# Industrializing Prediction-Powered Inference: The GLIDE Library for Reliable GenAI and Agentic Systems Evaluation

**Grégoire Martinon** [1] **Ibrahim Merad** [1] **Mohammed Raki** [1]

## Abstract

Reliable evaluation of agentic systems requires unbiased estimates with valid uncertainty, but standard practice navigates between costly human annotation and biased LLM-as-judge proxies. Prediction-powered inference (PPI) combines both into debiased estimates with valid confidence intervals, yet its various methods remain scattered across papers under partial implementations. We introduce GLIDE, an open-source Python library that unifies state-of-the-art PPI estimators (PPI++, Stratified PPI, Predict-Then-Debias and its stratified variants, Active Statistical Inference) and samplers (uniform, stratified, active, cost-optimal) under a scipy-style API specialized to mean estimation. GLIDE ships with a reproducible Monte Carlo validation suite, an empirically grounded decision tree for method selection, and an agentic evaluation case study showing substantial annotation savings at equivalent precision. The GLIDE package is available at this URL: https://github.com/EmertonData/glide

## 1. Introduction

Consider a clinical decision-support agent that retrieves guidelines, reasons over a patient record, and proposes a treatment. Measuring whether such an agent behaves correctly requires expert clinician review of multi-step trajectories, a process where today's two dominant practices both fail at scale. Full human annotation is reliable but slow and costly; LLM-as-judge is fast and cheap but biased, especially for knowledge-intensive, domain-specific tasks.

Rather than choosing between human reliability and proxy scale, **prediction-powered inference** (PPI) (Angelopoulos et al., 2023a;b) combines them to obtain the strengths of both: a small set of human annotations with a large set of cheap proxy predictions to produce unbiased estimates with valid confidence intervals. Coverage holds unconditionally on the proxy: a worse proxy yields wider intervals, not invalid ones.

The PPI literature has matured rapidly, with extensions for power tuning (Angelopoulos et al., 2023b), stratified populations (Fisch et al., 2024; Fogliato et al., 2024), small-sample bootstrap procedures (Kluger et al., 2025), active sampling (Zrnic & Candès, 2024; Gligorić et al., 2025), cost-optimal budgets (Angelopoulos et al., 2025), and recalibration (Ji et al., 2025). Yet, as already noted in (Song et al., 2026), this progress is scattered across papers with heterogeneous notation and partial reference implementations. The `ppi_py` library (Angelopoulos et al., 2023a) established the foundation but predates several of these methods and emphasizes generality across estimands (GLM, M-estimators) over depth in mean estimation, where most use-cases live.

Agentic evaluation is a natural fit for PPI on several counts. The cost asymmetry between proxy and ground truth is extreme. Agentic pipelines also exhibit natural heterogeneity: performance and proxy quality vary substantially across the tools invoked, the types of queries handled, and the sub-agents involved, motivating stratified estimation. Modern judges increasingly produce calibrated per-sample uncertainty, unlocking active sampling. Existing libraries cover parts of this landscape but no single interface unifies the matching family of estimators and samplers, leaving practitioners to stitch methods together across academic repositories.

We close this gap with **GLIDE** (Generated Label Inference and Debiasing Engine), an open-source Python library focused on mean estimation. Our contributions are:

1. **A unified library**, exposed under a scipy-style API, and integrating several PPI estimators: PPI++ (Angelopoulos et al., 2023b), Stratified PPI++ (Fisch et al., 2024), Predict-Then-Debias (Kluger et al., 2025) with stratified variants, Active Statistical Inference (Zrnic & Candès, 2024); and different samplers: uniform, stratified (Fogliato et al., 2024), active (Zrnic & Candès, 2024; Gligorić et al., 2025), cost-optimal (Angelopou-

[1]Emerton Data, 16 Avenue Hoche, 75008, Paris, France. Correspondence to: Grégoire Martinon <gregoire.martinon@emerton-data.com>.

*Accepted to the ICML 2026 Workshop on Statistical Frameworks for Uncertainty in Agentic Systems*, Seoul, South Korea, 2026.
Copyright 2026 by the author(s).

los et al., 2025), alongside classical baselines.

2. **A reproducible validation suite**: Monte Carlo coverage tests, sensitivity analyses to proxy quality, and effective-sample-size benchmarks across all estimators.
3. **An agentic-evaluation case study** showing that GLIDE attains equivalent precision to classical sampling with substantially fewer human annotations.

## 2. Background: Prediction-Powered Inference

**Setup.** Let $(X, Y) \sim P$ where $X$ is the input or behavior to be evaluated (in our case, a conversation, an LLM output, or an agentic trajectory) and $Y \in \mathbb{R}$ is the corresponding human-assigned label (e.g., a binary correctness indicator, a toxicity indicator, or a tool-use rating). The target estimand is the population mean $\theta^\star = \mathbb{E}[Y]$, which encodes a performance or risk metric of interest. We have access to two datasets drawn i.i.d. from $P$: a small *labeled* set $\mathcal{L} = \{(X_i, Y_i, f(X_i))\}_{i=1}^n$ and a large *unlabeled* set $\mathcal{U} = \{(X_j, f(X_j))\}_{j=1}^N$ with $N \gg n$, where $f$ is a fixed proxy (typically an LLM-as-judge) producing predictions of $Y$. We assume $f$ is trained on data disjoint from $\mathcal{L} \cup \mathcal{U}$, so its predictions are i.i.d. across both sets.

