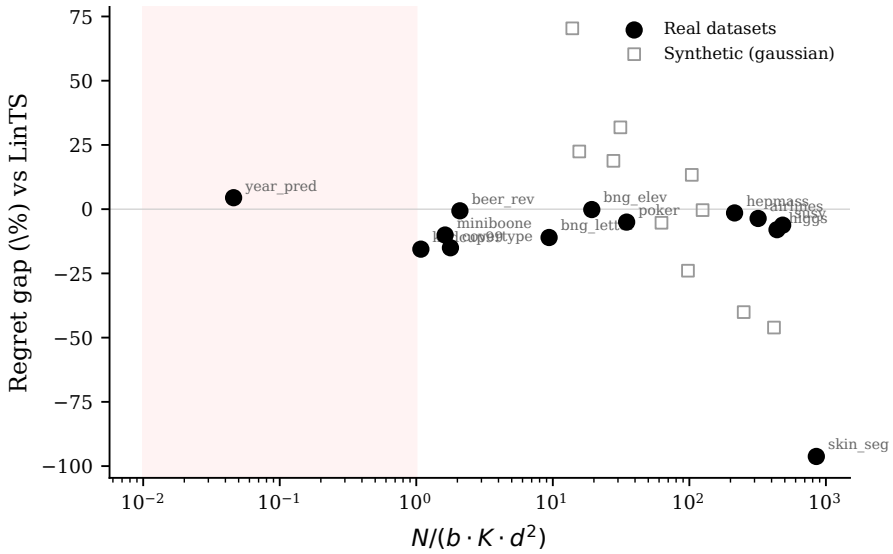


Figure 5: Data sufficiency diagnostic: regret gap of LinTS-RQ vs. LinTS as a function of $N/(bKd^2)$. All real datasets with ratio > 1 (outside the shaded region) are wins or ties; year_prediction (ratio $= 0.04$) is the sole loss. Synthetic Gaussian data (gray squares) confirm the threshold (Appendix K).

Nonlinear baselines. LinTS-RQ achieves regret competitive with Doubling-XGB (7W/4L/2T). The key differentiator is memory: at T=5M, LinTS-RQ uses 620 KB vs. XGB's 1.1 GB—a gap that grows linearly with T. Against TabNet-LinTS (same 50K holdout), LinTS-RQ wins on 11 of 13 datasets (11W/2L), with losses only on hepmass ($+0.5\%$) and year_prediction ($+1.7\%$), showing that unsupervised k-means produces a partition that matches or exceeds TabNet's neural embedding, while the RQ codebook trains in seconds via k-means versus minutes for TabNet pretraining. Per-step latency is also low: TS-RQ runs at 50–750 μs/step across all datasets (Appendix Q).

Data sufficiency. LinTS-RQ maintains a $d \times d$ covariance matrix for each of the bK centroid-arm pairs, so the total parameter count scales as bKd^2. The ratio $N/(bKd^2)$ measures contexts per parameter and is computable before any bandit interaction (Figure 5). On 12 of 13 datasets this ratio exceeds 1.1 and LinTS-RQ wins; on year_prediction (d=90, K=84, ratio=0.04), LinTS-RQ degrades even at depth 1. TS-RQ needs only $O(1)$ parameters per cell (a count and a reward sum), so it is unaffected: it matches the TS baseline on year_prediction (93.0% vs 93.4%). The failure is structural, not fixable by tuning: LinTS-RQ degrades with depth (96.3% at ℓ=1, 97.4% at ℓ=4) as each level fragments the already-sparse data further, while TS-RQ stays flat at 93.0% across all depths. On synthetic data, LinTS-RQ wins at all ratios ≥ 14 (Appendix K).

Effective rank. On datasets with effective rank ($\mathrm{tr}(\Sigma)/\lambda_{\max}(\Sigma)$) below 1.5 (skin_segmentation, miniboone), SGD-LinTS performs *worse than context-free TS*: the diagonal covariance approximation spreads its d parameters uniformly across coordinates, wasting $d-1$ of them on noise when most variance lies on a single axis. TS-RQ avoids this failure mode entirely—the codebook geometry captures dominant directions through centroid placement, without estimating any covariance—and achieves 2.6% and 14.7% normalized regret on these two datasets (Appendix N).

Table 4: TS-RQ regret rate (%) at fixed vs. adaptive depth.

	ℓ=1	ℓ=2	**Adaptive**
covertype	48.3%	42.2%	38.8% (depth 4)
higgs	45.4%	43.6%	43.7% (depth 3)

Depth adaptation. Table 4 compares TS-RQ at fixed depths against shadow-promoted adaptive depth. On covertype, shadow promotion discovers depth 4 and gains 3.4 percentage points over fixed ℓ=2. On higgs, the benefit beyond depth 2 is marginal, and the adaptive method correctly settles at depth 3 with matching regret: the marginal gain from depth 4 falls within the noise floor of the Freedman test, so the shadow is never promoted.

5 Related Work

Linear and GLM bandits. LinTS (Agrawal and Goyal, 2013) and LinUCB (Li et al., 2010; Abbasi-Yadkori et al., 2011; Dani et al., 2008) achieve $O(d\sqrt{KT \log T})$ regret under linear rewards. The misspecification bound (Lattimore et al., 2020) gives $\Delta \cdot T$ linear regret under misspecification—the basis for our asymptotic analysis (Appendix C).

Neural and tree-based bandits. NeuralUCB (Zhou et al., 2020) and Neural TS (Zhang et al., 2021) require per-step gradient and inverse-Hessian computation; efficient approximations (EE-Net (Ban and He, 2022), Neural-Linear (Riquelme et al., 2018)) reduce to LinTS on last-layer features. TETS (Nilsson et al., 2024) combines XGBoost with TS but must refit on accumulated history, incurring $O(T^2)$ cumulative cost. Our offline codebook avoids both: no gradients, no refitting.

Partition-based bandits. The X-armed bandit framework (Bubeck et al., 2011) and the Zooming algorithm (Kleinberg et al., 2008; Slivkins, 2014) adaptively refine partitions to balance resolution against sample complexity. Nonparametric histogram bandits (Rigollet and Zeevi, 2010; Perchet and Rigollet, 2013) achieve regret $O(T^{(d+2)/(d+1)})$ with flat partitions. Tile coding (Sutton and Barto, 2018) uses additive partitions in RL but with uniform tilings; our data-driven RQ codebook adapts to $P(X)$. The additive parameterization uses ℓb parameters for b^ℓ cells—a potentially exponential compression.

Model selection and quantization. CORRAL (Agarwal et al., 2017) and Regret Balancing (Pacchiano et al., 2020) run multiple base learners in parallel. Our shadow promotion monitors one additional depth level sequentially without influencing arm selection (Theorem 3.3). Product quantization (Jegou et al., 2010), residual quantization (Chen et al., 2010), and VQ-VAE (van den Oord et al., 2017) compress representations; we show the same structure compresses *exploration* (b^ℓ cells from ℓb parameters). Self-supervised methods (SCARF (Bahri et al., 2022), VIME (Yoon et al., 2020), SubTab (Ucar et al., 2021), TabNet (Arik and Pfister, 2021)) learn representations from unlabeled data; our codebook achieves this via unsupervised k-means.

6 Conclusion

We repurposed RQ codebooks as a representation layer for contextual bandits, enabling gradient-boosted reward models with fixed $O(1)$ memory. RQ variants improve on most datasets at orders-of-magnitude less memory than replay-based methods, while TS-RQ remains robust when LinTS-RQ's covariance starves. Extensions to continuous rewards, online codebook adaptation, and alternative per-level features (PCA, neural-linear) are straightforward.

7 Limitations

The data sufficiency condition $N/(bKd^2) \gg 1$ is the sharpest limitation: on year_prediction (d=90, K=84, ratio 0.04), LinTS-RQ degrades even at depth 1; a p-dimensional projection ($p \ll d$) would relax this to $N/(bKp^2)$. The linear variants use squared loss for binary rewards (Pólya-Gamma augmentation could eliminate this). Shadow promotion provides false-promotion control, though post-promotion performance depends on the new policy's data distribution (Remark D.2). The fixed-depth

asymptotic analysis (Appendix C) is conditional on base-learner convergence properties; the appendix makes these assumptions explicit and verifies them for our model classes. The codebook is trained via unsupervised k-means, which minimizes reconstruction error rather than reward prediction; a reward-aware partition (e.g., CART splits) could reduce the misspecification Δ_ℓ directly, at the cost of requiring labeled data for codebook construction. The codebook requires an offline holdout; in cold-start settings it may degrade. As with all bandit methods, reward-signal biases may be amplified; our method neither exacerbates nor mitigates this.

8 Broader Impact

This work introduces a general-purpose bandit representation and does not target a specific application domain. Standard fairness and privacy concerns for bandit algorithms apply; we do not foresee risks specific to residual quantization.

Acknowledgments and Disclosures

This work received no external funding. The authors declare no competing interests.

References

Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, volume 24, pages 2312–2320, 2011.

Alekh Agarwal, Haipeng Luo, Behnam Neyshabur, and Robert E. Schapire. Corralling a band of bandit algorithms. In *Conference on Learning Theory*, pages 12–38, 2017.

Shipra Agrawal and Navin Goyal. Analysis of Thompson Sampling for the multi-armed bandit problem. In *Conference on Learning Theory*, pages 39.1–39.26, 2012.

Shipra Agrawal and Navin Goyal. Thompson sampling for contextual bandits with linear payoffs. In *International Conference on Machine Learning*, pages 127–135. PMLR, 2013.

Sercan O. Arik and Tomas Pfister. TabNet: Attentive interpretable tabular learning. In *AAAI Conference on Artificial Intelligence*, volume 35, pages 6679–6687, 2021.

Dara Bahri, Heinrich Jiang, Yi Tay, and Donald Metzler. SCARF: Self-supervised contrastive learning using random feature corruption. In *International Conference on Learning Representations*, 2022.

Pierre Baldi, Peter Sadowski, and Daniel Whiteson. Searching for exotic particles in high-energy physics with deep learning. *Nature Communications*, 5:4308, 2014. doi: 10.1038/ncomms5308.

Pierre Baldi, Kyle Cranmer, Taylor Faucett, Peter Sadowski, and Daniel Whiteson. Parameterized neural networks for high-energy physics. *The European Physical Journal C*, 76(5):235, 2016. doi: 10.1140/epjc/s10052-016-4099-4.

Yikun Ban and Jingrui He. EE-Net: Exploitation-exploration neural networks in contextual bandits. In *International Conference on Learning Representations*, 2022.

Thierry Bertin-Mahieux, Daniel P.W. Ellis, Brian Whitman, and Paul Lamere. The million song dataset. In *International Society for Music Information Retrieval Conference*, 2011.

Jock A. Blackard and Denis J. Dean. Comparative accuracies of artificial neural networks and discriminant analysis in predicting forest cover types from cartographic variables. *Computers and Electronics in Agriculture*, 24(3):131–151, 1999. doi: 10.1016/S0168-1699(99)00046-0.

Sébastien Bubeck, Rémi Munos, Gilles Stoltz, and Csaba Szepesvári. X-armed bandits. *Journal of Machine Learning Research*, 12:1655–1695, 2011.

Yongjian Chen, Tao Guan, and Cheng Wang. Approximate nearest neighbor search by residual vector quantization. *Sensors*, 10(12):11259–11273, 2010. doi: 10.3390/s101211259.

Varsha Dani, Thomas P Hayes, and Sham M Kakade. Stochastic linear optimization under bandit feedback. In *Conference on Learning Theory*, pages 355–366, 2008.

Dheeru Dua and Casey Graff. UCI machine learning repository, 2019. URL `https://archive.ics.uci.edu/ml`.

David A. Freedman. On tail probabilities for martingales. *The Annals of Probability*, 3(1):100–118, 1975. doi: 10.1214/aop/1176996452.

Jerome H. Friedman. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29(5):1189–1232, 2001. doi: 10.1214/aos/1013203451.

Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 2nd edition, 2013. ISBN 978-0521548236.

Herve Jegou, Matthijs Douze, and Cordelia Schmid. Product quantization for nearest neighbor search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1):117–128, 2010. doi: 10.1109/TPAMI.2010.57.

Robert Kleinberg, Aleksandrs Slivkins, and Eli Upfal. Multi-armed bandits in metric spaces. In *ACM Symposium on Theory of Computing*, pages 681–690, 2008.

Tor Lattimore and Csaba Szepesvári. *Bandit Algorithms*. Cambridge University Press, 2020.

Tor Lattimore, Csaba Szepesvari, and Gellert Weisz. Learning with good feature representations in bandits and in RL with a generative model. In *International Conference on Machine Learning*, pages 5662–5671. PMLR, 2020.

Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 661–670, 2010.

Hannes Nilsson, Rikard Johansson, Niklas Åkerblom, and Morteza Haghir Chehreghani. Tree ensembles for contextual bandits. *Transactions on Machine Learning Research*, 2024.

Aldo Pacchiano, Christoph Dann, Claudio Gentile, and Peter Bartlett. Regret bound balancing and elimination for model selection in bandits and RL. *arXiv preprint arXiv:2012.13045*, 2020.

Vianney Perchet and Philippe Rigollet. The multi-armed bandit problem with covariates. *Annals of Statistics*, 41(2):693–721, 2013. doi: 10.1214/13-AOS1101.

Philippe Rigollet and Assaf Zeevi. Nonparametric bandits with covariates. In *Proceedings of the 23rd Conference on Learning Theory*, pages 54–66, 2010.

Carlos Riquelme, George Tucker, and Jasper Snoek. Deep Bayesian bandits showdown: An empirical comparison of Bayesian deep networks for Thompson sampling. In *International Conference on Learning Representations*, 2018.

Byron P. Roe, Hai-Jun Yang, Ji Zhu, Yong Liu, Ion Stancu, and Gordon McGregor. Boosted decision trees as an alternative to artificial neural networks for particle identification. *Nuclear Instruments and Methods in Physics Research Section A*, 543(2–3):577–584, 2005. doi: 10.1016/j.nima.2004.12.018.

