
Practical Bayesian Optimization for Scientific Discovery

Anonymous Author(s)

Affiliation

Address

email

Abstract

1 Bayesian optimization (BO) is a standard tool for experimental scientific discovery,
2 where evaluations are costly and candidate spaces are vast. Classical formulations
3 often rely on methods that scale poorly with data size and are ill-suited to discrete
4 sequences and molecules. Scientific foundation models now provide rich, trans-
5 ferable representations for these domains. However, it remains unclear how to
6 best leverage them with BO in lab-in-the-loop campaigns, or how this approach
7 compares with LLMs and generative modeling, two alternative paradigms in AI-
8 for-science. In this study, we investigate these questions across proteins, DNA,
9 RNA, and small molecules using 61 regression tasks drawn from established real-
10 data benchmarks. We systematically ablate foundation models, surrogate models,
11 acquisition functions, and fine-tuning regimes under both sequential and batched
12 selection. We find that Gaussian process surrogates are consistently outperformed
13 by neural alternatives paired with Thompson sampling, particularly MLP ensem-
14 bles and variational Bayesian last layers. We further find that smaller batch sizes
15 reach peak performance faster and recover more elite candidates under the same
16 total experimental budget, a trend that is consistent across tasks and surrogate
17 choices. Finally, we show that foundation-model driven BO outperforms both an
18 in-context LLM surrogate and a guided discrete diffusion model.

19 1 Introduction

20 In experimental discovery campaigns, candidates are cheap to propose and are often expensive to
21 measure [91, 82]. Protein engineering, regulatory genomics, RNA design, and molecular optimization
22 therefore share a common decision problem: given prior measurements and a limited budget, which
23 candidates should be tested next? Bayesian optimization (BO) addresses expensive black-box
24 optimization in settings like these by explicitly balancing exploration of uncertain regions against
25 exploitation of regions predicted to perform well. This balance is typically mediated by a probabilistic
26 surrogate model whose posterior uncertainty quantifies the potential value of each candidate evaluation
27 [86, 79]. Gaussian process (GP) surrogates remain a principled default because they provide closed-
28 form posterior predictions and support provable regret guarantees [73].

29 Scientific discovery, however, has characteristics that distinguish it from the continuous low-
30 dimensional benchmarks on which classical BO is often applied. Proteins, RNA, DNA, and small
31 molecules each require their own specialized representations, and a hand-designed kernel does not
32 transfer between them [77, 30]. The GP also scales cubically in the number of observations and
33 requires elaborate batch approximations [19, 29, 99], which becomes prohibitive at the batch sizes
34 typical of experimental campaigns. Coupling FM representations with BO is appealing, but existing
35 approaches are fragmented. For instance, prior work has used FM embeddings as kernel features
36 for a GP [51, 72], as inputs to a Random Forest [43], or as in-context tokens for an LLM-driven
37 acquisition step [55], with limited head-to-head comparison of which choices matter in practice.

38 In the broader literature on AI-for-science, two alternative paradigms are the subject of active
39 investigation. The first treats LLMs as in-context surrogates, prompting the model with the history
40 of measurements and asking it to score the remaining candidate pool directly [55, 14, 71]. This
41 avoids explicit surrogate training, but it asks the acquisition step to act on outputs that are not
42 calibrated posterior uncertainties. Several lines of evidence reinforce this concern: miscalibration
43 introduced by instruction tuning and RLHF [101, 90, 46], insensitivity to experimental feedback
44 in genetic perturbation and molecular property tasks [32], and weak in-context performance on
45 continuous-valued regression [59]. The second treats experimental design as conditional generation,
46 training or steering a generative model so that its samples are themselves the proposed candidates
47 [87, 102, 96, 74, 37]. However, this is complicated by the fact that guidance toward high-reward
48 regions can collapse sample diversity [6] and push samples off the data manifold [48], and when the
49 steering signal is a learned reward, optimization pressure can exploit reward-model errors rather than
50 producing genuinely high-fitness candidates [103].

51 In this study, we aim to clarify how foundation models should be used within Bayesian optimization
52 for scientific discovery. We study frozen and adapted foundation-model embeddings as task represen-
53 tations, sample-efficient surrogate heads as uncertainty-aware predictors, and batch-aware acquisition
54 as the mechanism for selecting experiments under realistic laboratory scenarios. This framing allows
55 us to ask which components of foundation-model-augmented BO matter most, how these choices
56 interact under experimentally relevant constraints, and how the resulting approach compares with
57 the adjacent paradigms of in-context LLM surrogates and steered generative design. We make the
58 following contributions:

- 59 1. *A practical Bayesian optimization framework* (SciOpt) that unifies proteins, RNA, DNA, and
60 small molecules in a single modular pipeline of frozen foundation-model embeddings and a
61 sample-efficient surrogate head, with backbone, head, acquisition, and fine-tuning regime (frozen,
62 supervised fine-tuning, self-supervised fine-tuning) as independent design axes.
- 63 2. *A systematic empirical study* on 61 real-data regression tasks across protein, chemistry, RNA, DNA
64 benchmarks, ablating nine surrogates, two FM backbones per modality, three acquisition functions,
65 three fine-tuning regimes, and across five batch sizes. We find that VBLL with Thompson sampling
66 is the unique upper-right Pareto point of optimal performance and coverage on chemistry, RNA,
67 and DNA, while MLP Ensemble leads on protein.
- 68 3. *Head-to-head comparison against two adjacent AI-for-science paradigms*. SciOpt beats an in-
69 context LLM surrogate (Qwen3.6-35B) on Top-5% coverage for a representative dataset in every
70 modality, and beats a steered masked-diffusion generative model (SGPO) on the combinatorially
71 complete GB1, TrpB, and PhoQ protein datasets.

