# Posterior-Predictive Pass-at-$k$

A Two-Parameter Replacement for the Independence Extrapolation in Fine-Tuning Diagnostics

Peter Cotton

September 2, 2026

**Abstract**

A prompt that fails a handful of scored attempts is rated impossible by the standard pass-at-$k$ extrapolation $1-(1-\hat{p})^k$, and rated so at every $k$. On nine thousand prompts with eight scored samples each, that plug-in step — the same one the coverage diagnostic of Malladi et al. (2026) applies as $f_{16}(p) = 1-(1-p)^{16}$ — costs seven-fold in held-out log loss and understates aggregate pass@8 by thirteen points. Replacing the plug-in evaluation with the posterior expectation $\mathbb{E}[1-(1-p)^k \mid s]$ under a two-parameter difficulty prior, fitted once across prompts by marginal maximum likelihood, removes most of the damage: log loss 0.48 against 3.37, the extrapolated curve within two points of truth. Probit-normal and beta-binomial priors tie to three decimals, so nothing hangs on the family. The bias shrinks as the per-prompt sample count grows, and the data cover one model on one task suite.

## 1 Introduction

Language models are improved after pretraining by having them attempt problems and scoring the attempts automatically: a unit test passes, a final answer matches, a verifier accepts. Training on one's own scored attempts is what reinforcement-style post-training is, and the evaluations that steer it sample each prompt many times, because a model that solves a problem once in sixteen tries can often be trained into solving it reliably. Pass@$k$ is the probability that at least one of $k$ sampled attempts succeeds, averaged over prompts, and it is the working currency of post-training evaluation. When $n \geq k$ scored samples per prompt are available, the unbiased estimator of Chen et al. (2021), $1 - \binom{n-c}{k}/\binom{n}{k}$ with $c$ the correct count, settles the matter. The trouble begins when $k$ exceeds the samples at hand and the curve must be extrapolated. The common device is the independence formula: estimate a per-prompt success rate $\hat{p}$, report $1-(1-\hat{p})^k$. It appears inside diagnostics that steer training decisions. Malladi et al. (2026), whose filtered fine-tuning method concentrates learning on the prompts a model has not yet mastered, decide where the method should be applied by comparing coverage — the set of prompts solvable within sixteen attempts — before and after standard fine-tuning, and write "we convert pass@1 to an estimated pass@16 value using $f_{16}(p) := 1-(1-p)^{16}$" as a step of that rule. The rule works well in their experiments.

This note is about the extrapolation step, not the diagnostic built on it. The formula is exact when $p$ is known; applied to a noisy $\hat{p}$ it inherits a specific pathology. A prompt with zero successes among $m$ training samples gets $\hat{p} = 0$ and a predicted pass@$k$ of zero at every $k$, a confident impossibility claim that a latent success rate of, say, 0.1 will refute given enough

samples. We measure the cost of the plug-in on released rollout records and show that the two-parameter empirical-Bayes repair, a posterior expectation in place of a plug-in evaluation, removes most of it.

## 2 The estimation problem

Each prompt $i$ carries a latent per-attempt success probability $p_i$, drawn independently from a difficulty distribution $G$ on $[0,1]$; conditional on $p_i$, scored attempts at that prompt are independent Bernoulli($p_i$) trials. From $m$ training attempts the observed success count is $s_i \sim \mathrm{Bin}(m, p_i)$. Two estimands are of interest. Per prompt, the probability that a fresh batch of $k$ attempts contains a success,

$$q_k(p_i) \;=\; 1-(1-p_i)^k, \tag{1}$$

which must be predicted from $s_i$ alone; and in aggregate, the pass@$k$ the whole prompt population would exhibit,

$$\pi_k \;=\; \int_0^1 \big(1-(1-p)^k\big)\, dG(p). \tag{2}$$

The conditional independence of attempts given $p_i$ is not the assumption under attack: attempts are drawn with independent seeds, and the model grants that. What varies across the field is $p_i$ itself, and everything below is about respecting that variation when $s_i$ is a noisy glimpse of it.