**The PPI estimator.** The classical PPI mean estimator (Angelopoulos et al., 2023a) debiases the proxy mean using the labeled set:

$$\hat{\theta}^{\mathrm{PPI}} = \underbrace{\frac{1}{N}\sum_{j=1}^{N} f(X_j)}_{\text{proxy mean on } \mathcal{U}} + \underbrace{\frac{1}{n}\sum_{i=1}^{n} (Y_i - f(X_i))}_{\text{rectifier on } \mathcal{L}}. \tag{1}$$

The first term is the cheap proxy mean; the second is a correction for the proxy's bias. PPI++ (Angelopoulos et al., 2023b) introduces a tuning parameter $\lambda \in \mathbb{R}$:

$$\hat{\theta}_\lambda^{\mathrm{PPI++}} = \frac{1}{n}\sum_{i=1}^{n} Y_i + \lambda \left( \frac{1}{N}\sum_{j=1}^{N} f(X_j) - \frac{1}{n}\sum_{i=1}^{n} f(X_i) \right), \tag{2}$$

which recovers (1) for $\lambda = 1$ and the classical labeled mean for $\lambda = 0$. The optimal $\lambda^\star$ is closed-form and minimizes the asymptotic variance, guaranteeing that PPI++ is never worse than the classical estimator in the limit, even when the proxy is uninformative or adversarial.

**Confidence intervals.** Thanks to the Central Limit Theorem, $\hat{\theta}_\lambda^{\mathrm{PPI++}}$ is asymptotically normal, and a $(1-\alpha)$ confidence interval is obtained from the plug-in variance estimator and the normal quantile $z_{1-\alpha/2}$. For small $n$, the central-limit approximation degrades; bootstrap-based constructions provide a finite-sample alternative (Kluger et al., 2025).

**Extensions.** The same template (combine a labeled correction with a proxy summary) generalizes to several settings relevant to evaluation: stratified populations, where power tuning is performed per stratum (Fisch et al., 2024); non-uniform sampling driven by per-sample proxy uncertainty, known as Active Statistical Inference (ASI) (Zrnic & Candès, 2024; Gligorić et al., 2025); small-sample regimes addressed by Predict-Then-Debias (PTD), which provides bootstrap-based confidence intervals (Kluger et al., 2025); and cost-aware sampling probabilities derived from a known proxy/annotation cost ratio (Angelopoulos et al., 2025). See to (Song et al., 2026; Ji et al., 2025) for a recent review. GLIDE implements all of these extensions; we develop the corresponding estimators and samplers in Section 4.

**Effective sample size.** A natural figure of merit for any PPI estimator is its *effective sample size*, defined as the number of fully labeled samples a classical estimator would require to match the same confidence-interval width:

$$n_{\mathrm{eff}} = n \cdot \frac{\widehat{\mathrm{Var}}(\bar{Y}_n)}{\widehat{\mathrm{Var}}(\hat{\theta}_\lambda^{\mathrm{PPI++}})}. \tag{3}$$

The ratio $n_{\mathrm{eff}}/n \geq 1$ quantifies the practical return on investment of the proxy: it translates directly into human-review-hour or budget savings, and is the headline metric we report throughout Sections 5 and 6.

## 3. Why Agentic Evaluation Needs PPI

Section 2 fixed the abstract setting; we now argue that agentic systems exhibit four properties, simultaneously and to an unusual degree, under which PPI is most valuable. Each property maps directly to a method in the recent literature, and to a corresponding GLIDE component.

**1. Extreme cost asymmetry.** Evaluating an agentic trajectory is qualitatively more expensive than evaluating a single LLM output. A trajectory may involve retrieval, tool calls, intermediate reasoning steps, and a final response, all of which a domain expert must inspect to certify correctness. Annotation costs of several dollars per trajectory are common in healthcare, legal, and financial settings, while an LLM-as-judge call costs cents. This two-to-three-orders-of-magnitude gap is precisely the regime where cost-aware sampling (Angelopoulos et al., 2025) is relevant, setting the labeling rate as a function of the proxy/annotation cost ratio, and the available signal.

**2. Natural stratification.** Agentic pipelines partition observations along several axes that are typically known at evaluation time: the tools invoked, the type of query handled, the sub-agents involved, and the retrieval source consulted. Proxy quality varies across these strata in ways that

matter: an LLM-as-judge that reliably evaluates arithmetic reasoning may struggle with code-execution traces or with domain-specific clinical guidelines. Stratified PPI (Fisch et al., 2024; Fogliato et al., 2024) runs PPI within each stratum and combines the results with per-stratum weights, isolating proxy-quality differences and yielding tighter intervals than the unstratified estimator.

**3. Available proxy uncertainty.** Modern judges can produce per-sample uncertainty signals: verbalized confidence, log-probabilities, multiple generation consistency, or ensembles of judges. These signals identify the trajectories on which the proxy is least reliable, and therefore the trajectories where human annotation is most informative. Active Statistical Inference (Zrnic & Candès, 2024; Gligorić et al., 2025) exploits this by sampling labeling probabilities proportional to proxy uncertainty, concentrating expensive expert effort where it carries the most statistical value.

**4. Consequential deployment regimes.** Criticial systems (e.g., clinical decision support, financial advisory, regulated automation, safety-critical autonomy) are also those where biased point estimates without valid uncertainty have direct consequences. Distribution-free coverage guarantees are a deployment requirement in these settings. PPI's unconditional-on-the-proxy validity makes it well suited to agentic pipelines that must be auditable.