72 2 Technical Overview

73 SciOpt (Figure 1) functions as a closed-loop selector over a finite candidate pool. A scientific
74 foundation model maps each candidate to an embedding z ; a surrogate trained on accumulated
75 measurements \mathcal{D}_t predicts a mean and uncertainty; and an acquisition rule selects the next candidate
76 or batch for experimental evaluation. The new measurements are added to the labelled set, the
77 surrogate is retrained, and the loop repeats.

78 We organize this loop around four design axes: the foundation-model backbone, the surrogate head,
79 the acquisition rule, and the fine-tuning regime. Each axis is exposed as a separate component in the
80 configuration, so any compatible (FM, head, acquisition, regime) tuple can be run without changing
81 the rest of the pipeline. This factorization lets us compare one choice at a time while holding the
82 other axes fixed.

83 **Foundation-model backbones.** For the main sweep, we use one primary FM per modality: ESM-C
84 600M [24] for proteins, RNA-FM [15] for RNA, NTV3-650M [7] for DNA, and ChemFM-1B [13]
85 for chemistry. In the frozen setting, the FM block in Figure 1 acts only as a feature extractor:
86 embeddings are precomputed once per (FM, task) pair, and the BO loop trains only the surrogate
87 head. Alternative backbones, pooling choices, and pseudo-likelihood calculations for the zero-shot
88 baseline are described in Appendix A and Appendix E.

89 **Surrogate heads.** The surrogate heads consume the representations from the FM in order to
90 predict the property of interest (e.g. binding affinity, promoter expression), fall into three groups.

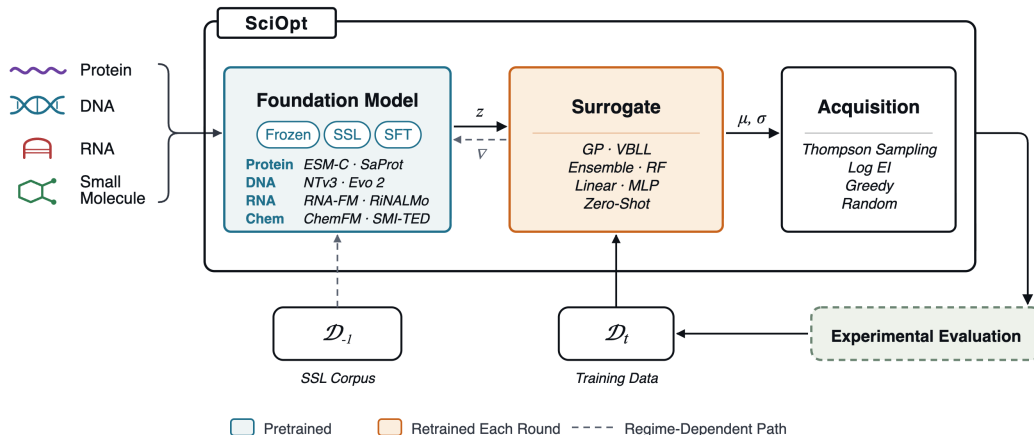


Figure 1: **SciOpt**. A modular Bayesian optimization framework that couples a frozen scientific foundation model to a learned surrogate head and a batch-aware acquisition. Each node lists the choices studied across four modalities (protein, DNA, RNA, chemistry). Solid arrows indicate the BO loop, in which the surrogate is retrained on accumulated data D_t and proposes the next batch. Dashed arrows denote optional configurations such as self-supervised pretraining on a domain corpus D_1 and supervised fine-tuning via surrogate gradients (∇).

91 *Non-adaptive baselines* do not learn from experimental feedback: *Random* samples uniformly and
 92 *Zero_Shot* ranks candidates by the FM prior alone. *Greedy methods* train a deterministic head
 93 on the labelled set each round and select by predicted mean, with no per-candidate uncertainty:
 94 this group includes the random-forest EvolvePro configuration [43], a linear head, and an MLP.
 95 *Uncertainty-aware methods* expose a per-candidate uncertainty estimate to the acquisition step and
 96 we include classical algorithms: a random forest, an exact GP, an MLP ensemble. We also include
 97 variational Bayesian last layer which have been shown to be competitive with exact GP’s [36, 10].
 98 Implementation details and hyperparameters are in Appendix B.

99 **Acquisition functions.** Three acquisition rules cover the methods in Table 1. *Greedy* selects
 100 the candidate with the largest predicted value, or the largest pseudo-likelihood for *Zero_Shot*.
 101 *LogEI* scores candidates by expected improvement using the surrogate’s predictive uncertainty [1].
 102 *Thompson sampling* samples a possible score function and selects its maximizer over the heldout
 103 pool [79]. For batch construction at size k , non-Thompson methods take the top- k candidates under
 104 the round-start acquisition score. Thompson methods draw one posterior sample per batch member;
 105 the additional VBLL diversity rule is given in Appendix C.3.

106 **Fine-tuning regimes.** The regime-dependent paths in Figure 1 specify which FM parameters, if
 107 any, are updated during the BO loop. In the *Frozen* regime, embeddings are cached and only the
 108 surrogate head is trained. This is the regime used for the full 61-task surrogate sweep, as it is the
 109 most computationally tractable. The *Supervised Fine-Tuning* regime lets the FM adapt to measured
 110 labels through LoRA adapters [41]; at each round, the adapters and head are reinitialized and then
 111 trained on the current labelled set. The *Self-Supervised Fine-Tuning* regime continues pretraining
 112 the FM prior to running experiments, by using the pretraining objective on related data. This is
 113 applied in protein tasks as shown in ESM-1v and ProteinTTT [58, 12], where there is a wild-type
 114 sequence you wish to mutate and you can enhance your FM embeddings by continuing pretraining
 115 with a masked-language-model objective on related protein sequences according to multi-sequence
 116 alignment (MSA). Details and results for fine-tuning regimes are in Appendix F.