Aleksandrs Slivkins. Contextual bandits with similarity information. *Journal of Machine Learning Research*, 15:2533–2568, 2014.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2nd edition, 2018.

Talip Ucar, Ehsan Hajiramezanali, and Lindsay Edwards. SubTab: Subsetting features of tabular data for self-supervised representation learning. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *Advances in Neural Information Processing Systems*, pages 6306–6315, 2017.

Jinsung Yoon, Yao Zhang, James Jordon, and Mihaela van der Schaar. VIME: Extending the success of self- and semi-supervised learning to tabular data. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Weitong Zhang, Dongruo Zhou, Lihong Li, and Quanquan Gu. Neural Thompson Sampling. In *International Conference on Learning Representations*, 2021.

Dongruo Zhou, Lihong Li, and Quanquan Gu. Neural contextual bandits with UCB-based exploration. In *International Conference on Machine Learning*, pages 11492–11502. PMLR, 2020.

A Base Learner Details

This appendix specifies the per-cell operations for each model class $\mathcal{M}$ in Algorithm 1. All three variants share the same outer loop; they differ only in how each cell $\mathcal{M}[i, c, a]$ maintains its posterior and produces predictions. Each cell exposes three operations: `sample`(x') returns a Thompson-sampling draw (used for arm selection in the **Score** step); `mean`(x') returns the posterior mean prediction (used to build the cumulative prediction $\hat{y}_i$ in the **Targets** step); `update`($x', \tilde{r}$) incorporates a new observation. Here $x' = x^{(i-1)}$ is the context residual from (1) and $\tilde{r}$ is the pseudo-target from Algorithm 1, line 6. For the linear variants (SGD-LinTS-RQ, LinTS-RQ), a constant intercept feature is appended: $x' \leftarrow [x^{(i-1)}; 1]$, so the effective per-centroid dimension is $d{+}1$. This ensures $\|\phi_i(x)\| \geq 1$ (Assumption C.1), which is needed for the exploration guarantee even when deep-level residuals are small.

Algorithm 2 TS-RQ cell operations (count/sum)

Require: Regularization λ; exploration scale ν

State: count $n \leftarrow 0$; sum $s \leftarrow 0$
function SAMPLE(x') ▷ x' is ignored
  $\hat{\mu} \leftarrow s\,/(\lambda + n)$
  return sample from $\mathcal{N}\big(\hat{\mu},\ \nu^2/(\lambda + n)\big)$
end function
function MEAN(x') ▷ x' is ignored
  return $s\,/(\lambda + n)$
end function
function UPDATE(x', $\tilde{r}$) ▷ x' is ignored
  $n \leftarrow n + 1$; $s \leftarrow s + \tilde{r}$
end function

Algorithm 3 SGD-LinTS-RQ cell operations (diagonal linear)

Require: Regularization λ; exploration scale ν

State: diagonal precision $p \in \mathbb{R}^{d+1} \leftarrow \lambda\mathbf{1}$; sufficient statistic $f \in \mathbb{R}^{d+1} \leftarrow \mathbf{0}$ ▷ $x' \leftarrow [x'; 1]$ (intercept appended)
function SAMPLE(x')
  $\hat{\theta} \leftarrow f \oslash p$ ▷ element-wise division
  $\tilde{\theta} \sim \mathcal{N}\Big(\hat{\theta},\ \nu^2\,\mathrm{diag}(p)^{-1}\Big)$
  return $\tilde{\theta}^\top x'$
end function
function MEAN(x')
  return $(f \oslash p)^\top x'$
end function
function UPDATE(x', $\tilde{r}$)
  $p \leftarrow p + x' \odot x'$ ▷ element-wise
  $f \leftarrow f + \tilde{r} \cdot x'$
end function

Algorithm 4 LinTS-RQ cell operations (full-covariance LinTS)

Require: Regularization $\lambda_i = \lambda \cdot 0.25^{i-1}$; exploration scale ν

State: inverse precision $B^{-1} \in \mathbb{R}^{(d+1)\times(d+1)} \leftarrow \lambda_i^{-1} I$; $f \in \mathbb{R}^{d+1} \leftarrow \mathbf{0}$ ▷ $x' \leftarrow [x'; 1]$ (intercept appended)

function SAMPLE(x')

  $\hat{\theta} \leftarrow B^{-1} f$

  $\tilde{\theta} \sim \mathcal{N}\left(\hat{\theta},\, \nu^2 B^{-1}\right)$

  return $\tilde{\theta}^\top x'$

end function

function MEAN(x')

  return $(B^{-1} f)^\top x'$

end function

function UPDATE(x', $\tilde{r}$) ▷ Sherman–Morrison on B^{-1}

  $v \leftarrow B^{-1} x'$

  $B^{-1} \leftarrow B^{-1} - v\, v^\top / (1 + x'^\top v)$

  $f \leftarrow f + \tilde{r} \cdot x'$

end function

B Base Learner Properties

Algorithm 1 is parameterized by a per-cell base learner $\mathcal{M}$. Theorem 3.1 (§3) establishes asymptotic regret under three properties of $\mathcal{M}$; the full proof is in Appendix C. We state these properties below and establish them for each variant. We use the notation from Appendix C: $\phi_i(x)$ is the level-i feature vector, $\mu_a(x_t)$ the score mean (22), $\mathrm{Var}(s_a \mid x_t, \mathcal{F}_{t-1})$ the score variance (23), $T_{a,c,i}(t)$ the visit count for arm a at centroid c and level i, and $\tilde{r}_{i,t}$ the per-level target (24).

Property 1 (Eventual boundedness). Score means are eventually bounded almost surely:

$$\exists_{t_0, B}\ \forall_{a,\, t>t_0}\ |\mu_a(x_t)| \le B \qquad \text{a.s.} \tag{7}$$

Property 2 (Starvation ⇒ positive variance). If arm a visits centroid c at level i at most M times ($T_{a,c,i}(t) \le M$ for all t), then

$$\mathrm{Var}(s_a \mid x_t, \mathcal{F}_{t-1}) \ge \sigma^2_{\min} > 0 \qquad \forall\, t \text{ with } k_i(x_t) = c, \tag{8}$$

where $\sigma^2_{\min}$ depends on M, λ, ν, C but not on t.

Property 3 (Robust consistency). Suppose all centroids at level i are visited infinitely often by arm a: $T_{a,c,i}(t) \to \infty$ for all c with $P(k_i(x) = c) > 0$. Suppose the targets have vanishing perturbation: $\tilde{r}_{i,t} = \tilde{r}^\infty_{i,t} + \delta_t$ with $\delta_t \to 0$ a.s., where $\tilde{r}^\infty_{i,t}$ is the stationary target under converged shallower levels. Then predictions converge:

$$\phi_i(x)^\top \hat{\theta}_{a,i,t} \;\to\; \phi_i(x)^\top \theta^*_{a,i} \qquad \text{a.s. for all } x \text{ in the support}, \tag{9}$$

where $\theta^*_{a,i}$ is the best predictor of $\tilde{r}^\infty_i$ from $\phi_i(x)$ in the function class of $\mathcal{M}$.

B.1 Verification for LinTS-RQ

LinTS-RQ maintains a full-covariance precision $V_{a,i,t} = \lambda_i I + \sum_{s \le t:\, a_s = a} \phi_i(x_s)\, \phi_i(x_s)^\top$ and posterior mean $\hat{\theta}_{a,i,t} = V^{-1}_{a,i,t} \sum_{s: a_s = a} \phi_i(x_s)\, \tilde{r}_{i,s}$.

Property 1. For any arm a, level i, and centroid c: if $T_{a,c,i}(t) \le M$ for all t, the estimate receives at most M updates with bounded targets ($|\tilde{r}| \le 1$), so $\|\hat{\theta}_{a,i,c}\| \le MC/\lambda$ (since $\|V^{-1}\| \le 1/\lambda$ and $\|\sum \phi \tilde{r}\| \le MC$), giving $|\hat{\theta}^\top_{a,i,c} \phi_i(x)| \le MC^2/\lambda$, deterministically bounded. If $T_{a,c,i}(t) \to \infty$, the strong law gives $\hat{\theta}_{a,i,c} \to \theta^*_{a,i,c}$, which is finite. Since the number of arm/centroid combinations is finite, score means are eventually bounded.

Property 2. If $T_{a,c,i}(t) \le M$ for all t, the centroid-c block of $V_{a,i,t}$ has at most M rank-one updates, giving the time-uniform bound $V_{a,i,t}\big|_c \preceq (\lambda + MC^2)I$. When $k_i(x_t) = c$:

$$\mathrm{Var}(s_a \mid x_t, \mathcal{F}_{t-1}) \ge \nu^2\, \phi_i(x_t)^\top V_{a,i,t}^{-1} \phi_i(x_t) \ge \frac{\nu^2}{\lambda + MC^2} =: \sigma_{\min}^2 > 0. \tag{10}$$

Property 3. Suppose $T_{a,c,i}(t) \to \infty$ for all centroids c and targets satisfy $\tilde{r}_{i,s} = \tilde{r}_{i,s}^{\infty} + \delta_s$ with $\delta_s \to 0$. The ridge estimator decomposes as

$$\hat{\theta}_{a,i,t} = \underbrace{V_{a,i,t}^{-1} \sum_{s:a_s=a} \phi_i(x_s)\, \tilde{r}_{i,s}^{\infty}}_{\to\, \theta_{a,i}^*} + \underbrace{V_{a,i,t}^{-1} \sum_{s:a_s=a} \phi_i(x_s)\, \delta_s}_{\text{transient bias}}. \tag{11}$$

The first term converges by standard ridge consistency (stationary targets, $T_{a,c,i} \to \infty$). For the transient bias, apply Cauchy–Schwarz under $\|v\|_M = \sqrt{v^\top M\, v}$ with $M = V_{a,i,t}^{-1}$:

$$\left| \phi_i(x)^\top V_{a,i,t}^{-1} \sum_{s:a_s=a} \phi_i(x_s)\, \delta_s \right| \le \|\phi_i(x)\|_{V_{a,i,t}^{-1}} \cdot \left\| \sum_{s:a_s=a} \phi_i(x_s)\, \delta_s \right\|_{V_{a,i,t}^{-1}}. \tag{12}$$

First factor. Write $\hat{\Sigma}_t = \frac{1}{T_a(t)} \sum_{s:a_s=a} \phi_i(x_s)\, \phi_i(x_s)^\top$ and $\Sigma = \mathbb{E}[\phi_i(x)\phi_i(x)^\top \mid a]$. Since $V_{a,i,t} = T_a(t)\big(\frac{\lambda}{T_a(t)}\, I + \hat{\Sigma}_t\big)$:

$$\begin{aligned}
\|\phi_i(x)\|_{V_{a,i,t}^{-1}}^2 &= \phi_i(x)^\top V_{a,i,t}^{-1}\, \phi_i(x) \\
&= \frac{1}{T_a(t)}\, \phi_i(x)^\top \Big(\tfrac{\lambda}{T_a(t)}\, I + \hat{\Sigma}_t \Big)^{-1} \phi_i(x) \\
&\xrightarrow{\text{a.s.}} \frac{1}{T_a(t)}\, \phi_i(x)^\top \Sigma^{-1} \phi_i(x) \\
&\le \frac{c}{T_a(t)} \qquad \text{eventually,}
\end{aligned} \tag{13}$$

where the third line uses $\hat{\Sigma}_t \xrightarrow{\text{a.s.}} \Sigma$ (the strong law), and $c = 2C^2 / \min_c \lambda_{\min}(\Sigma_c) < \infty$. Each centroid block Σ_c is positive definite under a mild nondegeneracy condition: the conditional distribution of $\phi_i(x)$ given $k_i(x) = c$ has full-rank covariance. In practice, the intercept feature ensures at least one nonzero direction, and the RQ residuals within each centroid span the remaining coordinates when $T_{a,c}$ is large enough. If Σ_c is rank-deficient, the regularized estimator still converges in prediction space: since $\phi_i(x)$ for centroid c lies in the column space of Σ_c, the prediction $\phi_i(x)^\top \hat{\theta}$ converges regardless. Also, $\|\phi_i(x)\| \le C$ by Assumption C.1.

Second factor. Write $\Phi \in \mathbb{R}^{bp \times T_a(t)}$ for the matrix whose columns are $\phi_i(x_s)$ and $\Delta \in \mathbb{R}^{T_a(t)}$ for the vector of drifts δ_s. By the push-through identity $(\lambda I + AB)^{-1} A = A(\lambda I + BA)^{-1}$:

$$\begin{aligned}
\left\| \sum_{s:a_s=a} \phi_i(x_s)\, \delta_s \right\|_{V_{a,i,t}^{-1}}^2 &= \Delta^\top \Phi^\top (\lambda I + \Phi\Phi^\top)^{-1} \Phi\, \Delta \\
&= \Delta^\top \underbrace{(\lambda I + \Phi^\top \Phi)^{-1} \Phi^\top \Phi}_{\preceq I}\, \Delta \\
&\le \|\Delta\|^2 = \sum_{s:a_s=a} \delta_s^2.
\end{aligned} \tag{14}$$

Combining.

$$\left| \phi_i(x)^\top V_{a,i,t}^{-1} \sum_{s:a_s=a} \phi_i(x_s)\, \delta_s \right|^2 \le \frac{c}{T_a(t)} \sum_{s:a_s=a} \delta_s^2. \tag{15}$$

Since $\delta_s \to 0$ and $T_a(t) \to \infty$, the Cesàro mean $\frac{1}{T_a(t)} \sum_{s:a_s=a} \delta_s^2 \to 0$, so the transient bias prediction vanishes. □

B.2 Verification for SGD-LinTS-RQ

SGD-LinTS-RQ maintains a diagonal precision $p \in \mathbb{R}^d$ with $p_k = \lambda + \sum_{s:a_s=a} x'^2_{s,k}$ and sufficient statistic $f_k = \sum_{s:a_s=a} \tilde{r}_s\, x'_{s,k}$, yielding the coordinate-wise estimator $\hat{\theta}_k = f_k/p_k$. This is d independent scalar regressions, one per feature coordinate.