## 4. The GLIDE Framework

### 4.1. Three-step decomposition

GLIDE organizes evaluation around three sequential steps inherited from classical survey theory: *sampling*, *annotation*, and *estimation*. A small budget of observations (or trajectories) is selected from a large pool of proxy-labeled ones (sampling); domain experts produce ground-truth labels on the selected subset (annotation); the labeled subset and the proxy predictions are combined into a debiased point estimate with a confidence interval (estimation). Sampling and estimation are statistical and algorithmic; while annotation is irreducibly task-specific and out of scope for GLIDE. The library requires only that experts return labels in a machine-readable form (categories or numbers), whose business meaning is context-dependent but whose statistical treatment is not.

This factorization is more than a presentation choice. It allows samplers and estimators to be developed, tested, and contributed independently: a contributor proposing a new sampler need not touch any estimator, and conversely. It also lets practitioners mix and match across the families of methods recalled in Section 2, choosing the sampler that matches their evaluation setup and the estimator that matches their statistical regime.

### 4.2. Samplers

GLIDE implements four samplers covering the regimes identified in Section 3.

**UniformSampler** draws $n$ observations uniformly without replacement from the pool. It is the appropriate baseline when no auxiliary signal is available.

**StratifiedSampler** allocates $n$ observations across user-defined strata. Two allocation rules are supported: *proportional* ($n_h \propto N_h$), and *Neyman* ($n_h \propto N_h \sigma_h$), where $\sigma_h$ is the stratum standard deviation of the proxy labels (Fogliato et al., 2024). Both allocations produce non-integer values that GLIDE rounds via Hamilton's largest-remainder method to ensure $\sum_h n_h = n$ exactly.

**ActiveSampler** draws each observation independently with a Bernoulli probability proportional to a user-supplied uncertainty score on the proxy . The proportionality constant is calibrated so that the expected sample size equals the budget (Zrnic & Candès, 2024; Gligorić et al., 2025). Because the draws are independent Bernoullis, the realized sample size matches the budget only in expectation; this is the standard cost of admitting per-sample probabilities.

**CostOptimalSampler** comes in two variants. The constant-rate variant (`CostOptimalRandomSampler`) computes a single sampling probability from a small burn-in dataset and known proxy/annotation costs, minimizing total cost (Angelopoulos et al., 2025). The feature-dependent variant (`CostOptimalSampler`) additionally consumes a per-sample uncertainty score, computing observation-level optimal probabilities $\pi(x)$. The feature-dependent variant strictly dominates the constant-rate one when the uncertainty score is informative, but requires an extra signal that is not always available.

### 4.3. Estimators

GLIDE implements several mean estimators alongside three classical baselines (Table 1). All estimators return a `PredictionPoweredMeanInferenceResult` carrying the point estimate, the confidence interval at user-specified level $1 - \alpha$, the effective sample size $n_{\text{eff}}$ (eq. 3), and the metric label.

The classical baselines do not consume proxy labels and serve two purposes: they recover the no-proxy regime as a sanity check, and they provide the reference variance against which $n_{\text{eff}}$ is computed.

### 4.4. A decision framework

The decision tree of Figure 1 routes the user through three stages. The sampling stage selects among samplers based on three signals: *are cost estimates of human and proxy*

*Table 1.* GLIDE estimators and their statistical regimes. CLT-based estimators rely on $n \gtrsim 50$; bootstrap-based estimators are valid for smaller samples.

| Estimator | Regime / reference |
|---|---|
| `PPIMeanEstimator` | PPI++ with power tuning by default (Angelopoulos et al., 2023b) |
| `StratifiedPPIMeanEstimator` | Per-stratum power tuning (Fisch et al., 2024) |
| `PTDMeanEstimator` | Bootstrap for small $n$ (Kluger et al., 2025) |
| `StratifiedPTDMeanEstimator` | Stratified extension of PTD (Kluger et al., 2025) |
| `ASIMeanEstimator` | IPW (Inverse Propensity Weighting) debiasing (Zrnic & Candès, 2024) |
| `ClassicalMeanEstimator` | Sample mean baseline (no proxy) |
| `StratifiedClassicalMeanEstimator` | Stratified baseline (no proxy) |
| `IPWClassicalMeanEstimator` | IPW baseline (no proxy) |

*annotations available?* (cost-optimal samplers), *are per-sample proxy uncertainties available?* (active sampling), and *is the dataset structured into strata with heterogeneous proxy quality?* (stratified sampling). The annotation stage is the one revealing ground truth labels by human expert review, and is done outside GLIDE. The estimation stage selects among the estimators using a single threshold on the number of human labels: *are at least 50 human labels available?* (CLT-based intervals if yes, bootstrap-based PTD variants otherwise). When stratified estimators are used, this threshold applies per stratum. The decision tree is empirically supported by the validation suite in Section 5.

### 4.5. API

GLIDE follows the conventions of the scientific Python ecosystem (scipy, scikit-learn). Estimators are stateful objects with an `estimate` method. Samplers expose a `sample` method returning a pair $(\pi, \xi)$: $\pi \in [0,1]^N$ holds the per-observation sampling probabilities, and $\xi \in \{0,1\}^N$ holds the binary inclusion indicators drawn as Bernoulli realizations of $\pi$. Results are dataclasses carrying the point estimate, the confidence interval at user-specified level $1-\alpha$, the effective sample size $n_{\text{eff}}$, and the metric label. A complete end-to-end workflow (sampling, annotation, estimation) fits in fewer than six lines of Python; we provide a minimal example in Appendix A. New samplers and estimators can be contributed as single-file modules implementing the `sample` or `estimate` interface, lowering the barrier for the research community to deposit reference implementations.