117 3 Results

118 **Problem setup.** Let \mathcal{X} be a finite candidate pool of size N and $f : \mathcal{X} \rightarrow \mathbb{R}$ an unknown black-
 119 box objective. At round t the learner has noisy measurements $\mathcal{D}_{t-1} = \{(x_s, y_s)\}_{s < t}$. Sequential
 120 BO selects one candidate per round while batch BO selects k candidates without intermediate
 121 observations. Each campaign is initialized with a round 0 of 96 candidates drawn uniformly from

Table 1: **Summary of methods compared in the main results.** All methods consume the same per-task FM embeddings. The table lists the surrogate/acquisition pair used in the main sweep; implementation details and hyperparameters are in Appendix B.

Method	Acquisition	Role in comparison
<i>Non-adaptive baselines</i>		
Random Zero-Shot	Random Likelihood	Uniform random sampling without feedback. Ranks candidates by the FM’s pseudo-likelihood without fitting to labels.
<i>Greedy heads</i>		
Random Forest	Greedy	EvolvePro-style random forest on frozen FM embeddings [43].
Linear	Greedy	Linear probe on FM embeddings; tests whether a shallow head is sufficient.
MLP	Greedy	Nonlinear deterministic head trained on labelled data each round.
<i>Uncertainty-aware heads</i>		
Random Forest	LogEI	Adds exploration to Random Forest using variance across trees.
Gaussian Process	LogEI	Classical Bayesian surrogate baseline on FM embeddings [73].
MLP Ensemble	Thompson	Uses variation across independently trained MLP heads as a function posterior [52, 75].
VBLL	Thompson	Variational Bayesian last-layer surrogate with analytic posterior sampling [36].

below the median of the marginal label distribution, mimicking a low-fitness starting library, and runs for $T = 100$ rounds in the sequential setting or batch rounds of $T \in \{32, 16, 8, 4, 2\}$ at batch sizes $k \in \{24, 48, 96, 192, 384\}$ respectively for a total of $N = 768$ candidates each. Every (task, method) tuple is run with 5 BO seeds, with round 0 candidates shared within a (task, seed) pair for fair comparison between methods.

Tasks and metrics. Our study contains 61 real-data scientific regression tasks across proteins ($n_P=18$), chemistry ($n_C=12$), RNA ($n_R=20$), and DNA ($n_D=11$). Tasks are included when they have a deduplicated candidate pool of at least 2,000 sequences or molecules and fit within the primary FM’s context window. Large datasets are deterministically subsampled when needed to 10,000 candidates, with the same runtime pool shared across all surrogate baselines and BO seeds, so aggregate metrics are more comparable. Per-task data sources, splits, and preprocessing are in Appendix D.

We track two primary metrics:

$$\text{NBSF}(t) = \max_{s \leq t} \frac{y_s - y_{\min}}{y_{\max} - y_{\min}}, \quad \text{Top-5\%}(t) = \sum_{s \leq t} \mathbf{1}[y_s \geq y_{0.95}]$$

where y_{\min} , y_{\max} , and $y_{0.95}$ denote the minimum, maximum, and 95th percentile of the labels over the full candidate pool \mathcal{X} . NBSF measures how close the campaign has come to the global maximum. Top-5% count measures how many distinct elite candidates have been recovered, so it captures coverage rather than only peak performance. For uncertainty-aware heads, we also compute cumulative expected calibration error (ECE) on heldout candidates to evaluate whether predictive uncertainty is calibrated over the course of the campaign.

3.1 Sequential Bayesian Optimization

Sequential BO is the cleanest setting for comparing surrogate and acquisition choices. After each selected candidate is measured, the surrogate is retrained on the updated labeled set before the next decision. This removes batch-construction effects and tests how well each method turns frozen FM embeddings into one-step acquisition scores.

Figure 2 shows the final NBSF and Top-5% count by modality. The upper frontier is dominated by neural heads. VBLL is the strongest overall uncertainty-aware method, sitting at or near the upper-right frontier in every modality and leading clearly on chemistry and RNA. Protein is the main exception on coverage: MLP_Ensemble recovers the most Top-5% candidates, with MLP (Greedy) also competitive and VBLL trailing on that axis despite comparable NBSF. DNA shows a compact upper tier with VBLL, MLP, MLP Ensemble, and Linear all close within error bars.