Property 1. Same structure as LinTS-RQ: bounded updates when starved, SLLN when explored, applied coordinate-wise.

Property 2. If $T_{a,c,i}(t) \le M$, then $p_{a,i,k,t} \le \lambda + MC^2$ for every coordinate k, giving

$$\mathrm{Var}(s_a \mid x_t, \mathcal{F}_{t-1}) = \sum_{j=1}^{\ell} \nu^2 \sum_{k=1}^{d+1} \frac{(\phi_{j,k})^2}{p_{a,j,k,t}} \ge \frac{\nu^2 \|\phi_i(x_t)\|^2}{\lambda + MC^2} \ge \frac{\nu^2}{\lambda + MC^2} > 0 \tag{16}$$

by Assumption C.1, where the sum is over $d{+}1$ coordinates (including the intercept that ensures $\|\phi_i(x)\| \ge 1$ even when context residuals are small at deep levels).

Property 3. By the strong law applied to each coordinate independently:

$$\hat{\theta}_{a,i,k,t} \to \theta^*_{a,i,k} \coloneqq \frac{\mathbb{E}[\tilde{r}_i^{\infty} \cdot x'_k]}{\mathbb{E}[x'^2_k]}, \tag{17}$$

the coordinate-wise optimal coefficient. The transient bias at coordinate k satisfies

$$|f_k^{\mathrm{bias}}/p_k| \le C\, \frac{1}{T_a(t)} \sum_{s:a_s=a} |\delta_s| \to 0 \tag{18}$$

by the Cesàro argument. Note that $\theta^*_{a,i}$ is the coordinate-wise optimal, which may differ from the full-linear optimal when features are correlated; consequently $\Delta_\ell(\mathcal{M})$ for SGD-LinTS-RQ is evaluated over the coordinate-wise linear function class. □

B.3 Verification for TS-RQ

TS-RQ is the scalar special case: the per-cell model ignores features and maintains only a count n and sum s (Algorithm 2). The estimate $\hat{\mu} = s/(\lambda + n)$ is a scalar ridge regression with constant feature 1. All three properties follow from the LinTS-RQ checks with $d = 1$ and $\phi_i(x) = 1$. □

Remark B.1 (Adaptive sampling). The checks above invoke the strong law of large numbers for per-centroid sample averages. Under Thompson Sampling, the data arriving at each centroid is not i.i.d.: arm selection depends on the posterior, which depends on all past observations. However, each centroid's samples form a *conditionally* unbiased sequence: conditioned on $a_t = a$ and $k_i(x_t) = c$, the target $\tilde{r}_{i,t}$ has the same conditional expectation regardless of how a_t was selected. The key requirement is that every centroid is visited infinitely often (Lemma C.1(a)), which ensures the denominators grow. Given infinite visits, the Cesàro convergence argument applies to the prediction error regardless of the adaptive selection mechanism, since the per-step bias δ_s vanishes and the averaging is over a growing number of bounded terms. This is the standard approach in the TS literature (Agrawal and Goyal, 2013, 2012); a fully formal treatment under adaptive sampling remains open for misspecified TS in general.

C Fixed-Depth Regret Analysis

This appendix develops the full fixed-depth analysis for Algorithm 1, including the asymptotic regret theorem and the depth-rate decomposition of Theorem 3.2.

Assumption C.1. *All per-level feature vectors are bounded:* $1 \le \|\phi_i(x)\| \le C$ *for all* i, x.

The lower bound $\|\phi_i(x)\| \ge 1$ is ensured by appending a constant intercept feature to each $\phi_i(x)$.

Definition 3.1 (Additive misspecification, restated from §3). The additive misspecification $\Delta_\ell(\mathcal{M})$ is defined in (2). Since deeper models contain shallower ones ($f_{\ell+1} \equiv 0$), $\Delta_{\ell+1} \le \Delta_\ell$.

Theorem 3.1 (Asymptotic regret, restated from §3). *For any base learner* $\mathcal{M}$ *satisfying Properties 1–3 (Appendix B), the asymptotic regret rate satisfies* (3)*, where* $\hat{r}_\infty$ *is the converged prediction of the sequential residual-fitting limit (Lemma C.1(b)). Since* $\hat{r}_\infty$ *is one feasible additive predictor, the right-hand side is* $\ge 2\,\Delta_\ell(\mathcal{M})$*, with equality when the stagewise and joint optima coincide.*

Lemma C.1 (Exploration and convergence)**.** *For any base learner $\mathcal{M}$ satisfying Properties 1–3, for each arm a and level $i \in \{1, \ldots, \ell\}$:*

(a) *Every arm visits every centroid at level i infinitely often: $T_{a,c}(T) \to \infty$ a.s. for all centroids c with $P(k_i(x) = c) > 0$.*

(b) *The per-level predictions converge: $\phi_i(x)^\top \hat{\theta}_{a,i,t} \to \phi_i(x)^\top \theta^*_{a,i}$ for all x in the support.*

The proof proceeds by induction on the level i. Each level's targets depend on all shallower levels' estimates, creating a coupled non-stationary system. Properties 1–3 (Appendix B) abstract this coupling for three distinct base learners. The per-level posterior can be viewed as a block-diagonal approximation of the joint posterior analyzed in Appendix F.

Notation. Each feature vector $\phi_j(x)$ is supported on a single centroid block at level j (placing the context residual $x^{(j-1)}$ at centroid $k_j(x)$ and zeros elsewhere). The TS draw, score, and selection for arm a at context x_t are

$$\forall_a \tilde{\theta}_{a,j} \sim \mathcal{M}[j,\, k_j,\, a].\text{sample}(x_t^{(j-1)}), \tag{19}$$

$$\forall_a s_a = \sum_{j=1}^{\ell} \tilde{\theta}_{a,j}^\top \phi_j(x_t), \tag{20}$$

$$a_t = \arg\max_a s_a = \arg\max_a \sigma(s_a) \tag{21}$$

independently across levels; the last equality holds since σ is monotone. The *score mean* and conditional variance are

$$\mu_a(x_t) = \mathbb{E}[s_a \mid x_t, \mathcal{F}_{t-1}] = \sum_{j=1}^{\ell} \hat{\theta}_{a,j,t}^\top \phi_j(x_t), \tag{22}$$

$$\text{Var}(s_a \mid x_t, \mathcal{F}_{t-1}) = \sum_{j=1}^{\ell} \nu^2\, \phi_j(x_t)^\top V_{a,j,t}^{-1} \phi_j(x_t). \tag{23}$$

The reward prediction is $\hat{y}_a = \sigma(\mu_a) \in [0, 1]$, and the per-level target (Algorithm 1, line 6) is

$$\tilde{r}_{j,t} = r_t - \sigma\left(\sum_{i=1}^{j-1} \hat{\theta}_{a_t,i,t}^\top \phi_i(x_t)\right) \in [-1, 1]. \tag{24}$$

For clarity, the damping factor η from Algorithm 1 is set to 1 in the analysis; absorbing η^{j-1} into ϕ_j does not affect the proof structure, though it scales the effective variance at level j by $\eta^{2(j-1)}$, which slows exploration at deeper levels when $\eta < 1$. The asymptotic result is unaffected (the lower bound $\sigma^2_{\min}$ remains positive), but the pre-asymptotic exploration cost at deep levels grows as $\eta^{-2(j-1)}$.

Lemma C.2 (Score means are eventually bounded)**.** *All arm score means are eventually bounded:*

$$\exists_{t_0, B} \forall_{a,\, t > t_0} |\mu_a(x_t)| \le B \qquad \textit{a.s.} \tag{25}$$

Proof. Immediate from Property 1 (Appendix B). □

Lemma C.3 (Exploration and convergence, restated)**.** *For any base learner $\mathcal{M}$ satisfying Properties 1–3 (Appendix B), for each arm a and level $i \in \{1, \ldots, \ell\}$:*

(a) *Every arm visits every centroid at level i infinitely often: $T_{a,c}(T) \to \infty$ a.s. for all centroids c with $P(k_i(x) = c) > 0$.*

(b) *The per-level predictions converge: $\phi_i(x)^\top \hat{\theta}_{a,i,t} \to \phi_i(x)^\top \theta^*_{a,i}$ for all x in the support, where $\theta^*_{a,i}$ is the best linear predictor for the clipped residual at level i given converged shallower levels.*

Proof of Lemma C.1. The proof is by induction on the level i and applies to any base learner $\mathcal{M}$ satisfying Properties 1–3.

Induction base ($i = 1$).

(a) Per-centroid exploration. Assume that for some a, c,

$$T_{a,c}(T) \le M < \infty, \tag{26}$$

$$P(k_1(x) = c) =: p_c > 0. \tag{27}$$

By Property 2, the score variance satisfies

$$\mathrm{Var}(s_a \mid x_t, \mathcal{F}_{t-1}) \ge \sigma^2_{\min} > 0 \qquad \forall\, t \text{ with } k_1(x_t) = c. \tag{28}$$

By Lemma C.2, there exist (a.s.) finite B and t_0 such that every arm's score mean satisfies $|\mu_{a'}(x_t)| \le B$ for all $t \ge t_0$. For $t \ge t_0$ with $k_1(x_t) = c$, the probability that arm a is selected satisfies

$$\begin{aligned} P(a_t = a \mid x_t, \mathcal{F}_{t-1}) &\ge P(s_a > B) \prod_{a' \ne a} P(s_{a'} \le B) \\ &\ge \left[1 - \Phi(2B/\sigma_{\min})\right] \cdot (\tfrac{1}{2})^{K-1} =: p_{\min} > 0, \end{aligned} \tag{29}$$

where the first factor uses anti-concentration of the score distribution: all three variants use Gaussian sampling (Algorithms 2–4), so s_a is Gaussian with variance $\ge \sigma^2_{\min}$ and mean $\ge -B$, giving the stated normal tail bound. Each factor in the product is $\ge \frac{1}{2}$ because $\mu_{a'} \le B$. Since $k_1(x_t) = c$ occurs infinitely often (i.i.d. contexts with $p_c > 0$) and each occurrence satisfies $P(a_t = a \mid x_t, \mathcal{F}_{t-1}) \ge p_{\min}$ (a uniform conditional lower bound, not independence), the Lévy extension of the Borel–Cantelli lemma gives $T_{a,c} \to \infty$, contradicting (27).

(b) Prediction convergence. The targets $\tilde{r}_{1,t} = r_t - \sigma(0)$ are stationary (no dependence on shallower levels), so $\delta_t = 0$. By (a), $T_{a,c}(T) \to \infty$ for every centroid c. Property 3 (with $\delta_t = 0$) gives $\phi_1(x)^\top \hat{\theta}_{a,1,t} \to \phi_1(x)^\top \theta^*_{a,1}$.

Induction step ($i \ge 2$). Assume (a) and (b) hold for levels $1, \dots, i-1$.

(a) Per-centroid exploration. Similar to the induction base: if $T_{a,c}(T) \le M$ for some centroid c at level i, Property 2 gives $\mathrm{Var}(s_a) \ge \sigma^2_{\min} > 0$ when $k_i(x_t) = c$. By Lemma C.2, score means are eventually bounded, so Gaussian anti-concentration and Borel–Cantelli yield the contradiction $T_{a,c} \to \infty$.

(b) Prediction convergence. The clipped residual target for level i is

$$\tilde{r}_{i,t} = r_t - \sigma\left(\sum_{j=1}^{i-1} \hat{\theta}^\top_{a,j,t} \phi_j(x_t)\right). \tag{30}$$

Since σ is 1-Lipschitz and levels $1, \dots, i-1$ have converged in prediction space by the induction hypothesis (b), the target drift satisfies

$$|\tilde{r}_{i,t} - \tilde{r}^{\infty}_{i,t}| \le \sum_{j=1}^{i-1} \left|(\hat{\theta}_{a,j,t} - \theta^*_{a,j})^\top \phi_j(x_t)\right| =: \delta_t \to 0, \tag{31}$$

where each summand vanishes by the induction hypothesis. By (a), $T_{a,c,i}(t) \to \infty$ for all centroids. Property 3 (with $\delta_t \to 0$) gives $\phi_i(x)^\top \hat{\theta}_{a,i,t} \to \phi_i(x)^\top \theta^*_{a,i}$.

□

Proof of Theorem 3.1 (Asymptotic regret for Algorithm 1). By Lemma C.1(b), for each arm a, the converged prediction is

$$\hat{r}_\infty(x, a) = \sigma\left(\sum_{i=1}^{\ell} \theta^{*\top}_{a,i} \phi_i(x)\right). \tag{32}$$

Define

$$\hat{a}(x) = \arg\max_a \hat{r}_\infty(x, a) \tag{33}$$

as the arm selected under the converged model.

Post-convergence regret. Once predictions have converged, TS selects $\hat{a}(x_t)$ at every round (the posterior variance vanishes; model-level ties $\hat{r}_\infty(x,a) = \hat{r}_\infty(x,a')$ occur on a measure-zero set of contexts under continuous features, and contribute at most $2\Delta_\ell$ regret per round regardless). The per-round regret is

$$\begin{aligned} r^*(x_t, a_t^*) - r^*(x_t, \hat{a}(x_t)) = &\left[r^*(x_t, a_t^*) - \hat{r}_\infty(x_t, a_t^*)\right] \\ &- \left[r^*(x_t, \hat{a}) - \hat{r}_\infty(x_t, \hat{a})\right] \\ &+ \underbrace{\left[\hat{r}_\infty(x_t, a_t^*) - \hat{r}_\infty(x_t, \hat{a})\right]}_{\leq 0}, \end{aligned} \tag{34}$$

where the last term is ≤ 0 because $\hat{a}$ maximizes $\hat{r}_\infty$. Therefore

$$r^*(x_t, a_t^*) - r^*(x_t, \hat{a}(x_t)) \leq 2 \max_a |r^*(x_t, a) - \hat{r}_\infty(x_t, a)|\,. \tag{35}$$

Taking expectation over x_t:

$$\mathbb{E}[r^*(x_t, a_t^*) - r^*(x_t, \hat{a}(x_t))] \leq 2\,\mathbb{E}_x\Big[\max_a |r^*(x,a) - \hat{r}_\infty(x,a)|\Big] =: 2\,\bar{\Delta}_\ell(\mathcal{M}), \tag{36}$$

where $\bar{\Delta}_\ell$ is the approximation error of the sequential residual-fitting limit $\hat{r}_\infty(x,a) = \sigma(\sum_i \theta_{a,i}^{*\top}\phi_i(x))$. Since $\hat{r}_\infty$ is one feasible additive predictor, $\bar{\Delta}_\ell \geq \Delta_\ell$ (Definition 3.1), with equality when the stagewise and joint optima coincide.