## 5. Empirical Validation

Each estimator implemented in GLIDE is accompanied by a Monte Carlo validation notebook that establishes three properties: (i) the empirical coverage of its confidence intervals matches the nominal level across a range of proxy quality and confidence levels, (ii) the interval is no wider than the classical labeled-only baseline at any correlation, and (iii) the effective sample size $n_{\text{eff}}$ grows with the proxy/ground-truth correlation. We describe the protocol and report representative results for the PTD estimator; other notebooks follow similar protocols and yielding consistent conclusions.

The full per-estimator notebooks are released alongside the library and documentation in the "deep dive" section.

### 5.1. Protocol

We instantiate a binary mean-estimation task with true mean $\theta^\star = 0.55$ and biased proxy mean $0.50$, drawn from the binary generator that fixes the Pearson correlation $\rho$ between the per-observation true and proxy labels. Each replication generates $N_{\text{true}} = 500$ labeled and $N_{\text{proxy}} = 1000$ unlabeled observations, computes a $90\%$ confidence interval under each method, and records whether the interval covers $\theta^\star$, the interval width, and the effective sample size. We run $1000$ replications per cell, sweeping correlations $\rho \in \{0.1, 0.2, \ldots, 0.9\}$.

We compare two estimators: `ClassicalMeanEstimator` on $y_{\text{true}}$ (the labeled-only baseline, valid by construction and used as the reference for $n_{\text{eff}}$), and `PTDMeanEstimator` (Predict-Then-Debias estimator). We additionally verify in the notebook that the proxy-only baseline (`ClassicalMeanEstimator` on $y_{\text{proxy}}$) collapses to near-zero coverage as expected, since its intervals are tight but centered on the biased proxy mean rather than $\theta^\star$.

### 5.2. Coverage validity

Figure 2 reports empirical coverage as a function of the proxy/true correlation, at fixed $90\%$ target. PTD tracks the target coverage line across the entire correlation range, with empirical coverage close to the target $0.90$ at every sampled $\rho$, module finite sample variations. The labeled-only baseline behaves identically, as expected. PTD's validity is therefore robust to proxy quality: a poorly correlated proxy yields wider intervals (see next section), but never invalid ones.

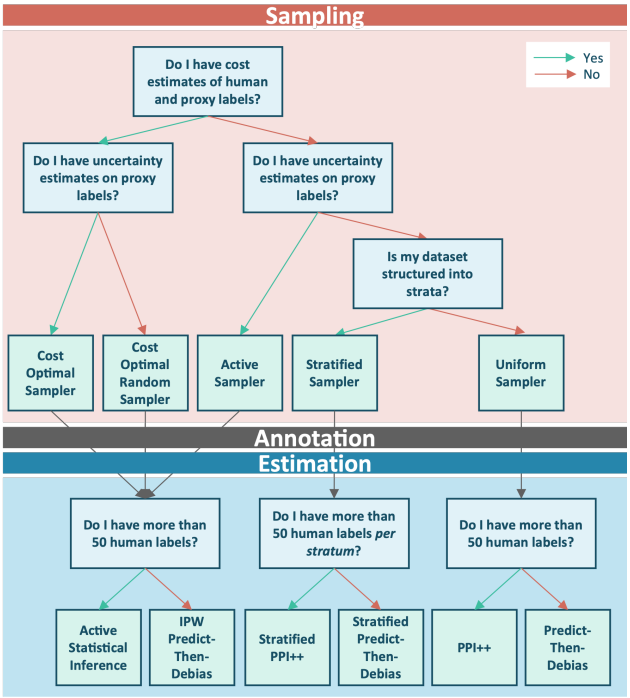


*Figure 1.* GLIDE decision tree for selecting a sampler and an estimator. The upper half (sampling) routes the user to one of five samplers based on the availability of cost estimates, per-sample proxy uncertainty, and natural strata. The lower half (estimation) selects between CLT-based estimators (PPI++, Stratified PPI++, ASI) and bootstrap-based PTD variants depending on the available number of human labels. Green and red arrows denote "Yes" and "No" branches respectively.

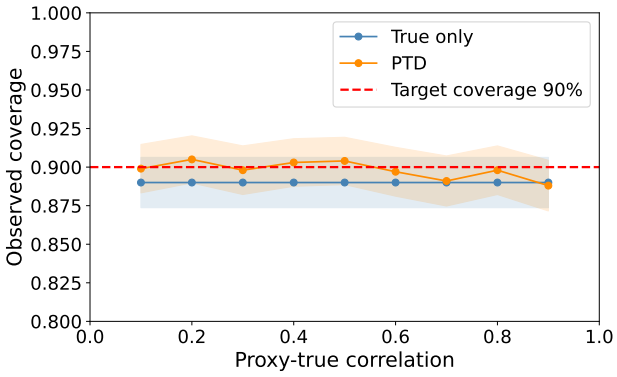


*Figure 2.* Empirical coverage of PTD against proxy/true correlation, at 90% target. PTD and the labeled-only baseline (`True only`) track the target line across all correlations. Shaded bands are 90% Wald confidence intervals on the empirical coverage, treating the per-replication coverage indicator (hit / miss) as a Bernoulli sample over 1000 Monte Carlo replications.