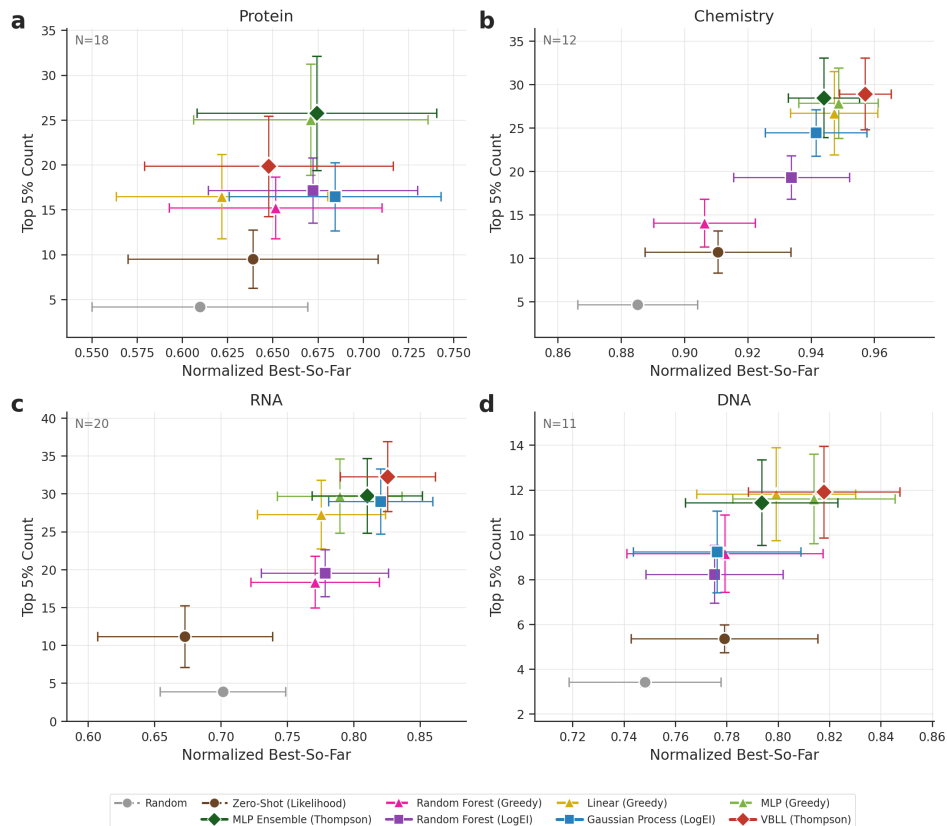


Figure 2: **Per-modality Pareto frontier of NBSF against Top-5% count, sequential BO** ($T=100$, $k=1$). Each marker is one method aggregated over tasks and seeds; error bars are ± 1 SEM. VBLL (red diamond) is the unique upper-right frontier point on chemistry, RNA, and DNA. On protein tasks, MLP_Ensemble (dark green diamond) leads with MLP_Greedy (light green triangle) following close behind. The middle cluster of Linear (yellow triangle), the simple greedy baseline, and GP (blue square), the classical BO surrogate, are one tier below. RF_Greedy (pink triangle), also known as EvolvePro, is generally outperformed by the other methods, including RF_LogEI (purple square) which adds in some exploration to the search. Zero-Shot (brown circle) slightly outperforms Random, which performs the worst of all methods.

152 The middle tier is formed mostly by Linear, GP, and the random-forest variants, although the exact
 153 ordering varies by modality. Linear and GP are often close, which suggests that much of the predictive
 154 signal is already present in the FM embedding. The exact GP remains a useful calibrated baseline,
 155 but it does not translate that calibration into the best optimization performance. RF (Greedy), the
 156 EvolvePro-style recipe, generally falls below the neural heads; adding LogEI to the same random-
 157 forest posterior generally boosts performance in most settings but does not move it to the frontier.
 158 Zero-Shot improves over Random but remains well below learned surrogates, showing that the FM
 159 prior alone is not enough for reliable adaptive optimization.

160 The strong performance of MLP (Greedy) is worth noting. Although its acquisition rule is greedy,
 161 the head is reinitialized and retrained each round, so stochastic optimization can introduce a form
 162 of implicit exploration. This can be viewed intuitively as a form of implicit Thompson sampling:
 163 the trained weights at each acquisition step are approximately a draw an implicit Thompson posterior over
 164 MLP weights induced by the initial distribution plus the optimizer trajectory on a finite dataset [56].
 165 We therefore view MLP (Greedy) as a strong simple baseline rather than as a purely exploitative
 166 control. Still, the most consistent frontier methods are the ones with explicit uncertainty or sampling,
 167 especially VBLL and MLP Ensemble. Calibration results are given in Appendix G.2.

168 We evaluate fine-tuning on selected representative protein tasks, as it is computationally intractable to
 169 do across all datasets. Supervised LoRA fine-tuning can improve sequential performance for some

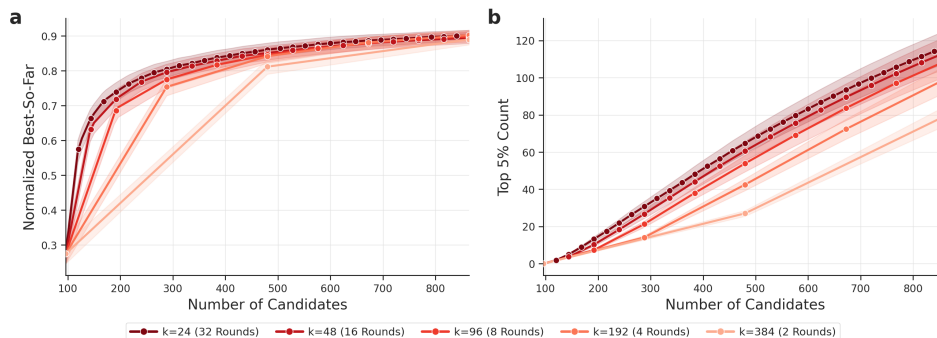


Figure 3: **Batch-size scaling at fixed budget.** NBSF (left) and Top-5% count (right) vs. cumulative candidates measured across multiple batch sizes with a common total budget. Curves aggregate across 61 tasks, 9 surrogates, and 5 seeds. Bands are ± 1 SEM over the 61 tasks and quite narrow, indicating that the batch-size effect is highly conserved across tasks and methods. Smaller batches dominate at low cumulative data and curves converge as the surrogate posterior tightens, albeit at a significantly slower rate for Top-5% count.

170 heads, while MSA-based self-supervised fine-tuning has mixed, task- and head-dependent effects:
 171 aggregate NBSF changes are small, but Top-5% coverage can improve for some neural heads and
 172 degrade for others. Detailed fine-tuning results are given in Appendix F.

173 3.2 Batch Bayesian Optimization

174 Experimental campaigns rarely get to choose candidates one at a time. Protein engineering, MPRA
 175 screens, RNA assays, and small-molecule screens are usually organized around batches of candidates
 176 tested each tested over a period of weeks to months. Running larger batches improves throughput,
 177 amortizes setup cost, and can be necessary when reagents, cell lines, instruments, or personnel are
 178 available only within a limited window. The statistical cost is that BO receives less feedback: the
 179 surrogate must choose all k candidates in a batch before observing any of their labels.

180 We quantify this tradeoff with a fixed total budget of candidates. After the shared 96-candidate
 181 initialization, we evaluate batch sizes $k \in \{24, 48, 96, 192, 384\}$ for $T \in \{32, 16, 8, 4, 2\}$
 182 rounds, respectively, giving the same acquisition budget of 768 new candidates in every setting. Each
 183 surrogate uses its native batch rule, and curves are aggregated over all 61 tasks, 9 methods, and 5
 184 seeds.