Exploration regret. Before convergence, TS may select suboptimal arms. By Lemma C.1(b), $\hat{\theta}_{a,i,t} \to \theta_{a,i}^*$ for all arms and levels (in prediction space), so the posterior means converge. Since the posterior variances vanish ($\nu^2 V_{a,i,t}^{-1} \to 0$ as $T_a(t) \to \infty$ by Lemma C.1(a)), the TS draws converge:

$$\tilde{\theta}_{a,i,t} \to \theta_{a,i}^* \tag{37}$$

a.s., and arm selection converges to $\hat{a}(x_t)$. For contexts where the optimal arm gap $\hat{r}_\infty(x, \hat{a}(x)) - \max_{a \neq \hat{a}(x)} \hat{r}_\infty(x,a) > 0$, the TS draws eventually resolve to the correct arm; exact ties have zero probability under continuous TS draws. For any $\epsilon > 0$, there exists T_0 such that for $t > T_0$, $a_t = \hat{a}(x_t)$ except on a set of measure $< \epsilon$. The exploration regret (rounds where $a_t \neq \hat{a}(x_t)$) therefore satisfies

$$\sum_{t=1}^{T} \mathbf{1}[a_t \neq \hat{a}(x_t)] = o(T). \tag{38}$$

Since per-round regret is bounded by 1, the cumulative exploration regret is $o(T)$.

Total.

$$\frac{R(T,\ell)}{T} = \underbrace{\frac{o(T)}{T}}_{\to 0} + \underbrace{2\,\bar{\Delta}_\ell(\mathcal{M})}_{\text{from (36)}}, \tag{39}$$

giving $\limsup_{T\to\infty} R(T,\ell)/T \leq 2\,\bar{\Delta}_\ell(\mathcal{M}) = 2\,\mathbb{E}_x[\max_a |r^* - \hat{r}_\infty|]$. □

C.1 Proof of the depth-rate result

Fix one arm and let $\mathcal{H} = L^2(P)$. Because each RQ symbol takes finitely many values, A_ℓ is a closed finite-dimensional subspace of the closed saturated space S_ℓ. Denote their orthogonal projections by P_{A_ℓ} and P_{S_ℓ}.

Proof of Theorem 3.2. Since $A_\ell \subseteq S_\ell$, $P_{A_\ell} = P_{A_\ell} P_{S_\ell}$. We may therefore write

$$r^* - P_{A_\ell} r^* = \big(r^* - P_{S_\ell} r^*\big) + \big(P_{S_\ell} r^* - P_{A_\ell} r^*\big). \tag{40}$$

The first summand lies in $S_\ell^\perp$ and the second in S_ℓ. Pythagoras gives the first identity in (4). For the reconstruction bound, conditional expectation is the L^2 projection onto S_ℓ, so

$$(\delta_\ell^{\mathrm{sat}})^2 = \mathbb{E}[\mathrm{Var}(r^*(x) \mid z_{1:\ell})]\,. \tag{41}$$

Within a prefix cell c, let x, x' be independent draws from $P(\cdot \mid c)$. Lipschitz continuity yields

$$\mathrm{Var}(r^*(x) \mid c) = \tfrac{1}{2}\mathbb{E}[(r^*(x) - r^*(x'))^2 \mid c] \tag{42}$$

$$\leq \tfrac{L^2}{2}\mathbb{E}[\|x - x'\|^2 \mid c] = L^2 \,\mathrm{Var}(x \mid c). \tag{43}$$

The conditional mean minimizes squared reconstruction error, whereas $\hat{x}_\ell$ is merely constant on each prefix cell. Hence $\mathrm{Var}(x \mid c) \leq \mathbb{E}[\|x - \hat{x}_\ell\|^2 \mid c]$. Averaging over cells proves the second inequality in (4).

Finally, put $a^* = P_{A_\infty} r^*$ and $r_\perp = r^* - a^*$. Then $\|r_\perp\| = \delta_\infty$. For any $a_\ell \in A_\ell$, projection contraction gives

$$\mathrm{dist}(P_{S_\ell} r^*, A_\ell) \leq \|P_{S_\ell} r^* - a_\ell\| \tag{44}$$

$$\leq \|P_{S_\ell}(a^* - a_\ell)\| + \|P_{S_\ell} r_\perp\| \tag{45}$$

$$\leq \|a^* - a_\ell\| + \delta_\infty. \tag{46}$$

Taking the infimum over a_ℓ proves the interaction-tail bound. Combining it with the first two claims gives (5). □

Proof of Corollary 3.1. Theorem 3.2 and $\sqrt{u^2 + v^2} \leq u + v$ for $u, v \geq 0$ imply

$$\delta_\ell \leq L\sqrt{\mathrm{MSE}_\ell} + \delta_\infty + \tau_\ell. \tag{47}$$

Substitution of the two geometric assumptions gives the stated bound with $\rho = \max\{\sqrt{\rho_x}, \rho_\tau\}$. □

D Promotion Proofs

Theorem 3.3 (Shadow false-promotion control, restated)**.** *The probability that the shadow is promoted despite having no lower expected squared prediction error than the active model, under the active policy's data distribution over the tested window, is $\leq \alpha$ over the full horizon.*

Shadow promotion differs from CORRAL (Agarwal et al., 2017) in a structurally important way: CORRAL runs M learners simultaneously and pays $\tilde{O}(\sqrt{M})$ regret overhead from model selection; our shadow trains passively and incurs *zero* regret overhead during evaluation. The regret during each phase is bounded by Theorem 3.1 at the *active* depth, with no model-selection penalty. We test the paired squared-error differences via Freedman's inequality (Freedman, 1975) (no i.i.d. assumption needed).

Remark D.1 (Freedman vs. t-test). The formal guarantee (Theorem 3.3) requires the Freedman-based test, which makes no i.i.d. assumption on the paired differences D_t. In our experiments, we also evaluate a paired t-test variant as a practical heuristic; because the D_t within each trailing window are not i.i.d. (the policy evolves), the t-test p-values are not formally calibrated, though empirically both tests yield similar promotion decisions.

Remark D.2 (Post-promotion covariate shift). The formal guarantee is false-promotion control (Type I error $\leq \alpha$). Post-promotion, the policy changes and arm frequencies shift. For all three variants this shift is structurally bounded: centroid assignments depend only on x_t and the frozen codebook, and TS ensures all cells are visited infinitely often (Lemma C.1(a)).

Proof of Theorem 3.3 (Shadow false-promotion control). Martingale formulation. Since the shadow updates only its own statistics (line 13) and does not appear in the scoring (line 4) or arm selection (line 5), the action sequence is identical to Algorithm 1 at fixed depth. To decide whether the shadow model genuinely outperforms the active model, we construct a statistical test. At each round t, define the paired squared-error difference between the active and shadow predictions. Let $\hat{y}_{\mathrm{act}}(x_t, a_t)$ and $\hat{y}_{\mathrm{shd}}(x_t, a_t)$ denote the posterior mean predictions of the active and shadow models, respectively. Define:

$$D_t = (r_t - \hat{y}_{\mathrm{act}}(x_t, a_t))^2 - (r_t - \hat{y}_{\mathrm{shd}}(x_t, a_t))^2 . \tag{48}$$

When $D_t > 0$, the shadow's prediction was closer to the observed reward at round t.

Since the shadow does not influence a_t, the conditional expectation $\mathbb{E}[D_t \mid \mathcal{F}_{t-1}]$ is a deterministic function of the models' states. Define the centered differences:

$$Z_t = D_t - \mathbb{E}[D_t \mid \mathcal{F}_{t-1}]. \tag{49}$$

By construction, $\mathbb{E}[Z_t \mid \mathcal{F}_{t-1}] = 0$, so $(Z_t)_{t\geq 1}$ is a martingale difference sequence adapted to $(\mathcal{F}_t)$. Rewards lie in $\{0, 1\}$. Since σ maps to $[0, 1]$, the posterior mean predictions satisfy $\hat{y}_{\text{act}}, \hat{y}_{\text{shd}} \in [0, 1]$ for all three variants. Each squared error is therefore in $[0, 1]$, so $|D_t| \leq 1 =: R_D$ and:

$$|Z_t| \leq |D_t| + |\mathbb{E}[D_t \mid \mathcal{F}_{t-1}]| \leq 2R_D \quad \text{a.s.} \tag{50}$$

Freedman-based promotion test. Fix a check time τ and a trailing window of W observations. The cumulative difference decomposes into signal and noise:

$$\bar{D}_W = \sum_{t=\tau-W+1}^{\tau} D_t = \underbrace{\sum_{t=\tau-W+1}^{\tau} \mathbb{E}[D_t \mid \mathcal{F}_{t-1}]}_{\text{signal: } \mu_W} + \underbrace{\sum_{t=\tau-W+1}^{\tau} Z_t}_{\text{noise: } M_W}. \tag{51}$$

The signal μ_W is positive when the shadow is genuinely better (in expected squared error) over the window. We need to distinguish $\mu_W > 0$ from noise fluctuations.

The noise $M_W = \sum_t Z_t$ is a sum of martingale differences with $|Z_t| \leq R \coloneqq 2R_D$. Its predictable quadratic variation is:

$$V_W = \sum_{t=\tau-W+1}^{\tau} \mathbb{E}\big[Z_t^2 \mid \mathcal{F}_{t-1}\big] \leq R^2 W. \tag{52}$$

By Freedman's inequality (Freedman, 1975):

$$P(M_W \geq a \text{ and } V_W \leq v) \leq \exp\left(\frac{-a^2/2}{v + Ra/3}\right). \tag{53}$$

To derive the threshold a^*, we set $v = R^2 W$ (the worst-case variance bound) and require the right-hand side of (53) to equal the per-test significance level α_k. This gives the equation:

$$\exp\left(\frac{-a^2/2}{R^2 W + Ra/3}\right) = \alpha_k. \tag{54}$$

Taking logarithms and writing $L = \log(1/\alpha_k)$:

$$\frac{a^2}{2} = L\left(R^2 W + \frac{Ra}{3}\right) \quad \Longrightarrow \quad a^2 - \frac{2RL}{3}\, a - 2R^2 W L = 0. \tag{55}$$

Applying the quadratic formula ($a > 0$):

$$\begin{aligned} a^*(\alpha_k, W) &= \frac{RL}{3} + \sqrt{\frac{R^2 L^2}{9} + 2R^2 W L} \\ &= \frac{R}{3}\log\frac{1}{\alpha_k} + R\sqrt{\frac{1}{9}\left(\log\frac{1}{\alpha_k}\right)^2 + 2W\log\frac{1}{\alpha_k}}. \end{aligned} \tag{56}$$

We promote if $\bar{D}_W > a^*(\alpha_k, W)$.

Under the null hypothesis (shadow is not better, $\mu_W \leq 0$), the signal is non-positive, so from (51):

$$\bar{D}_W = \mu_W + M_W \leq M_W. \tag{57}$$

Therefore, a promotion requires $M_W > a^*$, which by Freedman's inequality (53) occurs with probability at most α_k:

$$P(\text{promote} \mid \mu_W \leq 0) \leq P(M_W > a^*) \leq \alpha_k. \tag{58}$$

Here, a^* is the critical value that makes the noise term unlikely to exceed the threshold, ensuring false promotions are rare.

Bonferroni correction. The doubling schedule checks at times $\{T_0, 2T_0, 4T_0, \ldots\}$, producing at most n_c check intervals:

$$n_c = \lfloor \log_2(T/T_0) \rfloor + 1. \tag{59}$$

At each check time, the algorithm only evaluates the immediate shadow depth $k+1$ extending from the current active depth k. Therefore, it conducts exactly one hypothesis test per scheduled check interval. An additional test is only conducted if a promotion successfully advances the active frontier, prompting the algorithm to immediately evaluate the next depth. Note that this immediate re-test uses the trailing window W, which contains data collected under the previous depth's policy; the conservatism of the Freedman-based test (Theorem 3.3) provides robustness against this distribution shift. Because at most $\ell_{\max} - 1$ promotions can ever occur (from depth 1 to depth $\ell_{\max}$), the total number of hypothesis tests over the entire horizon is bounded by:

$$N \leq n_c + (\ell_{\max} - 1). \tag{60}$$

Setting $\alpha_k = \alpha/N$ for each individual test and applying the union bound:

$$P(\text{any false promotion}) \leq \sum_{k=1}^{N} \alpha_k = N \cdot \frac{\alpha}{N} = \alpha. \tag{61}$$

By tracking only the frontier tests, the multiplicity penalty is additive $\mathcal{O}(n_c + \ell_{\max})$ rather than multiplicative $\mathcal{O}(n_c \cdot \ell_{\max})$. The doubling schedule keeps $n_c = \mathcal{O}(\log T)$.

Combining. By (61), with probability at least $1 - \alpha$, *every* promotion is a true improvement under the active policy's data distribution:

$$\mu_W > 0, \tag{62}$$

meaning the shadow has strictly lower expected squared error on the (x_t, a_t) pairs generated by π_{act}. This is a *pre-promotion* guarantee; post-promotion, the policy changes and the data distribution shifts (see Remark D.2).