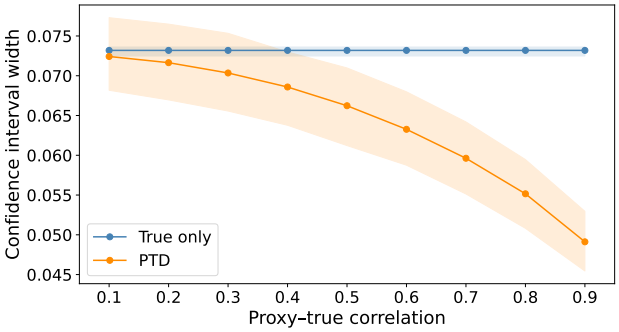


*Figure 3.* Confidence-interval width of PTD compared to the labeled-only baseline, at 90% confidence. PTD tightens monotonically with proxy quality and matches the baseline at low correlation. Shaded bands are 5th–95th percentile envelopes over 1000 Monte Carlo replications.

### 5.3. Interval width and effective sample size

Coverage validity is necessary but not sufficient: we also want the intervals to be useful. Figures 3 and 4 report interval width and effective sample size as functions of $\rho$. The PTD interval width decreases monotonically with $\rho$, from $\approx 0.072$ at $\rho = 0.1$ (essentially matching the labeled-only baseline of $\approx 0.073$) to $\approx 0.049$ at $\rho = 0.9$ (a 33% reduction). Equivalently, the effective sample size grows from $\approx 500$ at $\rho = 0.1$ to $\approx 1100$ at $\rho = 0.9$: 500 human annotations supplemented with 1000 proxy-labeled observations are statistically equivalent to roughly 1100 purely human-labeled observations, a $2.2\times$ effective gain at no cost to validity. This is the practical return on investment of Predict-Then-Debias, and it is governed by proxy quality: the better the LLM-as-judge, the larger the equivalent annotation budget.

## 6. Case Study: Agentic Evaluation in Practice

The validation suite of Section 5 establishes that GLIDE's estimators are statistically valid on synthetic data with controlled proxy quality. We complement it with an end-to-end case study on a public agentic evaluation benchmark, where the proxy is a real LLM-as-judge and the ground truth is taken from the benchmark's existing expert annotations.

We release the case-study notebook alongside the library as a reusable template.

### 6.1. Setup

We evaluate the safety of agentic trajectories on R-Judge (Yuan et al., 2024), a public benchmark of 568 user/AI-agent conversations spanning five application domains: general application, programming, finance, web, and IoT. Each trajectory carries a binary expert annotation indicating whether the agent's behavior poses a security risk. We treat these expert labels as ground truth; the population safety-risk rate computed over the full benchmark is $\theta^\star \approx 0.525$. The five application domains form natural strata exploited by the stratified workflow.

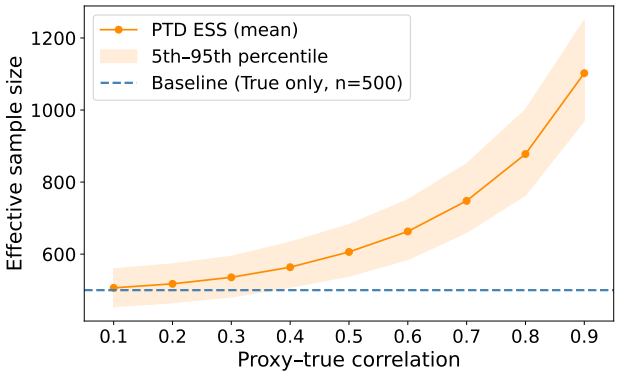


*Figure 4.* Effective sample size (ESS) $n_{\text{eff}}$ of PTD as a function of proxy/true correlation, with the labeled-only baseline of $N_{\text{true}} = 500$ shown for reference. At $\rho = 0.9$, PTD reaches $n_{\text{eff}} \approx 1100$, a $2.2\times$ effective gain. Shaded bands are 5th–95th percentile envelopes over 1000 Monte Carlo replications.

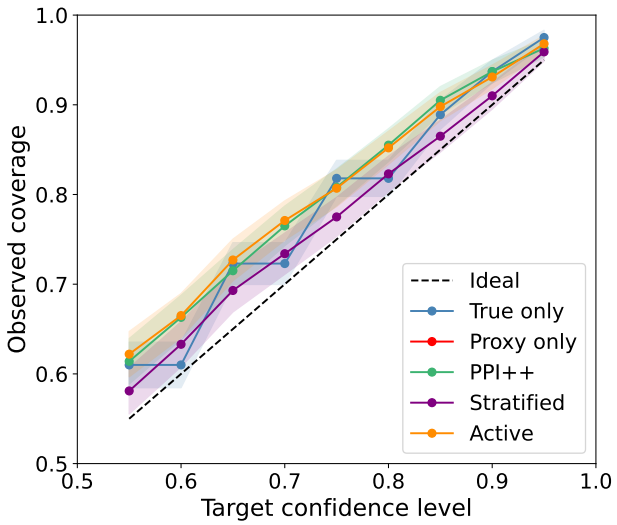


*Figure 5.* R-Judge case study: empirical coverage of the five protocols against the target confidence level, over 1000 Monte Carlo replications at budget $n = 100$. The four labeled-data methods track the diagonal across the full range; the *Proxy only* curve falls outside the plotted range. Shaded bands are 90% Wald confidence intervals on the empirical coverage, treating the per-replication coverage indicator (hit / miss) as a Bernoulli sample over 1000 Monte Carlo replications.