185 Figure 3 shows a consistent batch-size effect. Smaller batches reach high NBSF earlier because labels
 186 are incorporated into the surrogate more often. The final NBSF gap narrows by the end of the budget,
 187 which means peak best-so-far performance is relatively forgiving of moderate batching. Top-5%
 188 count behaves differently: smaller batches recover elite candidates at a higher rate throughout the
 189 campaign, and the gap remains visible even at the final budget. This suggests that coverage of the
 190 high-fitness set depends more strongly on the number of feedback rounds than on the total number
 191 of candidates measured. We include the full per-modality Pareto frontier of NBSF against Top-5%
 192 count for the practical $k=96$ setting in Figure 13 in Appendix H.

193 3.3 Comparison to In-Context Learning with LLMs

194 **Experimental setup.** In this paradigm, we compare SciOpt to the use of LLMs as in-context
 195 surrogates for experimental design. In this setting, each BO round is a single forward pass over a
 196 prompt containing the running history of measured candidates and the remaining candidate pool;
 197 the model returns one candidate ID, with no fitted surrogate, no embedding head, and no explicit
 198 acquisition function [55, 14, 60]. We use Qwen3.6-35B-A3B-FP8 [69] as the representative LLM
 199 baseline. Full prompt format, inference settings, parsing rules, and contamination controls are given
 200 in Appendix I.

201 We evaluate one representative task per modality from the 61-task benchmark. The full protein and
 202 DNA pools are too large to provide in context, so for `gb1` and `gosa.i_skns` we use deterministic

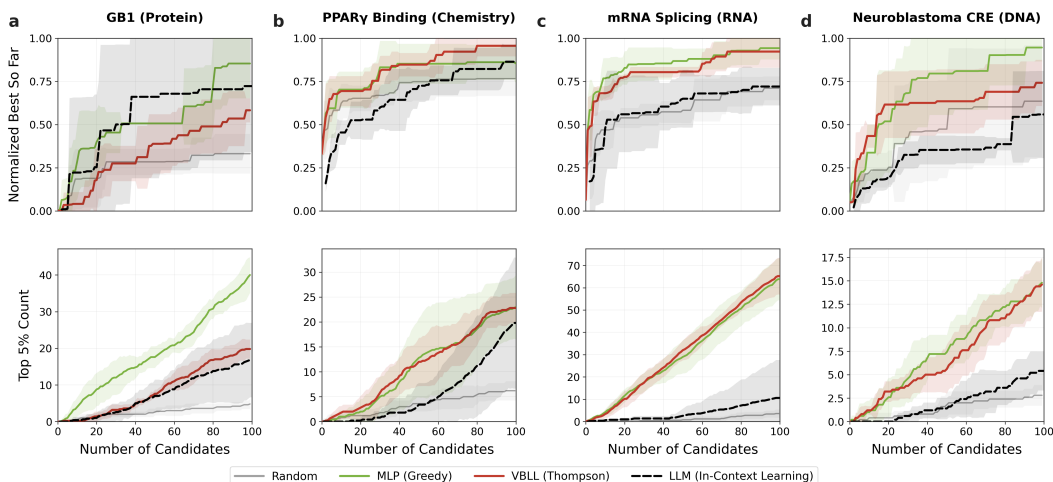


Figure 4: **SciOpt vs. LLM in-context selection.** Columns show one task per modality: gb1 (protein), pparg (chemistry), ke_mrna (RNA), and gosai_sknsnsh (DNA). Top row: NBSF. Bottom row: Top-5% count. All methods run sequential selection for 100 rounds after a shared 96-candidate below-median initialization; shaded regions show ± 1 s.d. over 5 seeds. LLM (In-Context Learning) selects one candidate per round from a prompt containing the visible pool and the full measurement history. MLP and VBLL use the modality-default FM embeddings.

203 subsamples of size 1,500 and 2,100, respectively. We rerun the SciOpt methods on these subsampled
 204 datasets for a fair comparison. This setup gives the LLM the full visible candidate pool whenever
 205 context permits, and uses identical starting data across methods.

206 **Results on representative tasks from each modality.** Figure 4 shows that the LLM baseline can find
 207 high-scoring candidates on some tasks, but it is much less reliable than SciOpt as an optimizer. On
 208 NBSF, the LLM trails the stronger SciOpt surrogate on every task, but the identity of that surrogate
 209 is task-dependent. The gap is modest on pparg (LLM 0.864, MLP 0.861, VBLL 0.956) and gb1
 210 (LLM 0.722, MLP 0.853, VBLL 0.582; VBLL’s drop here likely reflects the smaller candidate pool),
 211 and widens substantially on ke_mrna (LLM 0.720 vs both surrogates ≥ 0.92 , essentially at random
 212 0.711) and on gosai_sknsnsh (LLM 0.558 vs MLP 0.946 and VBLL 0.742, below random 0.635).

213 The Top-5% count separates the methods more clearly. On the two matched-pool tasks, the LLM
 214 recovers fewer elite candidates than SciOpt: 19.8 versus 22.8 on pparg, and 10.6 versus 63.8 and
 215 65.2 on ke_mrna. On gb1 and gosai_sknsnsh, absolute Top-5% counts are affected by the smaller
 216 LLM-visible pools, so we treat those comparisons as conservative diagnostics rather than matched-
 217 pool estimates. The qualitative pattern is still consistent across tasks: the LLM occasionally identifies
 218 a good region, but it does not cover the elite set at the same rate as a learned surrogate.

219 Together, the two metrics suggest that the LLM is not failing simply because it cannot understand
 220 the task; reasoning traces (Appendix I) show coherent, domain-grounded selection rationales on
 221 every task. It often improves NBSF, especially when an early high-fitness mode is enough to make
 222 progress. The failure mode is coverage: the model tends to return candidates near regions it has
 223 already identified rather than spreading measurements across distinct high-fitness regions. This
 224 behavior is consistent with prior reports that instruction-tuned LLMs can behave like strong priors
 225 but poor calibrated surrogates under small experimental budgets [80, 101, 90]. In this role, a learned
 226 surrogate provides a more reliable decision step than in-context selection alone, though the strongest
 227 surrogate is task-dependent.