Regardless of whether any individual promotion is true or false, the shadow never influences arm selection, so the regret during each phase (at fixed depth ℓ_k) is bounded by Theorem 3.1. If promotions occur at times $T_1, T_2, \ldots$ with depths $\ell_1, \ell_2, \ldots$, the total regret is the sum across phases:

$$R(T) \leq \sum_{k} R(T_k - T_{k-1},\, \ell_k), \tag{63}$$

where each term is bounded by Theorem 3.1 at the corresponding depth. Since $\Delta_{\ell_{k+1}} \leq \Delta_{\ell_k}$ (deeper models have lower bias), promotions can only reduce the bias term in subsequent phases. □

E Depth Ablation and Exponential Decay

This appendix examines empirical behavior relevant to the conditional geometric rate in Corollary 3.1 (§3) and the depth results in §4.2.

E.1 Exponential Decay of Misspecification

Conditional depth-rate prediction. Corollary 3.1 gives geometric convergence to an irreducible floor when both RQ reconstruction error and the additive interaction tail contract geometrically.

To test whether the observed behavior is consistent with this prediction, we compute the regret rate $R(T)/T$ at the terminal round for each (dataset, method, depth) combination using depths $\ell = 1, \ldots, 5$. Since $R(T)/T \to O(\Delta_\ell)$ as $T \to \infty$ (Theorem 3.1), the converged regret rate provides a proxy for Δ_ℓ.

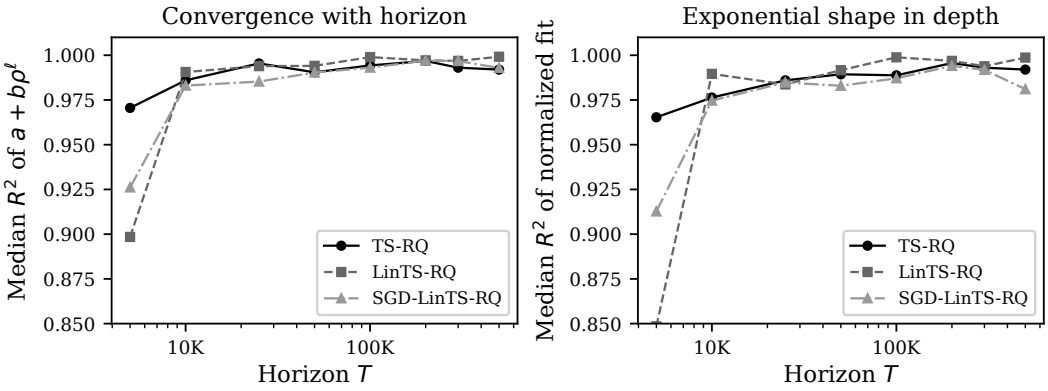


Figure 6: Empirical behavior consistent with the conditional geometric depth-rate law, one column per method. **Top:** Median R^2 of the exponential fit $a + b\rho^\ell$ as a function of horizon T; fit quality improves with more data. **Bottom:** Normalized regret decay $(\Delta_\ell - \Delta_5)/(\Delta_1 - \Delta_5)$ at terminal T for each dataset.

Exponential fit. We fit $R(T)/T \approx a + b\rho^\ell$ to each (dataset, method) pair via nonlinear least squares, yielding 39 fits (13 datasets $\times$ 3 RQ methods, 30 seeds each). The table below reports the median fit quality.

Metric	**Median**	**Mean**	> 0.90
R^2 of 3-parameter fit	0.99	0.94	24/28 (86%)

Leave-one-out cross-validation. For each (dataset, method), we hold out one depth, fit on the remaining four, and predict the held-out regret rate. This produces 135 predictions (some fits fail on 3 points). The median prediction error is 0.3%; 73% of predictions fall within 1% of the observed value (Figure 7).

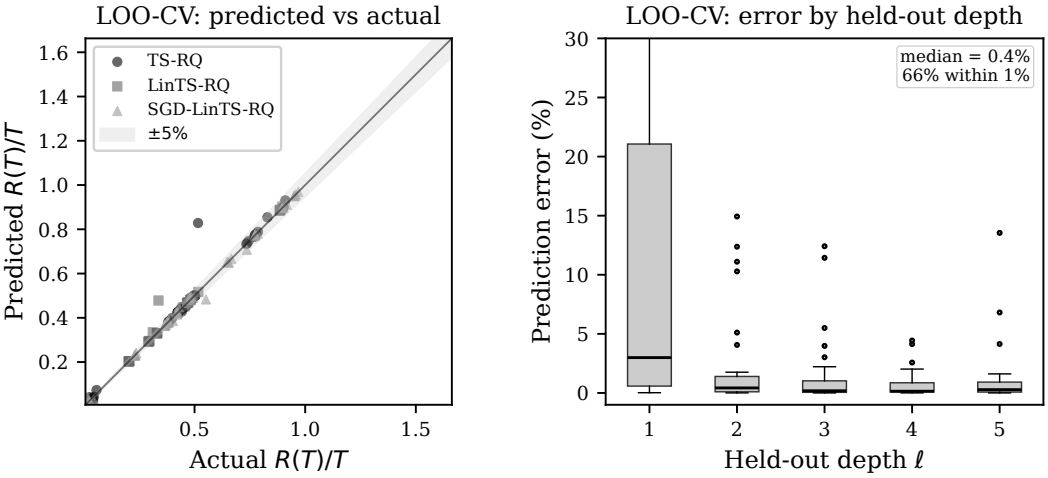


Figure 7: Leave-one-out cross-validation of the exponential decay model $\Delta_\ell \approx a + b\rho^\ell$. **Left:** Predicted vs. actual regret rate $R(T)/T$ for all (dataset, method, held-out depth) triples; the gray band shows $\pm 5\%$. **Right:** Prediction error by held-out depth; depths 2–5 are predicted within $\sim 1\%$, while depth 1 (furthest from Δ_∞) shows larger extrapolation error.

Exponential decay rate. Figure 9 fits $a + b\rho^\ell$ to the normalized within-cell variance $\mathrm{WCV}(\ell)/\mathrm{WCV}(0)$ for each dataset. The median fitted decay rate is $\rho = 0.86$ (range 0.17–0.93), with median $R^2 = 0.83$.

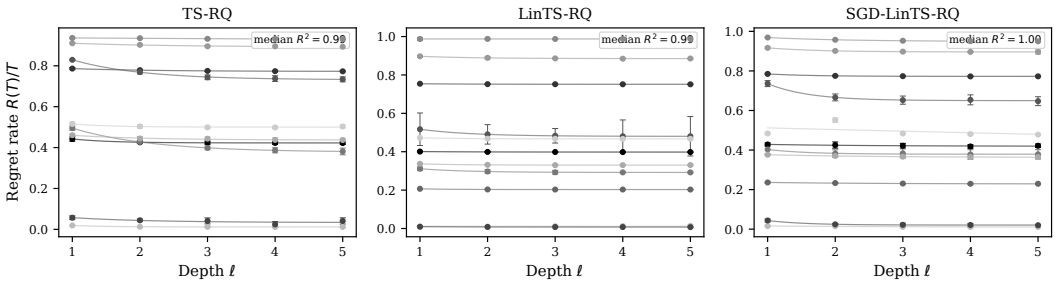


Figure 8: Regret rate $R(T)/T$ vs. depth ℓ for each RQ method, with fitted exponential decay curves $a + b\rho^{\ell}$ overlaid. Each line is a dataset; error bars show ± 1 std over 30 seeds. Median R^2 is reported per method.

E.2 Within-Cell Reward Variance

Figure 9 measures within-cell reward heterogeneity directly. For each dataset we train an RQ codebook (b=16, depth 5) and compute the weighted average within-cell reward variance $\mathrm{WCV}(\ell) = \sum_c (n_c/N)\,\mathrm{Var}(r \mid \mathrm{cell}_\ell{=}c)$, where cells at depth ℓ are defined by the first ℓ centroid assignments. On a log scale, all 13 datasets decay roughly linearly, consistent with geometric shrinkage over the measured depths, with median per-level ratio $\rho = 0.83$ (range 0.24–0.93).

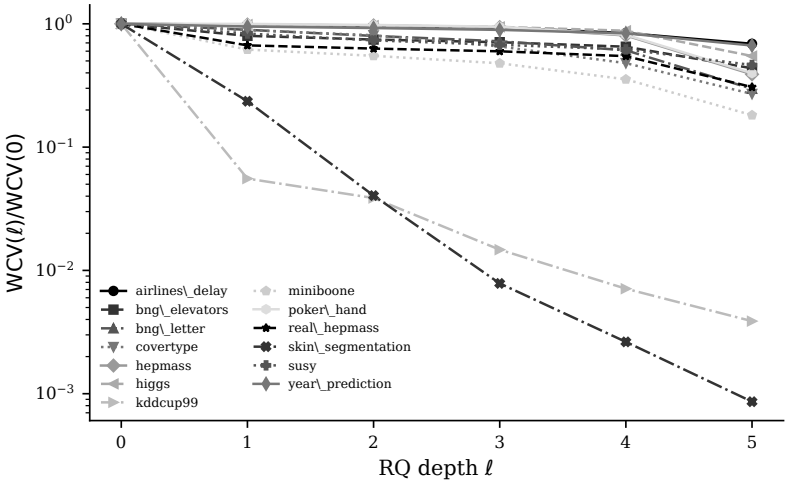


Figure 9: Normalized within-cell reward variance $\mathrm{WCV}(\ell)/\mathrm{WCV}(0)$ vs. RQ depth ℓ for all 13 datasets (log scale). Every dataset decays monotonically; the near-linear trajectories are consistent with geometric decay with median $\rho = 0.83$.

E.3 Fixed-Depth vs. Shadow Promotion

We compare TS-RQ at fixed depths ℓ=1 and ℓ=2 against the adaptive shadow-promoted variant (TS-RQ) on two representative datasets.

	ℓ=1	ℓ=2	**Adaptive**
covertype	48.3%	42.2%	38.8% (depth 4)
higgs	45.4%	43.6%	43.7% (depth 3)

On covertype, shadow promotion discovers depth 4 and gains 3.4 percentage points over fixed ℓ=2, confirming that adaptive depth selection adds value beyond simply committing to ℓ=2. On higgs, the benefit of depth beyond 2 is marginal, and the adaptive method correctly settles at depth 3 with regret matching ℓ=2.

F Analytical Variant: Finite-Time Bound

This appendix analyzes an analytical variant of Algorithm 1 that maintains the full $D \times D$ joint posterior covariance rather than the block-diagonal approximation used in practice. The variant admits an explicit finite-time regret bound (Proposition F.1 below), complementing the asymptotic guarantee for Algorithm 1 (Theorem 3.1). The finite-time bound is a standard application of LinTS analysis (Agrawal and Goyal, 2013) to our feature map; the contribution is not the proof technique but the *architectural insight* that the RQ additive structure ensures $D = b\ell p$ while achieving the approximation power of b^ℓ cells—an exponential compression not available to flat partitions.

Analytical variant details. The analytical variant (Proposition F.1) concatenates all per-level features into a block vector $\Phi_{\mathcal{M}}(x) \in \mathbb{R}^D$ with $D = b\ell\, p(\mathcal{M})$:

$$\Phi_{\mathcal{M}}(x) = \begin{pmatrix} \phi_1(x) \\ \phi_2(x) \\ \vdots \\ \phi_\ell(x) \end{pmatrix} \in \mathbb{R}^D, \qquad \Sigma_a = \begin{pmatrix} \Sigma_{11} & \cdots & \Sigma_{1\ell} \\ \vdots & \ddots & \vdots \\ \Sigma_{\ell 1} & \cdots & \Sigma_{\ell\ell} \end{pmatrix} \in \mathbb{R}^{D\times D}, \tag{64}$$

where each $\phi_i(x) \in \mathbb{R}^{bp}$ encodes the level-i assignment (one-hot for TS-RQ, context-at-centroid for LinTS-RQ) and $\Sigma_{ij} \in \mathbb{R}^{bp\times bp}$ captures cross-level correlations. A single parameter vector $\theta_a \in \mathbb{R}^D$ recovers the *unclipped* additive model: $r(x,a) = \theta_a^\top \Phi_{\mathcal{M}}(x)$.

The analytical variant runs LinTS on $\Phi_{\mathcal{M}}(x)$ with the full $D \times D$ posterior covariance Σ_a. Define the *variant misspecification* $\Delta(\Phi) = \mathbb{E}_x[\max_a |r^*(x,a) - \theta_a^{*\top}\Phi(x)|]$, where θ_a^* is the best linear predictor for arm a on Φ. Since $\theta_a^\top \Phi(x) = \sum_i f_i(k_i(x), a)$ recovers any unclipped additive model, $\Delta(\Phi) \le \Delta_\ell^{\text{uncl}}$ where $\Delta_\ell^{\text{uncl}}$ is the error of the stagewise limit without clipping. In practice, clipping (σ) can only reduce approximation error for bounded rewards, so $\Delta_\ell(\mathcal{M}) \le \Delta_\ell^{\text{uncl}}$, giving $\Delta(\Phi)$ and $\Delta_\ell(\mathcal{M})$ the same asymptotic scaling in depth.

Complexity comparison. Table 5 summarizes the exploration and memory costs of the analytical variant versus Algorithm 1.

Table 5: Exploration and memory scaling by base learner. "Joint" columns refer to the analytical variant; "Actual" is Algorithm 1.