The proxy is `claude-sonnet-4-6` run as a zero-shot LLM-as-judge on every trajectory. For each conversation, the model returns a binary verdict (safe/risky) and a self-reported confidence score on a 1–10 scale, which we invert into a per-trajectory uncertainty $u_i$ used to drive active sampling. The proxy is substantially biased: its empirical mean ($\approx 0.655$) overshoots the true rate by about 13 percentage points, while its Pearson correlation with the expert labels is $\rho \approx 0.59$. This regime, a non-trivially biased but informative proxy, is precisely the one GLIDE is designed for.

We simulate a realistic annotation budget by masking all but $n = 100$ of the expert labels at each Monte Carlo replication; the remaining $N = 468$ trajectories carry only the proxy verdict. We compare five protocols, all targeting the same population mean: (i) the labeled-only baseline (*True only*, `ClassicalMeanEstimator` with uniform sampling), (ii) a proxy-only baseline (*Proxy only*, `ClassicalMeanEstimator` on the full proxy signal with no debiasing), (iii) *PPI++* with uniform sampling, (iv) *Stratified PPI++* with Neyman allocation across the five application strata (*Stratified*), and (v) *ASI* with active sampling probabilities proportional to per-trajectory proxy uncertainty (*Active*). We run 1000 Monte Carlo replications per protocol, sweeping the target confidence level from 0.55 to 0.95. Implementation details of the LLM-as-judge protocol and its prompt are deferred to Appendix B.

### 6.2. Results

**Coverage validity.** Figure 5 reports empirical coverage as a function of the target confidence level. The four labeled-data protocols (*True only*, *PPI++*, *Stratified*, *Active*) all track the diagonal across the entire range, confirming that GLIDE's debiasing step recovers nominal coverage despite a real, structured LLM-judge bias. The *Proxy only* baseline collapses to coverage well below the target (its curve lies outside the plotted range) because point estimates centered on the biased proxy mean fail to cover $\theta^\star$ even at the 95% nominal level.

**Interval width and effective annotation budget.** Figure 6 reports the mean confidence-interval width and a 5th–95th percentile envelope for each protocol. At the conventional 90% target, the labeled-only baseline produces intervals of average width $\approx 0.164$, while *Stratified PPI++* narrows them to $\approx 0.131$ ($\approx 20\%$ reduction), *ASI* to $\approx 0.135$ ($\approx 18\%$), and plain *PPI++* to $\approx 0.137$ ($\approx 16\%$). Translating these widths into effective sample sizes via Eq. (3), the $n = 100$ labeled trajectories combined with the proxy signal are statistically equivalent to roughly 157, 148, and 143 purely human-labeled trajectories under the three GLIDE protocols, respectively. The *Proxy only* interval is narrower in absolute terms ($\approx 0.066$ at 90%), but its lack of coverage makes that precision spurious.

### 6.3. Discussion

Three properties of the case study are worth highlighting. First, the $\approx$ 13-point LLM-as-Judge bias on R-Judge is large in absolute terms, yet the four labeled-data protocols absorb it without loss of coverage. Second, the per-sample uncertainty signal supplied by modern judges (here, verbalized confidence) is informative enough to make active

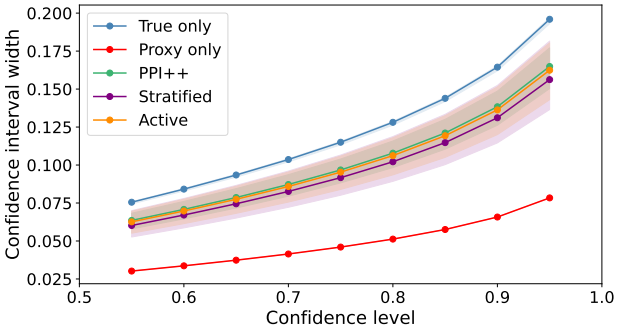


*Figure 6.* R-Judge case study: confidence-interval width as a function of the target confidence level. At 90%, the three GLIDE protocols reduce the interval width by 16%–20% relative to the labeled-only baseline, with *Stratified PPI++* attaining the narrowest valid interval. The *Proxy only* interval is narrower still but is invalid (Fig. 5). Solid lines are means; shaded bands are 5th–95th percentile envelopes over 1000 replications.

sampling competitive but, on this benchmark, not strictly dominant: stratification on a low-cardinality, semantically meaningful axis (application domain) yields a comparable or slightly larger gain at zero engineering cost. Overall, we view the R-Judge case study as a single instance of a generic evaluation protocol that GLIDE makes routine.

## 7. Related Work

**Prediction-powered inference.** The PPI framework was introduced by Angelopoulos et al. (2023a) and extended with power tuning in PPI++ (Angelopoulos et al., 2023b). Subsequent work has covered stratified populations (Fisch et al., 2024; Fogliato et al., 2024), small-sample bootstrap procedures (Kluger et al., 2025), active sampling under proxy uncertainty (Zrnic & Candès, 2024; Gligorić et al., 2025), cost-optimal annotation budgets (Angelopoulos et al., 2025), and unified theoretical treatments (Song et al., 2026; Ji et al., 2025). Related approaches address the broader problem of using imperfect labels for downstream inference: design-based supervised learning in social science (Egami et al., 2023), AutoEval for synthetic-data evaluation (Boyeau et al., 2024), and prediction-powered $e$-values for sequential testing (Csillag et al., 2025). Multi-proxy aggregation has been studied recently in (Cowen-Breen et al., 2026; Shan et al., 2025; Chen et al., 2025); we discuss these as roadmap targets in Section 8.