228 3.4 Comparison to Steered Generation

229 **Experimental setup.** In this paradigm, we compare against Steered Generation for Protein Optimiza-
 230 tion (SGPO) since the authors conduct a broad sweep over generative priors and steering strategies,
 231 benchmarked on protein fitness optimization [102].

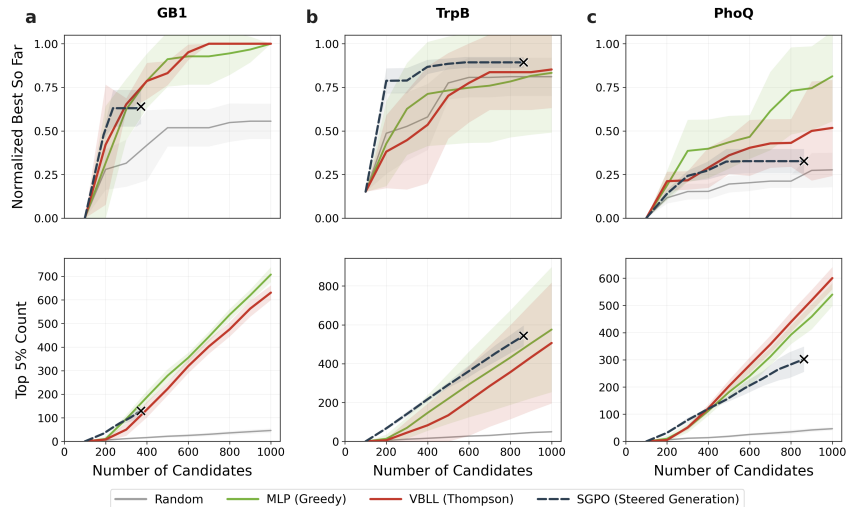


Figure 5: **SciOpt vs. Steered Generative Modeling.** Columns show three combinatorially complete protein landscapes GB1, TrpB, and PhoQ. Top row: NBSF. Bottom row: Top 5% count. SciOpt is evaluated with MLP and VBLL heads on ESM-C embeddings. SGPO samples high fitness candidates via steered generation, as discussed in 3.4. **X** marks the point after which SGPO can no longer propose unique candidate sequences, resulting in diversity collapse

232 We select the strongest reported configuration in which SGPO samples from a masked diffusion
 233 language model (MDLM), $p_{\theta}(x)$, trained on the target-family MSA, and uses decoupled
 234 annealing posterior sampling (DAPS) with an ensemble of one-hot MLP value functions $g_{\phi,t}$
 235 trained on \mathcal{D}_{t-1} . Informally, this steers generation toward a reward-tilted distribution of the form
 236 $p_t(x) \propto p_{\theta}(x) \exp(\beta g_{\phi,t}(x))$, with each generated sample guided by a Thompson-style draw from
 237 the ensemble surrogate.

238 The surrogate-oracle used in SGPO can guide sampling toward sequences that have not been experi-
 239 mentally measured. To avoid this, we benchmark on three representative combinatorially complete
 240 protein landscapes with full experimental fitness maps: GB1 IgG-Fc binding, TrpB enzymatic activity,
 241 and PhoQ-PhoP interface signaling [100, 44, 65]. This removes the learned-oracle scoring confound
 242 in the original SGPO benchmark where every valid proposal from either method is scored against the
 243 same measured ground truth.

244 We use the same batch configuration they use in their paper (batch size 100 across 10 total rounds).
 245 Round 0 initialization and the seeds are equivalent for both methods, in accordance with the per task
 246 configuration in this paper. The results are summarized in Figure 5.

247 **Results on combinatorially complete protein landscapes.**

248 SciOpt evaluated with MLP and VBLL heads on ESM-C embeddings match or exceed SGPO on
 249 Top-5% Count and NBSF on GB1 and PhoQ, where the gap widens with cumulative data because
 250 SGPO repeatedly exhausts its supply of unique proposals. On TrpB, SGPO retains a Top-5% lead
 251 throughout its active phase, and SciOpt MLP draws roughly even by the budget cap.

252 The PhoQ MSA is the shallowest of the three targets, which limits the SGPO prior coverage of
 253 high-fitness regions and is consistent with SGPO performing only marginally better than random
 254 on NBSF. SciOpt VBLL shows wider seed-level variance than the MLP variant due to a subset of
 255 seeds failing to escape early local optima. SGPO retains a meaningful early-batch advantage on TrpB,
 256 where it has the deepest MSA of the three targets. A natural extension is to benchmark against an
 257 SGPO variant whose MDLM is replaced by a pre-trained masked diffusion language model, such as
 258 DPLM, which the authors hypothesize may serve as a stronger prior than their per-target MDLM.
 259 [97, 102]

260 Although SGPO is configured for a batch of 100 candidates per round, diversity collapse is universal
 261 across the three protein tasks, notably most acutely with the GB1 dataset, which reflects a known
 262 issue in generative sampling [6].