## 3 The plug-in and its failure mode

The plug-in predictor evaluates $q_k$ at the point estimate $\hat{p}_i = s_i/m$:

$$\widehat{\mathrm{pass}}_k^{\,\mathrm{plug}} \;=\; 1-\Big(1-\tfrac{s_i}{m}\Big)^k. \tag{3}$$

At $s_i = 0$ the prediction is exactly zero at every $k$, however small $m$ is, so each eventual success on such a prompt costs unbounded log loss. The aggregate defect is one inequality.

*Proposition 1 (The plug-in is biased downward for every difficulty). Fix $m \geq 1$ and $k \geq 2$, and let $s \sim \mathrm{Bin}(m,p)$. Then*

$$\mathbb{E}\Big[\,1-\big(1-\tfrac{s}{m}\big)^k\Big] \;\leq\; 1-(1-p)^k, \tag{4}$$

*with equality only at $p \in \{0,1\}$; at $k=1$ equality holds for all $p$. Consequently the aggregate plug-in curve lies below $\pi_k$ for every $G$ that is not concentrated on $\{0,1\}$.*

*Proof.* Write $\varphi(x) = (1-x)^k$ on $[0,1]$. Its second derivative is $\varphi''(x) = k(k-1)(1-x)^{k-2} \geq 0$, strictly positive on $[0,1)$ when $k \geq 2$, so $\varphi$ is convex, and strictly convex there. The estimate $\hat{p} = s/m$ satisfies $\mathbb{E}\,\hat{p} = p$. Jensen's inequality gives $\mathbb{E}\,\varphi(\hat{p}) \geq \varphi(\mathbb{E}\,\hat{p}) = (1-p)^k$, and the inequality is strict whenever $\hat{p}$ is nondegenerate, which is exactly $p \in (0,1)$. Negating and adding one reverses the inequality into the display. At $k=1$, $\varphi$ is affine and Jensen holds with equality. The aggregate statement follows by integrating the pointwise inequality against $G$. □

The bias grows with the curvature of $\varphi$, hence with $k$, and shrinks with the variance of $\hat{p}$, hence with $m$: extrapolating further from fewer samples costs more, which is the regime diagnostics live in.

## 4 The posterior predictive

Give $p$ a prior $\pi_\theta$ with two parameters fitted by marginal maximum likelihood across prompts, and predict with the posterior expectation

$$\widehat{\text{pass}}_k^{\text{post}} \;=\; \mathbb{E}\big[\,1-(1-p)^k \;\big|\; s\,\big]. \tag{5}$$

Two convenient families serve. Under a Beta$(a,b)$ prior the posterior is Beta$(a+s,\; b+m-s)$ and the moment is a falling product,

$$\mathbb{E}\big[(1-p)^k \mid s\big] = \prod_{j=0}^{k-1} \frac{b+m-s+j}{a+b+m+j}, \tag{6}$$

so Equation (5) is closed-form. Under a probit-normal prior, $p=\Phi(\theta)$ with $\theta\sim N(\mu,\tau^2)$, the marginal likelihood of the observed counts and the posterior moment are one-dimensional Gauss–Hermite quadratures over $\theta$; this is the family that composes with correlated extensions, since a factor-correlated portfolio of successes conditions to exactly this form (Cotton, 2026). In the exchangeable setting of this note the two families are competitors on equal footing, and the study below treats them as such.

*Proposition 2 (The posterior predictive is the target). Let $Y$ indicate a success among $k$ fresh attempts at the same prompt, drawn independently of the training attempts given $p$. Under the model, with prior equal to the true difficulty law $G$: (i) $\mathbb{E}[Y \mid s] = \mathbb{E}[\,1-(1-p)^k \mid s\,]$, so the posterior predictive is the exact conditional success probability; (ii) it minimizes expected Brier score and expected log loss among all predictors that depend on the data only through $s$; (iii) its average over prompts is exactly $\pi_k$.*

*Proof.* Given $p$, $Y$ is Bernoulli with parameter $q_k(p)=1-(1-p)^k$ and is independent of $s$. The tower property gives $\mathbb{E}[Y \mid s] = \mathbb{E}\big[\mathbb{E}[Y \mid p,s] \;\big|\; s\big] = \mathbb{E}[q_k(p) \mid s]$, which is (i). For (ii), the conditional expectation minimizes mean squared error among $s$-measurable predictors, which is the Brier statement, and the true conditional probability minimizes expected log loss because the excess log loss of any other predictor is the expectation of a Kullback–Leibler divergence, which is nonnegative. For (iii), $\mathbb{E}\big[\mathbb{E}[q_k(p) \mid s]\big] = \mathbb{E}\, q_k(p) = \pi_k$, the tower property again. □

Proposition 2 holds at the true $G$; in practice $G$ is replaced by a two-parameter family fitted by marginal maximum likelihood, and the study below measures what survives that substitution.