$\mathcal{M}$	$p(\mathcal{M})$	Exploration	Joint memory	Actual memory
TS	1	$O(b\ell\sqrt{KT\log(T/\delta)})$	$O(b^2\ell^2 K)$	$O(b\ell K)$
LinTS	d	$O(b\ell d\sqrt{KT\log(T/\delta)})$	$O(b^2\ell^2 d^2 K)$	$O(b\ell d^2 K)$
SGD-LinTS	d	$O(b\ell d\sqrt{KT\log(T/\delta)})$	$O(b^2\ell^2 d^2 K)$	$O(b\ell d K)$

The analytical variant runs K independent LinTS instances (Agrawal and Goyal, 2013), one per arm, on the D-dimensional feature vector $\Phi_{\mathcal{M}}(x) \in \mathbb{R}^D$ with $D = b\ell\, p(\mathcal{M})$ (64). Each arm a maintains a ridge precision matrix

$$V_{a,t} = \lambda I + \sum_{s\le t:\, a_s = a} \Phi(x_s)\Phi(x_s)^\top \in \mathbb{R}^{D\times D}. \tag{65}$$

Let $r^*(x,a) = \mathbb{E}[r_t \mid x_t{=}x,\, a_t{=}a]$ denote the true expected reward, and let $\theta_a^* = \arg\min_\theta \mathbb{E}_x[(r^*(x,a) - \theta^\top\Phi(x))^2]$ be the best linear predictor for arm a. Define the misspecification function $\epsilon(x,a) = r^*(x,a) - \theta_a^{*\top}\Phi(x)$ and the observation noise $\eta_t = r_t - r^*(x_t,a_t)$. Note that $\Delta(\Phi) = \mathbb{E}_x[\max_a |\epsilon(x,a)|]$. The observed reward decomposes as

$$r_t = \theta_{a_t}^{*\top}\Phi(x_t) + \underbrace{\epsilon(x_t,a_t)}_{\text{bias}} + \underbrace{\eta_t}_{\text{noise}}. \tag{66}$$

At time t, the ridge posterior mean estimates θ_a^* by

$$\begin{aligned}\hat{\theta}_{a,t} &= V_{a,t}^{-1} \sum_{s:a_s=a} \Phi(x_s)\, r_s \\ &= \underbrace{V_{a,t}^{-1} \sum_{s:a_s=a} \Phi(x_s)(\theta_a^{*\top}\Phi(x_s) + \eta_s)}_{\hat{\theta}_{a,t}^{\mathrm{perf}}} \; + \; \underbrace{V_{a,t}^{-1} \sum_{s:a_s=a} \Phi(x_s)\, \epsilon(x_s, a_s)}_{\hat{\theta}_{a,t}^{\mathrm{bias}}},\end{aligned} \tag{67}$$

where the first term is the solution under perfect specification, and the second is the bias due to misspecification.

Lemma F.1 (Bias regret)**.** *For i.i.d. contexts,* $\mathbb{E}_x[\max_a |\epsilon(x,a)|] = \Delta(\Phi)$.

Proof. Immediate from the definitions of ϵ and $\Delta(\Phi)$. □

Lemma F.2 (Exploration regret under realizability)**.** *For the analytical variant under realizability ($\Delta(\Phi) = 0$, i.e. $\epsilon \equiv 0$), with bounded features $\|\Phi(x)\| \le C$, rewards in $[0,1]$, and D-dimensional parameter vectors, the cumulative regret satisfies, with probability $\ge 1-\delta$,*

$$R(T,\ell) = O\Big(D\sqrt{KT\log(T/\delta)}\Big). \tag{68}$$

Proof. Under realizability, $r^*(x,a) = \theta_a^{*\top}\Phi(x)$ for all (x,a), so the reward noise $\eta_t = r_t - \theta_{a_t}^{*\top}\Phi(x_t)$ is zero-mean conditional on (x_t, a_t). This is exactly the setting of Agrawal and Goyal (2013), Theorem 2 (stated for Gaussian noise; the extension to sub-Gaussian noise, which covers our bounded binary rewards, is standard (Lattimore and Szepesvári, 2020)), giving per-arm regret $O(D\sqrt{T_a \log(T/\delta)})$. Summing over arms ($\sum_a T_a = T$):

$$\begin{aligned}\sum_{a=1}^{K} O\Big(D\sqrt{T_a\log(T/\delta)}\Big) &= O\Big(D\sqrt{\log(T/\delta)}\Big)\sum_{a=1}^{K}\sqrt{T_a} \\ &\overset{\text{Jensen}}{\le} O\Big(D\sqrt{\log(T/\delta)}\Big)\sqrt{K\sum_{a=1}^{K} T_a} \\ &\overset{\sum T_a = T}{=} O\Big(D\sqrt{KT\log(T/\delta)}\Big).\end{aligned}$$

□

Remark F.1 (Variant regret (heuristic bound)). For the analytical variant with bounded features ($\|\Phi(x)\| \le C$) and rewards in $[0,1]$, under the heuristic decomposition below, with probability $\ge 1-\delta$,

$$\frac{R(T,\ell)}{T} \le \underbrace{O(\Delta(\Phi))}_{\text{approx. bias}} + \underbrace{O\left(\frac{D\sqrt{K\log(T/\delta)}}{\sqrt{T}}\right)}_{\text{exploration}} \xrightarrow{T\to\infty} O(\Delta(\Phi)). \tag{69}$$

Argument for Remark F.1. Let $a_t^* = \arg\max_a r^*(x_t, a)$. The per-round regret decomposes as

$$\begin{aligned}r^*(x_t,a_t^*) - r^*(x_t,a_t) = &\underbrace{\big[\theta_{a_t^*}^{*\top}\Phi(x_t) - \theta_{a_t}^{*\top}\Phi(x_t)\big]}_{\text{linear suboptimality}} \\ &+ \underbrace{\big[\epsilon(x_t,a_t^*) - \epsilon(x_t,a_t)\big]}_{\text{misspecification}}.\end{aligned} \tag{70}$$

The misspecification bracket satisfies $|\epsilon(x_t,a_t^*) - \epsilon(x_t,a_t)| \le 2\max_a|\epsilon(x_t,a)|$. Since contexts are i.i.d., Hoeffding's inequality gives $\sum_{t=1}^{T}\max_a|\epsilon(x_t,a)| \le T\cdot\Delta(\Phi) + O(\sqrt{T\log(1/\delta)})$ with probability $\ge 1-\delta$; the second term is absorbed into the exploration term below.

For the linear suboptimality, we apply Lemma F.2. Strictly, Lemma F.2 is stated under realizability ($\epsilon \equiv 0$); in the misspecified case, the systematic bias ϵ shifts the posterior mean, which can affect the algorithm's exploration behavior. For UCB-style algorithms, this additive decomposition is

standard (Lattimore and Szepesvári, 2020); for Thompson Sampling, the shifted posterior may alter the sampling distribution in non-additive ways. Following Lattimore et al. (2020), we treat the two terms as independent, yielding the bound below as an upper bound on the well-specified component of regret. The asymptotic guarantee (Theorem 3.1) is established independently via the property-based proof and does not rely on this decomposition. With $D = b\ell\, p(\mathcal{M})$:

$$R(T,\ell) \le \underbrace{O\Big(b\ell\, p(\mathcal{M})\sqrt{KT\log(T/\delta)}\Big)}_{\text{exploration}} + \underbrace{O(\Delta(\Phi)\cdot T)}_{\text{misspecification bias}}. \tag{71}$$

□

Variance deflation. The block-diagonal approximation used by Algorithm 1 inverts each level's precision block independently. The following lemma shows that this yields a posterior that is *tighter* than the exact marginal of the joint posterior above. For UCB-style algorithms, tighter confidence sets imply lower regret; for Thompson Sampling, the relationship between posterior width and regret is not monotone, so the finite-time bound above applies to the analytical variant only. The asymptotic guarantee for Algorithm 1 (Theorem 3.1) is established independently via the property-based proof in Appendix C.

Lemma F.3 (Marginal variance deflation)**.** *Let $\Sigma \succ 0$ be the joint posterior covariance over all ℓ levels* (64)*, partitioned into $bp \times bp$ blocks. For every level $i \in \{1,\dots,\ell\}$, $([\Sigma^{-1}]_{ii})^{-1} \preceq [\Sigma]_{ii}$.*

Proof. Partition $\Sigma^{-1} \succ 0$ into level i versus the remaining $\ell - 1$ levels:

$$\Sigma^{-1} = \begin{pmatrix} A & B \\ B^\top & C \end{pmatrix},$$

where $A = [\Sigma^{-1}]_{ii} \in \mathbb{R}^{bp\times bp}$ and $C \in \mathbb{R}^{(\ell-1)bp\times(\ell-1)bp}$. Since $\Sigma^{-1} \succ 0$, both A and C are positive definite. By the block inversion formula,

$$[\Sigma]_{ii} = (A - BC^{-1}B^\top)^{-1}. \tag{72}$$

Since $C \succ 0$, the Schur complement satisfies $BC^{-1}B^\top \succeq 0$, giving $A - BC^{-1}B^\top \preceq A$. Both sides are positive definite (the left-hand side by $\Sigma^{-1} \succ 0$), so inverting reverses the Loewner order (Horn and Johnson, 2013, Theorem 7.7.6):

$$([\Sigma^{-1}]_{ii})^{-1} = A^{-1} \preceq (A - BC^{-1}B^\top)^{-1} = [\Sigma]_{ii}. \qquad \square$$

G Dataset Details

Codebook training. We train the RQ codebook via scikit-learn's `MiniBatchKMeans` (k=16, `init='k-means++'`, `n_init=3`, `max_iter=100`, `batch_size=1024`, `random_state=42`) applied sequentially: we train level i on the context residuals $x^{(i-1)}$ from the n_{rq}=50K holdout contexts. All features are standardized to zero mean and unit variance before codebook training.

Preprocessing. We label-encode categorical features. We apply no feature selection, imputation, or augmentation.

Table 6: Datasets used in the evaluation. N = total samples, d = feature dimension, K = number of arms (classes), $p_{\max}$ = majority class frequency, eff. rank = effective rank of the feature covariance. All datasets are publicly available from OpenML and UCI (Dua and Graff, 2019); codebook trained on n_{rq}=50K held-out samples. Individual sources: HIGGS and SUSY (Baldi et al., 2014), HEPMASS (Baldi et al., 2016), covertype (Blackard and Dean, 1999), MiniBooNE (Roe et al., 2005), year_prediction (Bertin-Mahieux et al., 2011).

Dataset	N	d	K	$p_{\max}$	eff. rank
airlines_delay	500K	7	2	0.555	4.8
bng_elevators	1M	18	10	0.183	3.7
bng_letter	1M	16	26	0.041	4.6
covertype	581K	54	7	0.488	3.9
beer_reviews	500K	12	104	0.074	3.2
hepmass	5M	27	2	0.500	5.7
higgs	11M	28	2	0.530	6.7
kddcup99	494K	41	17	0.568	3.9
miniboone	130K	50	2	0.719	1.1
poker_hand	500K	10	9	0.502	9.7
skin_seg.	245K	3	2	0.793	1.3
susy	5M	18	2	0.542	3.7
year_prediction	500K	90	84	0.076	8.7

H Full Per-Dataset Results

Tables 7–9 show per-dataset results for all three method families.

Table 7: LinTS-RQ vs LinTS: per-dataset regret rates (%) and gap with 95% CIs.

Dataset	d	K	**Base (%)**	**RQ (%)**	**Gap (%)**	T
airlines	7	2	41.3	39.8	-3.6 ± 0.1	450K
beer_rev.	12	104	85.6	83.0	-2.9 ± 0.5	450K
bng_elev.	18	10	75.1	75.0	-0.2 ± 0.0	950K
bng_letter	16	26	52.4	46.6	-11.0 ± 0.2	950K
covertype	54	7	28.1	23.9	-15.0 ± 0.2	530K
hepmass	27	2	18.6	18.4	-1.1 ± 0.1	830K
higgs	28	2	35.5	32.6	-8.0 ± 0.1	6.7M
kddcup99	41	17	0.2	0.2	-16.0 ± 17.8	440K
miniboone	50	2	11.0	9.9	-10.1 ± 0.7	80K
poker	10	9	49.4	46.9	-5.0 ± 0.3	445K
skin_seg.	3	2	8.3	0.3	-96.1 ± 0.2	195K
susy	18	2	21.6	20.3	-6.1 ± 0.1	3.5M
year_pred.	90	84	93.6	97.8	$+4.5 \pm 0.1$	445K
11W / 1L / 1T						

Table 8: TS-RQ vs TS: per-dataset regret rates (%) and gap with 95% CIs.

Dataset	d	K	**Base (%)**	**RQ (%)**	**Gap (%)**	T
airlines	7	2	45.6	42.8	-6.2 ± 2.5	450K
beer_rev.	12	104	92.7	89.8	-3.2 ± 0.3	450K
bng_elev.	18	10	81.7	77.9	-4.6 ± 0.2	950K
bng_letter	16	26	96.0	71.4	-25.6 ± 0.5	950K
covertype	54	7	51.3	38.8	-24.4 ± 0.7	530K
hepmass	27	2	50.0	21.7	-56.6 ± 0.2	830K
higgs	28	2	48.0	43.7	-9.0 ± 1.6	6.7M
kddcup99	41	17	42.7	1.1	-97.4 ± 0.1	440K
miniboone	50	2	28.3	14.7	-48.1 ± 0.8	80K
poker	10	9	49.8	50.4	$+1.1 \pm 0.6$	445K
skin_seg.	3	2	20.8	2.6	-87.4 ± 2.4	195K
susy	18	2	46.6	24.8	-46.8 ± 1.1	3.5M
year_pred.	90	84	92.8	93.2	$+0.4 \pm 0.2$	445K
11W / 2L / 0T						

Table 9: SGD-RQ vs SGD-LinTS: per-dataset regret rates (%) and gap with 95% CIs.