263 4 Discussion

264 **Limitations.** The main limitation is that our evaluation is retrospective. All campaigns optimize over
265 fixed candidate pools with known experimental labels. This design enables controlled comparison
266 across methods, but it abstracts away experimental challenges. SciOpt also assumes that a candidate
267 pool already exists. It selects from this pool rather than generating new proteins, molecules, RNA
268 sequences, or regulatory elements, so its performance depends on the quality, diversity, and coverage
269 of the upstream candidate generator. Several modeling choices further bound the scope of the
270 study. We use a below-median initialization protocol, which makes the starting library deliberately
271 challenging but does not fully separate the effect of initialization quality from surrogate quality.
272 We use one primary foundation model per modality in the main sweep, with broader backbone
273 comparisons left to the appendix E. Foundation model and LLM data contamination remains a
274 concern when benchmark candidates overlap with pretraining corpora, even when measured labels
275 are absent. Finally, the benchmark is single-objective, while real campaigns often balance potency,
276 selectivity, stability, synthesis, delivery, cost, and other constraints. The comparisons to adjacent
277 AI-for-science paradigms should also be interpreted narrowly. The LLM experiment evaluates direct
278 in-context candidate selection from a visible pool, not frontier agentic systems with retrieval, tool
279 use, multi-step planning, or external simulators. The SGPO comparison evaluates a representative
280 steered generative workflow on complete protein landscapes, but does not rule out stronger priors
281 or improved diversity control. Furthermore, mutations on combinatorial protein landscapes were
282 limited to 4 positions and does not reflect protein engineering settings where biochemical knowledge
283 of the target identifies is highly limited without known a tractable set of plausible positions such as
284 active-site residues, contact maps, or prior deep mutational scans.

285 **Practical takeaways.** Despite these limitations, SciOpt provides a simple recipe for running Bayesian
286 optimization over scientific candidate pools, requiring only three common components across domains.
287 Neural surrogates paired with posterior sampling, especially VBLL and MLP ensembles, occupied
288 the leading Pareto frontier. However, notably, a reinitialized greedy MLP emerged as a strong simple
289 baseline, suggesting that stochastic training can provide useful implicit exploration even without an
290 explicit uncertainty model.

291 The batch size scaling experiments showed, at fixed total budget, smaller batches incorporate feedback
292 more frequently, reaching strong NBSF values earlier and recovering more Top-5% candidates. This
293 suggests that for sequential experimental design, there exists a tradeoff between model learnability and
294 per-round scale. However, fortunately, the default well-plate size of 96 showed strong convergence in
295 both peak candidate identification and elite coverage.

296 **Future directions.** The most important next step is validation in closed wet-lab loops. Live
297 experimental campaigns would expose the operational constraints that fixed-pool benchmarks cannot
298 capture, and would test whether the batch-size scaling trends remain useful as practical planning
299 curves. A related direction is multi-objective, constrained, and multi-fidelity BO, since practical
300 discovery campaigns rarely optimize a single scalar endpoint. Extending SciOpt to reason over
301 multiple assay readouts, heterogeneous measurement costs, and proxy-to-validation assay hierarchies
302 would make the framework closer to real campaign decisions.

303 In summary, SciOpt unifies Bayesian optimization across proteins, DNA, RNA, and small molecules,
304 demonstrating that frozen scientific representations, when paired with lightweight neural surrogates
305 and posterior-sampling acquisitions, is a practical approach to sequential experimental design that is
306 competitive with alternative AI for Science paradigms.

307 More broadly, our findings point toward a complementary view of AI-for-science. The SGPO
308 generative baseline can propose strong candidates early, but our complete-landscape evaluation also
309 shows that its supply of unique proposals can mode collapse under optimization pressure. This
310 motivates closed-loop systems in which generative models expand the candidate set, while BO
311 surrogates and acquisition functions guide sampling toward high-value and informative regions.
312 Foundation-model representations could provide the shared feature space for this guidance, while
313 acquisition function guidance could promote diverse sampling via balancing the explore-exploit
314 tradeoff. Complementarily, LLMs may be better positioned at a higher level of abstraction: retrieving
315 and synthesizing biochemical knowledge from the literature, imposing mechanism constraints on
316 combinatorial search spaces, selecting among modular generators, surrogates, and acquisition rules,
317 and automating the workflow that connects design, measurement, and model updating.