## 5 Numerical study on released rollouts

The data are the publicly released Pass8 rollout records of the RLVE training suite: procedurally generated reasoning tasks, each with an automatic verifier, of the kind used as reinforcement-learning environments for language models. They carry nine thousand prompts across eighteen environment families, eight scored samples per prompt from the Qwen3-4B-Thinking model, three gigabytes of response text reduced to a forty-one-kilobyte outcome table.[1] Samples 0–3 of each prompt are the training half; samples 4–7 are held out. The held-out per-prompt pass@4 is then a single Bernoulli outcome, whether any of the four succeeded, so predicted probabilities are scored

[1] Dataset: `CL-From-Nothing/RLVE-Qwen3-4B-Thinking-2507-Pass8-Rollouts` on Hugging Face. Extraction and experiment scripts are committed under `research/cavity_calculus/exp2_passk/` in the `winning` repository.

by log loss and Brier score with no estimator ambiguity. The fitted priors are Beta(0.40, 0.83) and probit-normal $\mu = -0.78$, $\tau = 1.39$; both put substantial mass near $p = 0$ and $p = 1$, which is what the environments look like.

| predictor of held-out pass@4 | log loss | Brier |
|---|---|---|
| plug-in $1 - (1 - s/4)^4$ | 3.367 | 0.192 |
| posterior predictive, probit-normal | 0.483 | 0.160 |
| posterior predictive, beta-binomial | 0.483 | 0.160 |

Table 1: Per-prompt scores over nine thousand held-out Bernoulli outcomes. The plug-in's log loss is dominated by zero-training- success prompts that succeed in the held-out half; the posterior predictive prices those prompts at their shrunken posterior rate instead of zero. The two prior families tie.

| aggregate | $k = 1$ | $k = 2$ | $k = 4$ | $k = 8$ |
|---|---|---|---|---|
| truth (all eight samples) | 0.322 | 0.442 | 0.569 | 0.682 |
| plug-in from the training half | 0.322 | 0.412 | 0.495 | 0.549 |
| probit-normal posterior | 0.323 | 0.444 | 0.561 | 0.663 |
| beta-binomial posterior | 0.323 | 0.444 | 0.560 | 0.661 |

Table 2: Aggregate pass@$k$ extrapolated from four samples per prompt, against the unbiased estimator of Chen et al. (2021) on all eight. The plug-in's downward bias grows with $k$ as Proposition 1 requires, reaching thirteen points absolute at $k = 8$; the posterior predictives land within two points.

The per-prompt comparison in Table 1 is seven-fold in log loss and twenty percent in Brier, for two fitted parameters. The aggregate comparison in Table 2 shows the plug-in bias growing with the extrapolation distance exactly as the convexity argument says it must.

## 6 Scope

The plug-in bias scales with the noise in $\hat{p}$, hence shrinks as the per-prompt sample count grows; the study uses four training samples per prompt, and Malladi et al. (2026) do not state the count behind their pass@1 estimates, so the numbers here bound the mechanism, not the magnitude in their setting. The data cover one model on one environment suite; transfer to math and code benchmarks is plausible and unshown. The two prior families are statistically indistinguishable here, so the choice between them can be made on convenience or, when correlated portfolios of attempts arise, on composability (Cotton, 2026). Diagnostics built on the plug-in, including the coverage ratio of Malladi et al. (2026), accept the posterior predictive as a drop-in for their extrapolation step, and measuring that substitution in their setting is the natural next experiment.

## References

M. Chen et al. Evaluating large language models trained on code. arXiv:2107.03374, 2021.

P. Cotton. Scalable inversion of contests with correlated performances, including softmax and multinomial probit. arXiv:2609.01133, 2026.

S. Malladi, S. Jelassi, D. Foster, J. T. Ash, and A. Krishnamurthy. TailSFT: filtered fine-tuning improves post-training performance. arXiv:2608.25756, 2026.