Dataset	d	K	**Base (%)**	**RQ (%)**	**Gap (%)**	T
airlines	7	2	46.2	42.4	-8.2 ± 1.1	450K
beer_rev.	12	104	91.7	89.2	-2.8 ± 0.2	450K
bng_elev.	18	10	80.0	77.6	-3.0 ± 0.6	950K
bng_letter	16	26	81.6	64.3	-21.2 ± 0.7	950K
covertype	54	7	48.9	32.0	-34.6 ± 1.4	530K
hepmass	27	2	20.5	19.1	-7.1 ± 0.2	830K
higgs	28	2	49.0	36.2	-26.2 ± 2.3	6.7M
kddcup99	41	17	20.7	1.2	-94.3 ± 2.4	440K
miniboone	50	2	32.9	14.1	-57.3 ± 6.1	80K
poker	10	9	53.6	49.4	-7.8 ± 0.8	445K
skin_seg.	3	2	42.0	2.5	-94.2 ± 1.1	195K
susy	18	2	29.2	23.0	-21.2 ± 0.6	3.5M
year_pred.	90	84	93.3	94.0	$+0.7 \pm 0.1$	445K
12W / 1L / 0T						

I Per-Dataset Regret Curves

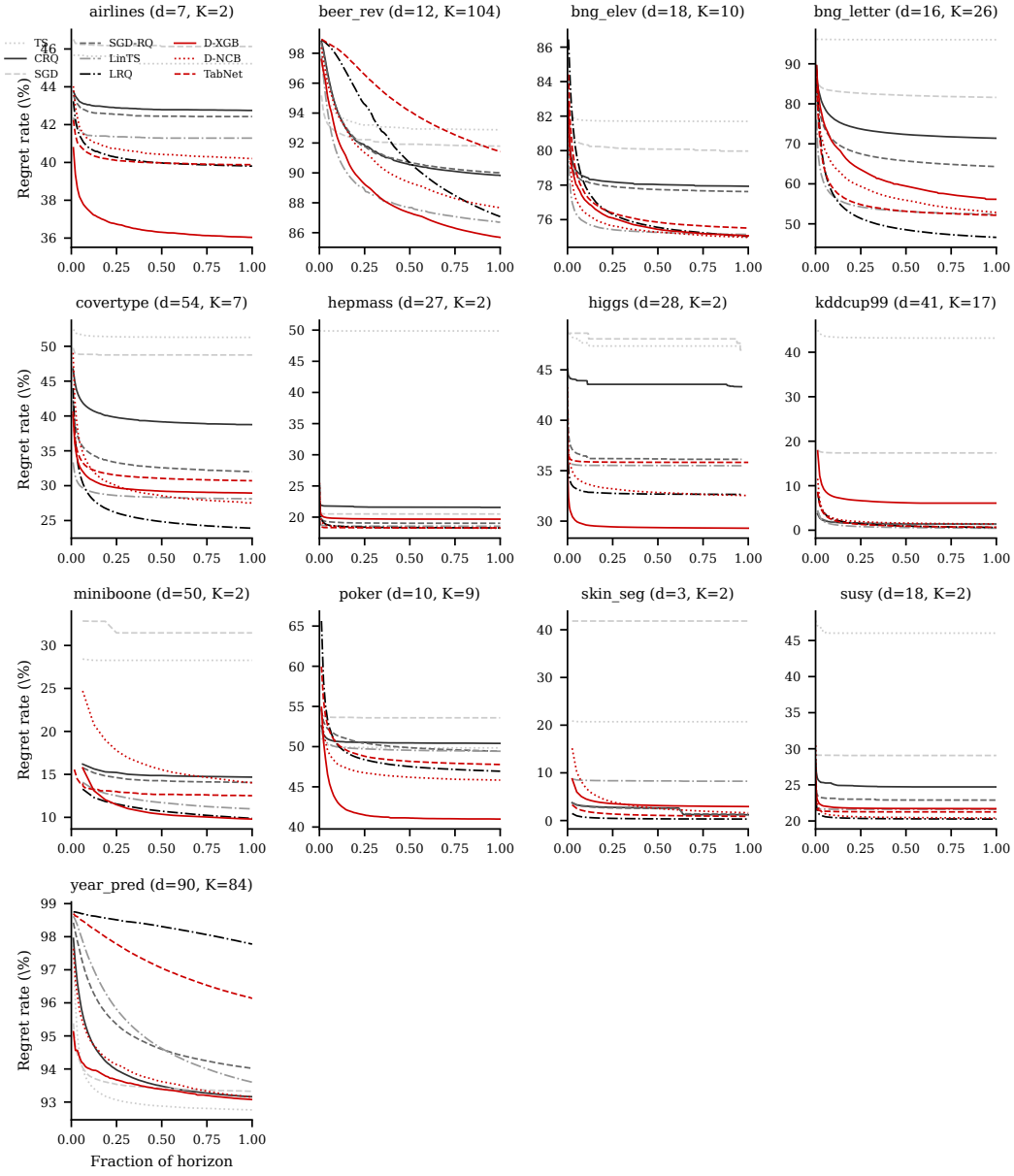


Figure 10: Regret rate vs. T for all 9 methods on each dataset. Shaded bands show ± 1 std across seeds.

J Regret Scaling Analysis

To empirically validate consistency with the regret bound (69), we fit two models to the TS-RQ cumulative regret curves across all datasets.

Power-law fit. We fit $R(T) = a \cdot T^{\alpha}$ via log-log regression. If regret were dominated by the $O(\sqrt{T \log T})$ exploration term, we would expect $\alpha \approx 0.5$. If dominated by the $O(\Delta_\ell T)$ bias term, we expect $\alpha \approx 1.0$. Figure 11 (left) shows that all 13 datasets yield $\alpha \in [0.87, 1.00]$, indicating that regret is overwhelmingly dominated by the linear bias term at the horizons in our experiments (T up to 11M).

Decomposition fit. We also fit $R(T) = a\sqrt{T} + c \cdot T$ directly, separating exploration and bias contributions. The bias coefficient c estimates the asymptotic regret rate $R(T)/T \to O(\Delta_\ell(\mathcal{M}))$. Figure 11 (right) shows the regret rate converging to a constant on selected datasets, confirming the linear bias term dominates.

Interpretation. This is exactly what Theorem 3.1 predicts: the exploration term $O(b\ell p\sqrt{KT\log(T/\delta)})$ grows as $\sqrt{T\log T}$ and becomes negligible relative to $O(\Delta_\ell T)$ for large T. The practical implication is that reducing Δ_ℓ (via deeper codebooks or richer base learners) is more impactful than reducing exploration cost—consistent with TS-RQ's strong empirical performance despite its greedy approximation of the joint model.

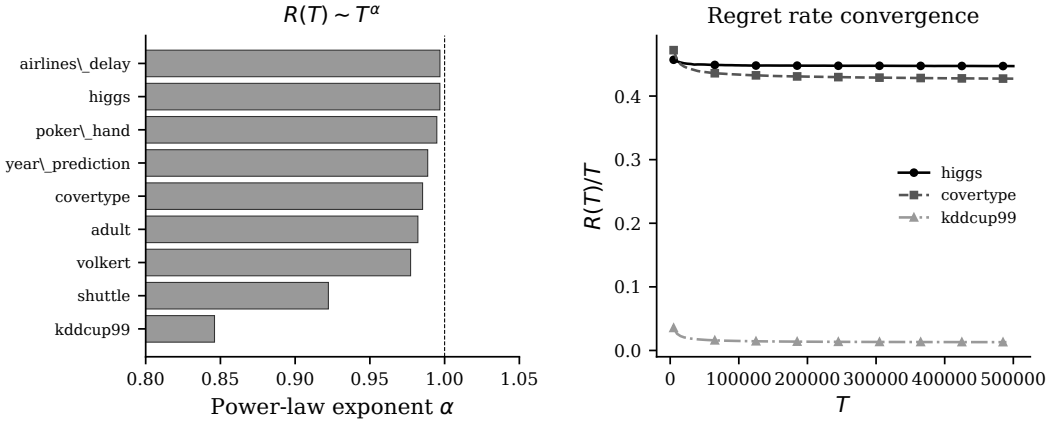


Figure 11: Regret scaling analysis for TS-RQ. **Left:** Power-law exponent α per dataset ($R(T) \sim T^\alpha$); all values are near 1.0, indicating bias-dominated regret. **Right:** Regret rate $R(T)/T$ vs. T for selected datasets, converging to a constant consistent with the $O(\Delta_\ell T)$ bias term in Theorem 3.1.

K Data Sufficiency: Synthetic Validation

We validate the data sufficiency condition $N/(bKd^2)$ on synthetic data with two feature distributions:

Gaussian features. N=500K i.i.d. standard normal in $\mathbb{R}^d$. Random per-centroid reward structure. LinTS-RQ wins when $N/(bKd^2) > 100$, ties at $30-100$, loses below 30. This is the pessimistic bound: structureless features provide no exploitable correlation for the codebook.

Structured features. Correlated continuous groups, binary categoricals, planted decision tree for class assignment. LinTS-RQ wins at *all* tested ratios (≥ 14), because the planted structure aligns with the RQ codebook's partition.

Real datasets behave like the structured synthetic: 12 of 13 datasets win at ratios as low as 1.1. The sole failure (year_prediction, ratio 0.04) is $25\times$ below any tested threshold.

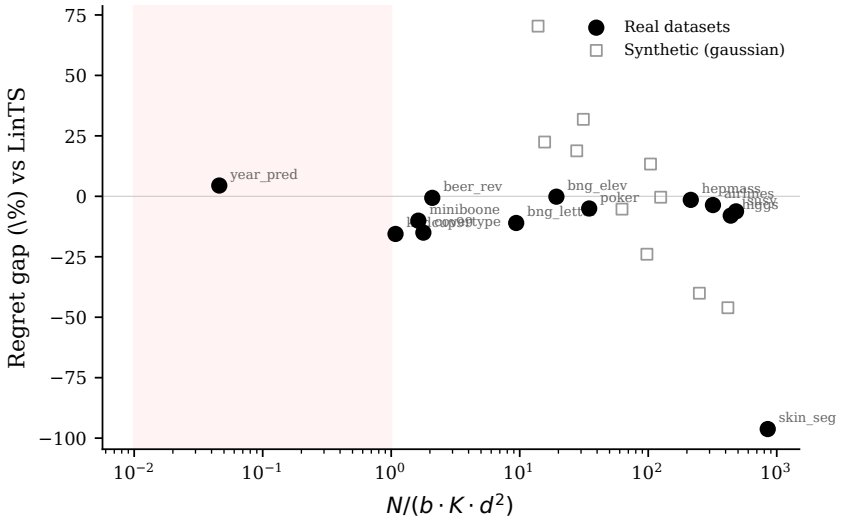


Figure 12: Data sufficiency: $N/(bKd^2)$ vs. LinTS-RQ regret gap. Real datasets (filled circles) win at ratios as low as 1.1. Synthetic Gaussian (open squares) requires ratio > 30. year_prediction is the sole outlier at ratio 0.04.

L year_prediction Depth Analysis

On year_prediction (d=90, K=84), LinTS-RQ degrades even at depth 1 (96.3% vs. LinTS baseline 93.4%). Adding depth makes it progressively worse (96.9% at ℓ=2, 97.4% at ℓ=4).

TS-RQ is unaffected: 93.0% at all depths, matching the baseline. SGD-LinTS-RQ improves slightly with depth ($96.9\% \rightarrow 95.0\%$) because diagonal estimation is more data-efficient than full covariance in $\mathbb{R}^{90}$.

The root cause is data fragmentation: with b=16 centroids, each centroid receives $\approx$ 28K samples to estimate $d^2K = 680$K parameters—a ratio of 0.04. Context-free TS achieves 92.7%, outperforming all feature-based methods, confirming that the features carry negligible signal for the 84-way quantile-binned target.

M Reconstruction Error

The normalized reconstruction error $\|x - \hat{x}\|^2 / \|x\|^2$ at depth 4 ranges from 0.005 (skin_segmentation) to 0.648 (year_prediction). year_prediction's high error confirms that b=16 centroids in $\mathbb{R}^{90}$ cannot capture the feature geometry, consistent with its status as the sole LinTS-RQ failure.

Table 10: Normalized reconstruction error at depth 4.

Dataset	NMSE (d=4)	LinTS-RQ wins?
skin_seg.	0.005	Yes (-96%)
kddcup99	0.018	Yes (-16%)
airlines	0.078	Yes (-3.6%)
miniboone	0.111	Yes (-10%)
covertype	0.185	Yes (-15%)
beer_rev.	0.186	Yes (-2.9%)
susy	0.187	Yes (-6.1%)
poker	0.196	Yes (-5.0%)
bng_elev.	0.211	Yes (-0.2%)
bng_letter	0.293	Yes (-11%)
hepmass	0.400	Yes (-1.1%)
higgs	0.431	Yes (-8.0%)
year_pred.	**0.648**	**No** ($+4.5\%$)

N Effective Rank and SGD-LinTS Failure

On datasets with effective rank < 1.5—skin_segmentation (eff. rank 1.3, d=3) and miniboone (eff. rank 1.1, d=50)—SGD-LinTS performs *worse than context-free TS* (42.0% vs 20.8% and 32.9% vs 28.3% regret rate). The diagonal approximation treats each feature coordinate independently; when most variance lies on a single axis, it wastes $d-1$ coordinates on noise. TS-RQ is unaffected: 2.6% on skin_segmentation and 14.7% on miniboone.

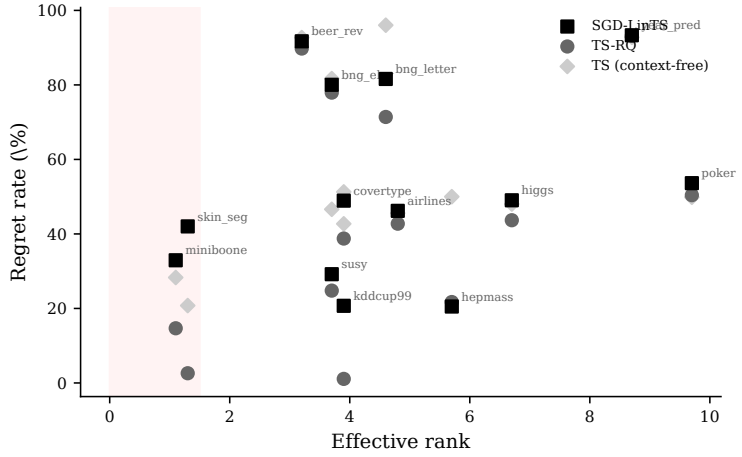


Figure 13: Effective rank vs. regret rate. SGD-LinTS (squares) fails severely when eff. rank < 1.5 (red zone), performing worse than context-free TS (diamonds). TS-RQ (circles) is unaffected.