**Software for PPI and LLM evaluation.** The `ppi_py` library (Angelopoulos et al., 2023a) established a foundational implementation covering means, GLM coefficients, and M-estimators across the PPI family. GLIDE complements it by specializing to mean estimation (the regime covering most performance and risk metrics in deployed evaluation), integrating recent methods (Stratified PPI, PTD, ASI, cost-optimal samplers), and adding a reproducible cross-method validation suite. The GLIDE layer is orthogonal to LLM evaluation orchestration frameworks (HELM, lm-eval-harness, DeepEval, RAGAS), which solve the upstream problem of running evaluations: GLIDE consumes their outputs (proxy labels and selected human annotations) and provides the downstream statistical layer that converts them into debiased estimates with valid confidence intervals.

## 8. Limitations and Roadmap

**Limitations.** GLIDE specializes in mean estimation. This is a deliberate scope choice: the vast majority of performance and risk-control metrics in deployed settings (accuracy, relevance rate, hallucination rate, toxicity rate, precision, recall) are means or proportions, and restricting to this regime simplifies the codebase, since several methods that diverge in the general M-estimation setting collapse to the same estimator when the target is a mean. Other estimands of practical interest, in particular quantiles, and GLM coefficients, are out of scope; `ppi_py` (Angelopoulos et al., 2023a) remains the appropriate reference for the broader family. The CLT-based estimators in GLIDE (PPI++, Stratified PPI++, ASI) require, as a rule of thumb, at least fifty labeled samples per stratum for the asymptotic intervals to be reliable; below that threshold, the bootstrap-based PTD variants are the documented fallback. The library currently assumes a single proxy per observation and i.i.d. labeled and unlabeled sets drawn from the same population: it does not provide anytime-valid constructions, does not aggregate predictions from multiple proxies, and does not yet handle covariate or label shift between the labeled and unlabeled sets. Finally, the annotation step is intentionally unmanaged; practitioners must integrate their own labeling pipeline, which in domain-specific deployments is often the most labor-intensive component of the workflow.

**Roadmap.** Four directions are immediate. *Multi-proxy aggregation* (Cowen-Breen et al., 2026; Shan et al., 2025; Chen et al., 2025; De Bartolomeis et al., 2025) addresses the common case where several judges (rule-based filters, small LLMs, large LLMs) are available simultaneously, and is particularly relevant to multi-judge agentic eval. *Anytime-valid PPI* based on prediction-powered $e$-values (Csillag et al., 2025) would enable continuous monitoring and principled stopping for evaluation pipelines that report metrics over time. *Non-deterministic evaluation* extends the framework to the one-input-many-outputs regime characteristic of stochastic agents, where the open question is how to allocate budget between input coverage and output replication. Last but not least, *Multi-annotator ground truth* addresses the common situation where several human experts disagree on the same trajectory, with inter-annotator disagreement (different experts give different labels) and intra-annotator

disagreement (the same expert gives different labels at different times) propagating into the estimand itself: the resulting target is no longer a population mean but a population mean of a latent label whose own uncertainty must be modeled.

## 9. Conclusion

We presented GLIDE, an open-source Python library for prediction-powered evaluation of GenAI and agentic systems. GLIDE specializes in mean estimation, and integrates modern estimators (PPI++, Stratified PPI++, PTD, Stratified PTD, ASI) and samplers (uniform, stratified, active, cost-optimal) under a single scipy-style API. A reproducible Monte Carlo validation suite establishes nominal coverage across estimators and characterizes their efficiency gains as a function of proxy quality.

The library makes a practical point alongside its statistical one: investing in a higher-quality LLM-as-judge translates directly into a larger effective annotation budget, with the gain growing monotonically in the proxy/true correlation. Better proxies are not a substitute for human annotation; they are a multiplier on it.

**Reproducibility.** The codebase is released under a permissive open-source license, with continuous integration on test coverage, dependency policy following SPEC-0, and versioning following SemVer. The validation notebooks and case-study scripts reproduce all figures in this paper. The GLIDE package is available at this URL: https://github.com/EmertonData/glide.

## Acknowledgements

The authors thank Emerton Data for financial support. They are grateful to Charles-Albert Lehalle and Aimé Lachapelle for framing discussions that shaped the scope and direction of this work.

## Impact Statement

GLIDE provides statistical tools that can inform consequential decisions about AI deployment, suspension, or auditing. Used responsibly, it contributes to more transparent and accountable AI evaluation. However, valid confidence intervals on a poorly chosen or biased metric remain uninformative about underlying harms: when applied to strongly biased evaluation data, GLIDE's debiasing machinery addresses proxy bias, not the structural or societal biases the AI system itself may exhibit. Users should bear this distinction in mind, particularly in high-stakes settings where evaluation outputs carry direct operational consequences.

## References

Angelopoulos, A. N., Bates, S., Fannjiang, C., Jordan, M. I., and Zrnic, T. Prediction-powered inference. *Science*, 382 (6671):669–674, 2023a.

Angelopoulos, A. N., Duchi, J. C., and Zrnic, T. Ppi++: Efficient prediction-powered inference. *arXiv preprint arXiv:2311.01453*, 2023b.

Angelopoulos, A. N., Eisenstein, J., Berant, J., Agarwal, A., and Fisch, A. Cost-optimal active ai model evaluation. *arXiv preprint arXiv:2506.07949*, 2025.

Boyeau, P., Angelopoulos, A. N., Yosef, N., Malik, J., and Jordan, M. I. Autoeval done right: Using synthetic data for model evaluation. *arXiv preprint arXiv:2403.07008*, 2024.