O XGBoost and NeuralCB Comparison

LinTS-RQ achieves regret competitive with both nonlinear replay baselines. Against Doubling-XGB: 7 wins, 4 losses, 2 ties (30 LinTS-RQ seeds, 10 XGB seeds, 95% CIs; Table 11). Against Doubling-NeuralCB: 5 wins, 2 losses, 5 ties (30 LinTS-RQ seeds, 10 NeuralCB seeds; Table 12). Wins tend to be larger in magnitude than losses for both comparisons. The key differentiator is memory: both baselines require an $O(T)$ replay buffer while LinTS-RQ uses fixed $O(1)$ memory (Table 2).

Table 11: Per-dataset regret rate (%): LinTS-RQ vs. Doubling-XGB. Negative gap = LinTS-RQ has lower regret.

Dataset	LinTS-RQ	D-XGB	Gap	Call
kddcup99	0.2	9.0	−8.8	Win
bng_letter	46.6	56.1	−9.5	Win
covertype	23.9	28.9	−5.0	Win
beer_reviews	83.1	85.7	−2.6	Win
skin_seg	0.3	2.9	−2.6	Win
susy	20.3	21.7	−1.4	Win
hepmass	18.4	19.7	−1.3	Win
bng_elevators	75.0	75.1	0.0	Tie
miniboone	9.9	9.8	+0.1	Tie
higgs	32.7	29.3	+3.4	Loss
airlines_delay	39.8	36.0	+3.8	Loss
year_prediction	97.8	93.1	+4.7	Loss
poker_hand	46.9	41.0	+6.0	Loss

Table 12: Per-dataset regret rate (%): LinTS-RQ vs. Doubling-NeuralCB (30 LinTS-RQ seeds, 10 NeuralCB seeds). Negative gap = LinTS-RQ has lower regret.

Dataset	LinTS-RQ	D-NeuralCB	Gap	Call
beer_reviews	83.0	87.7	−4.6	Win
bng_letter	46.6	52.8	−6.2	Win
covertype	23.9	27.6	−3.7	Win
kddcup99	0.2	1.4	−1.2	Win
skin_seg	0.3	1.6	−1.3	Win
airlines_delay	39.8	40.2	−0.4	Tie
bng_elevators	75.0	75.0	+0.1	Tie
hepmass	18.4	18.2	+0.1	Tie
higgs	32.6	32.4	+0.3	Tie
susy	20.3	20.4	−0.1	Tie
poker_hand	46.9	45.8	+1.1	Loss
year_prediction	97.8	93.1	+4.6	Loss

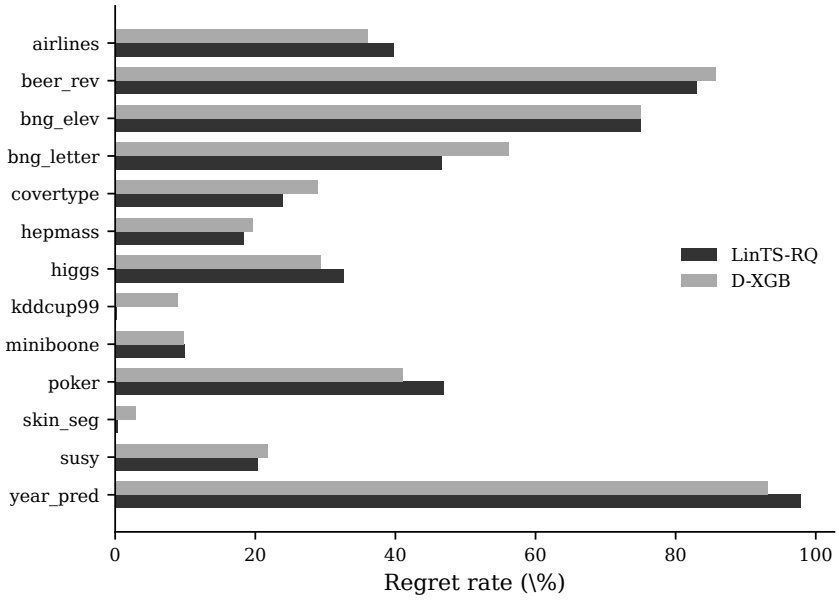


Figure 14: Per-dataset regret rate: LinTS-RQ vs. Doubling-XGB.

P TabNet-LinTS Comparison

LinTS-RQ wins on 11 of 13 datasets against TabNet-LinTS (11W/2L; losses: hepmass $+0.5\%$, year_prediction $+1.7\%$; Welch's t-test, 30 seeds each, 95% CIs). Both use the same 50K holdout (unsupervised); the RQ codebook matches or exceeds TabNet's reconstruction-based embedding.

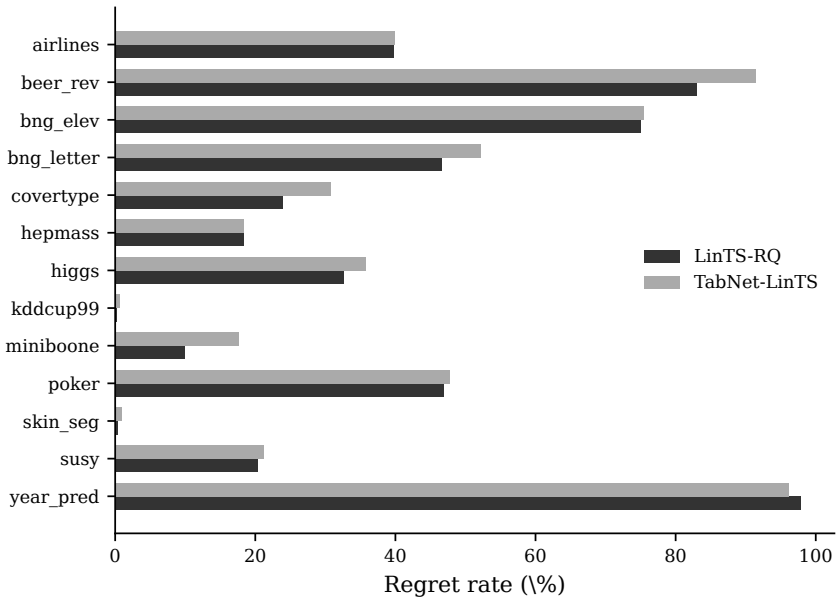


Figure 15: Per-dataset regret rate: LinTS-RQ vs. TabNet-LinTS.

Q Timing

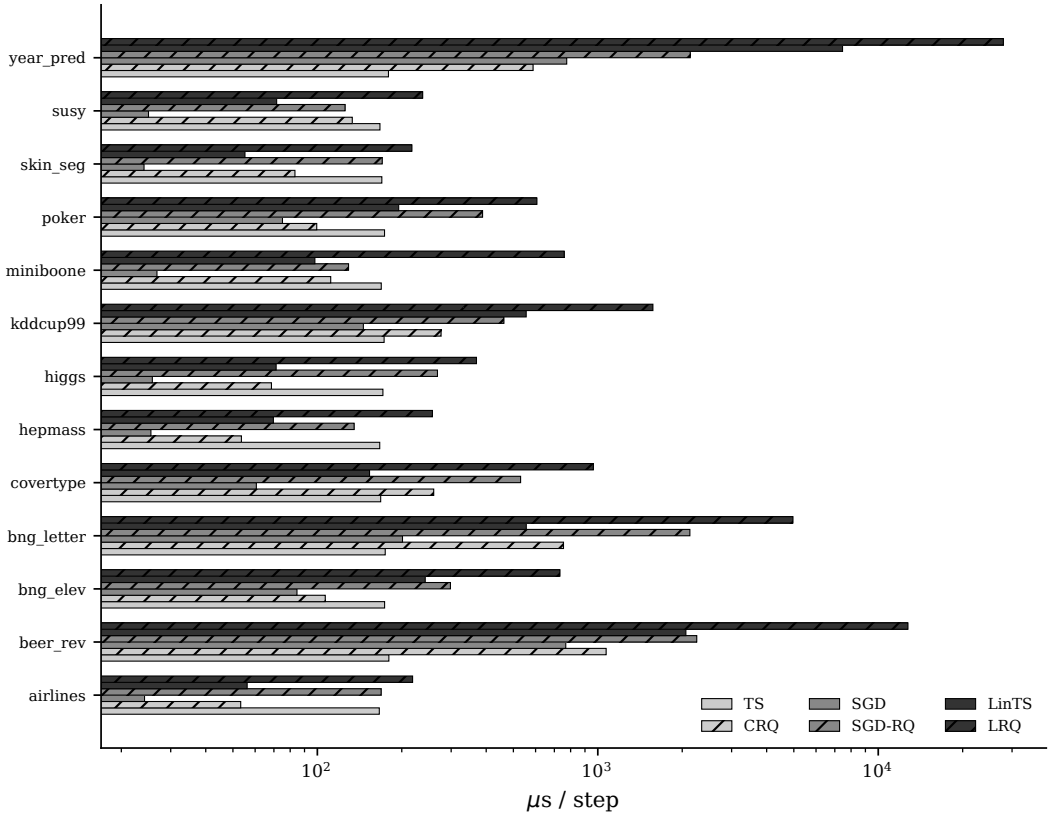


Figure 16: Wall-clock μs/step per method at adapted DRQm depths. TS-RQ is consistently the cheapest RQ variant ($50{-}750\,\mu$s). LinTS-RQ on year_prediction ($d{=}90$, $K{=}84$) reaches 28 ms.

R Memory Formulas

Model memory = (codebook + model parameters) $\times$ 8 bytes.

Component	**Parameters**	**Formula**
Codebook	ℓbd	Shared across arms
TS-RQ	$2\ell bK$	(sum, count) per cell
SGD-LinTS-RQ	$2\ell bdK$	(prec, f) vectors per cell
LinTS-RQ	$(d^2{+}d)\ell bK$	(V matrix + f vector) per cell
XGB replay	$T(d{+}2)$	Grows linearly with T

Adapted depth ℓ from shadow promotion varies by dataset (median across 30 seeds). TS-RQ goes deepest (up to $\ell{=}8$ on bng_letter); LinTS-RQ stays shallowest ($\ell{=}2{-}3$, limited by cascaded convergence).

Dataset	d	K	TS	TS-RQ	SGD-LinTS	SGD-RQ	LinTS	LinTS-RQ
airlines_delay	7	2	32B	4.1KB	224B	13.1KB	896B	29.8KB
beer_reviews	12	104	1.6KB	82.5KB	19.5KB	627.0KB	126.8KB	7.9MB
bng_elevators	18	10	160B	9.5KB	2.8KB	94.5KB	26.7KB	859.5KB
bng_letter	16	26	416B	68.0KB	6.5KB	742.0KB	55.2KB	5.2MB
covertype	54	7	112B	68.0KB	5.9KB	405.0KB	162.4KB	10.2MB
hepmass	27	2	32B	27.1KB	864B	50.6KB	11.8KB	384.8KB
higgs	28	2	32B	12.0KB	896B	105.0KB	12.7KB	619.5KB
kddcup99	41	17	272B	37.5KB	10.9KB	358.8KB	228.7KB	7.2MB
miniboone	50	2	32B	33.8KB	1.6KB	62.5KB	39.8KB	2.5MB
poker_hand	10	9	144B	7.0KB	1.4KB	95.0KB	7.7KB	250.0KB
skin_segmentation	3	2	32B	3.5KB	96B	5.6KB	192B	6.8KB
susy	18	2	32B	16.5KB	576B	22.5KB	5.3KB	263.2KB
year_prediction	90	84	1.3KB	64.5KB	118.1KB	3.7MB	5.2MB	252.0MB

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 claims RQ codebooks improve contextual bandits with fixed memory; this is supported by the asymptotic regret analysis (Theorem 3.1), shadow false-promotion control (Theorem 3.3), and experiments across 13 datasets (§4).

2. **Limitations**

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

   Answer: [Yes]

   Justification: §7 discusses the squared-loss approximation for binary rewards, the data sufficiency condition $N/(bKd^2) \gg 1$ (with year_prediction as a concrete failure case), and the dependence on a holdout set for codebook training.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: Theorem 3.1 (asymptotic regret, Appendix C) is conditional on three base-learner convergence properties, which are stated explicitly and verified for all three model classes (Appendix B). Theorem 3.3 (false-promotion control) is proved in Appendix D. Theorem 3.2 exactly decomposes depth-dependent approximation error, and Corollary 3.1 proves geometric decay under explicit reconstruction- and additive-tail contraction assumptions; their proofs are in Appendix C.

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: Algorithm 1 specifies the method. §4 describes all datasets, codebook parameters (b=16, initial depth ℓ_0=2, maximum depth $\ell_{\max}$=8), number of seeds (30), and baselines.

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: Code is available at `https://github.com/atavory/rqb`. All 13 datasets are publicly available from OpenML.

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: §2 specifies the clipping function (σ), and §4 specifies codebook size (b=16), Thompson sampling noise scale, and the holdout/bandit split. Appendix tables provide per-dataset details.

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: All experiments use 30 seeds. Figures show ± 1 standard deviation; tables report 95% CIs. The shadow promotion test uses Freedman's martingale inequality with Bonferroni correction at significance level α=0.05.

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: [No]

   Justification: Appendix Q reports wall-clock timings for CPU-only experiments, and Appendix R analyzes model memory. The CPU model and available RAM are not specified.

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 uses only publicly available datasets and proposes a general-purpose bandit algorithm with no foreseeable ethical concerns.

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: The introduction motivates efficient online decision-making with bounded memory. Section 7 discusses reward-signal biases, and Section 8 notes that standard fairness and privacy concerns for bandit algorithms apply.

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 pre-trained models, generative models, or scraped datasets. RQ-Bandit is a lightweight bandit algorithm.

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 are cited with their original publications in Appendix G. All are publicly available from OpenML and UCI under open licenses.

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 RQ-Bandit implementation is available at `https://github.com/atavory/rqb`.

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: No crowdsourcing or human subjects research was conducted.

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: No human subjects research was conducted.

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 were used only for writing assistance and editorial polishing. They are not a component of the RQ-Bandit method or experimental pipeline.