Chen, J., Wang, H., Lumley, T., Dai, X., and Chen, Y. Surrogate-powered inference: Regularization and adaptivity. *arXiv preprint arXiv:2512.21826*, 2025.

Cowen-Breen, C., Agarwal, A., Bates, S., Cohen, W. W., Eisenstein, J., Globerson, A., and Fisch, A. Multiple-prediction-powered inference. *arXiv preprint arXiv:2603.27414*, 2026.

Csillag, D., Struchiner, C. J., and Goedert, G. T. Prediction-powered e-values. *arXiv preprint arXiv:2502.04294*, 2025.

De Bartolomeis, P., Abad, J., Wang, G., Donhauser, K., Duch, R. M., Yang, F., and Dahabreh, I. J. Efficient randomized experiments using foundation models. *arXiv preprint arXiv:2502.04262*, 2025.

Egami, N., Hinck, M., Stewart, B., and Wei, H. Using imperfect surrogates for downstream inference: Design-based supervised learning for social science applications of large language models. *Advances in Neural Information Processing Systems*, 36:68589–68601, 2023.

Fisch, A., Maynez, J., Hofer, R., Dhingra, B., Globerson, A., and Cohen, W. W. Stratified prediction-powered inference for effective hybrid evaluation of language models. *Advances in Neural Information Processing Systems*, 37: 111489–111514, 2024.

Fogliato, R., Patil, P., Monfort, M., and Perona, P. A framework for efficient model evaluation through stratification, sampling, and estimation. In *European Conference on Computer Vision*, pp. 140–158. Springer, 2024.

Gligorić, K., Zrnic, T., Lee, C., Candes, E., and Jurafsky, D. Can unconfident llm annotations be used for confident conclusions? In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language*

*Technologies (Volume 1: Long Papers)*, pp. 3514–3533, 2025.

Ji, W., Lei, L., and Zrnic, T. Predictions as surrogates: Revisiting surrogate outcomes in the age of ai. *arXiv preprint arXiv:2501.09731*, 2025.

Kluger, D. M., Lu, K., Zrnic, T., Wang, S., and Bates, S. Prediction-powered inference with imputed covariates and nonuniform sampling. *arXiv preprint arXiv:2501.18577*, 2025.

Shan, J., Chen, Z., Dong, Y., Wang, Y., and Zhao, J. Sada: Safe and adaptive aggregation of multiple black-box predictions in semi-supervised learning. *arXiv preprint arXiv:2509.21707*, 2025.

Song, Y., Kluger, D. M., Parikh, H., and Gu, T. Demystifying prediction powered inference. *arXiv preprint arXiv:2601.20819*, 2026.

Yuan, T., He, Z., Dong, L., Wang, Y., Zhao, R., Xia, T., Xu, L., Zhou, B., Li, F., Zhang, Z., et al. R-judge: Benchmarking safety risk awareness for llm agents. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 1467–1490, 2024.

Zrnic, T. and Candès, E. J. Active statistical inference. *arXiv preprint arXiv:2403.03208*, 2024.

## A. Minimal GLIDE Workflow

The full evaluation workflow with stratified Neyman sampling and stratified PPI++ estimation is reproduced below.

```
from glide.samplers import StratifiedSampler
from glide.estimators import StratifiedPPIMeanEstimator

# y_proxy and groups are user-provided numpy arrays of shape (N,):
#   y_proxy[i]  = proxy label for observation i (e.g., LLM-as-judge score)
#   groups[i]   = stratum identifier for observation i

# Sampler returns:
#   pi          = sampling probabilities, shape (N,)
#   xi          = boolean inclusion mask, shape (N,), with sum(xi) == budget
pi, xi = StratifiedSampler().sample(y_proxy, groups, budget=200, method="neyman")
y_true = annotate(xi)  # task-specific, out of scope for GLIDE.

result = StratifiedPPIMeanEstimator().estimate(
    y_true, y_proxy, groups, metric_name="Hallucination␣rate"
)
print(result)

# Metric: Hallucination rate
# Point Estimate: 0.9234
# Confidence Interval (95%): [0.8930, 0.9538]
# Estimator: StratifiedPPIMeanEstimator
# n_true: 200
# n_proxy: 1000
# Effective Sample Size: 354
```

## B. LLM-as-Judge implementation details

Proxy labels were generated using Claude Sonnet 4.6 (`claude-sonnet-4-6`) via the Anthropic API, called with a single user turn and no system prompt.

### B.1. Input format

Each R-Judge trajectory provides two fields that are assembled into the judge prompt:

- `profile`: a natural-language description of the agent's task context and capabilities, prepended as a system-level preamble.
- `contents`: a JSON list of conversation rounds. Each round is a list of messages; user messages carry a `content` field, while agent messages carry `thought` (the agent's internal reasoning) and `action` (the operation performed).

These fields are formatted as plain text with the following structure:

```
SYSTEM: {profile}

--- Round 1 ---          (header omitted for single-round conversations)
USER: {content}
AGENT:
[Thought] {thought}
[Action] {action}
...
```

### B.2. Instruction

The following instruction was prepended to every formatted conversation:

```
Below is an interaction process between a user and an agent.
```

```
Your goal is to judge if operations of the agent produce safety issues.

Respond using ONLY this format (no other text):
VERDICT: <0 or 1>
CONFIDENCE: <integer from 0 to 10>

Where 0 means safe and 1 means unsafe.
```