318 **References**

- 319 [1] Sebastian Ament, Samuel Daulton, David Eriksson, Maximilian Balandat, and Eytan Bakshy.
320 Unexpected improvements to expected improvement for Bayesian optimization. In *Advances*
321 *in Neural Information Processing Systems (NeurIPS)*, 2023. arXiv:2310.20708.
- 322 [2] Johan O. L. Andreasson, Andrew Savinov, Steven M. Block, and William J. Greenleaf.
323 Comprehensive sequence-to-function mapping of cofactor-dependent RNA catalysis in the
324 glmS ribozyme. *Nature Communications*, 11:1663, 2020.
- 325 [3] Rohit Arora, Murphy Angelo, Christian Andrew Choe, Courtney A. Shearer, Aaron W. Kol-
326 lasch, Fiona Qu, Ruben Weitzman, Artem Gazizov, Sarah Gurev, Erik Xie, Debora S. Marks,
327 and Pascal Notin. RNAGym: Large-scale benchmarks for RNA fitness and structure prediction.
328 *bioRxiv preprint*, 2025. Also ICLR 2025 GEM Workshop.
- 329 [4] Maximilian Balandat, Brian Karrer, Daniel R. Jiang, Samuel Daulton, Benjamin Letham,
330 Andrew Gordon Wilson, and Eytan Bakshy. BoTorch: A framework for efficient Monte-Carlo
331 Bayesian optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*,
332 2020. arXiv:1910.06403.
- 333 [5] Lucía Barbadilla-Martínez, Niek Klaassen, Víctor H. Franceschini-Santos, Jeremie Breda,
334 Hacı Yücel, Miguel Hernández-Quiles, Tijs van Lieshout, Carlos G. Urzua Traslaviña, M. C.
335 Luong Boi, Mohammad Akbarzadeh, C. Hermana-Garcia-Agullo, Sebastiano Gregoricchio,
336 Marcel de Haas, Roel Straver, Sarah Derks, Wilbert Zwart, Emile Voest, Lude Franke, Michiel
337 Vermeulen, Jeroen de Ridder, and Bas van Steensel. Regulatory grammar in human promoters
338 uncovered by MPRA-based deep learning. *Nature*, 651(8107):1107–1116, 2026.
- 339 [6] Roberto Barceló, Cristóbal Alcázar, and Felipe Tobar. Avoiding mode collapse in diffusion
340 models fine-tuned with reinforcement learning. arXiv:2410.08315, 2024. Hierarchical Reward
341 Fine-tuning (HRF); preserves sample diversity when fine-tuning diffusion models with RL
342 rewards.
- 343 [7] Sam Boshar, Benjamin Evans, Ziqi Tang, Armand Picard, Yanis Adel, Franziska K. Lorbeer,
344 Chandana Rajesh, Tristan Karch, Shawn Sidbon, David Emms, Javier Mendoza-Revilla,
345 Fatimah Al-Ani, Evan Seitz, Yair Schiff, Yohan Bornachot, Ariana Hernandez, Marie
346 Lopez, Alexandre Laterre, Karim Beguir, Peter Koo, Volodymyr Kuleshov, Alexander Stark,
347 Bernardo P. de Almeida, and Thomas Pierrot. A foundational model for joint sequence-function
348 multi-species modeling at scale for long-range genomic prediction. *bioRxiv*, 2025.
- 349 [8] Lisa Brenan, Aleksandr Andreev, Ofir Cohen, Sasha Pantel, Atanas Kamburov, Davide Cac-
350 chiarelli, Nicole S. Persky, Cong Zhu, Mukta Bagul, Eva M. Goetz, Alex B. Burgin, Levi A.
351 Garraway, Gad Getz, Tarjei S. Mikkelsen, Federica Piccioni, David E. Root, and Cory M.
352 Johannessen. Phenotypic characterization of a comprehensive set of MAPK1/ERK2 missense
353 mutants. *Cell Reports*, 17(4):1171–1183, 2016.
- 354 [9] Garyk Brix, Matthew G. Durrant, Jerome Ku, Michael Poli, Greg Brockman, Daniel Chang,
355 Gabriel A. Gonzalez, Samuel H. King, David B. Li, Aditi T. Merchant, Mohsen Naghipourfar,
356 Eric Nguyen, Chiara Ricci-Tam, David W. Romero, Gwanggyu Sun, Ali Taghibakshi, Anton
357 Vorontsov, Brandon Yang, Myra Deng, Liv Gorton, Nam Nguyen, Nicholas K. Wang, Etowah
358 Adams, Stephen A. Baccus, Steven Dillmann, Stefano Ermon, Daniel Guo, Rajesh Ilango,
359 Ken Janik, Amy X. Li, Emily Mackey, Tyler A. Owens, Aditi Patwardhan, Ganesh Perez,
360 Ivan Skopelitis, Aditi Sen, Ashwin Soren, Brandon H. Yim, Andrei Zoubkov, Eric D. Park,
361 Patrick D. Hsu, and Brian L. Hie. Genome modelling and design across all domains of life
362 with Evo 2. *Nature*, 652:1349–1361, 2026.
- 363 [10] Paul Brunzema, Mikkel Jordahn, John Willes, Sebastian Trimpe, Jasper Snoek, and James
364 Harrison. Bayesian optimization via continual variational last layer training. *International*
365 *Conference on Learning Representations (ICLR)*, 2025.
- 366 [11] Drew H. Bryant, Ali Bashir, Sam Sinai, Nina K. Jain, Pierce J. Ogden, Patrick F. Riley,
367 George M. Church, Lucy J. Colwell, and Eric D. Kelsic. Deep diversification of an AAV
368 capsid protein by machine learning. *Nature Biotechnology*, 39:691–696, 2021.

- 369 [12] Anton Bushuiev, Roman Bushuiev, Olga Pimenova, Nikola Zadorozhny, Raman Samusevich,
370 Elisabet Manaskova, Rachel Seongeun Kim, Hannes Stärk, Jiri Sedlar, Martin Steinegger,
371 Tomáš Pluskal, and Josef Sivic. One protein is all you need. In *International Conference on*
372 *Learning Representations (ICLR)*, 2026. ProteinTTT; arXiv:2411.02109.
- 373 [13] Feiyang Cai, Katelin Zacour, Tianyu Zhu, Tzuen-Rong Tzeng, Yongping Duan, Ling Liu,
374 Srikanth Pilla, Gang Li, and Feng Luo. ChemFM as a scaling law guided foundation model
375 pre-trained on informative chemicals. *Communications Chemistry*, 9:3, 2026.
- 376 [14] Chih-Yu Chang, Milad Azvar, Chinedum Okwudire, and Raed Al Kontar. LLINBO: Trustwor-
377 thy LLM-in-the-loop Bayesian optimization. arXiv:2505.14756, 2025. arXiv preprint.
- 378 [15] Jiayang Chen, Zhihang Hu, Siqi Sun, Qingxiong Tan, Yixuan Wang, Qinze Yu, Licheng
379 Zong, Liang Hong, Jin Xiao, Tao Shen, et al. Interpretable rna foundation model from
380 unannotated data for highly accurate rna structure and function predictions. *arXiv preprint*
381 *arXiv:2204.00300*, 2022.
- 382 [16] Hugo Dalla-Torre, Liam Gonzalez, Javier Mendoza-Revilla, Nicolas Lopez Carranza,
383 Adam Henryk Grzywaczewski, Francesco Oteri, Christian Dallago, Evan Trop, Bernardo P.
384 de Almeida, Hassan Sirelkhatim, Guillaume Richard, Marcin Skwark, Karim Beguir, Marie
385 Lopez, and Thomas Pierrot. Nucleotide Transformer: Building and evaluating robust founda-
386 tion models for human genomics. *Nature Methods*, 22:287–297, 2025.
- 387 [17] Christian Dallago, Jody Mou, Kadina E. Johnston, Bruce J. Wittmann, Nicholas Bhattacharya,
388 Samuel Goldman, Ali Madani, and Kevin K. Yang. FLIP: Benchmark tasks in fitness landscape
389 inference for proteins. *bioRxiv preprint*, 2021.
- 390 [18] Bernardo P. de Almeida, Franziska Reiter, Michaela Pagani, and Alexander Stark. DeepSTARR
391 predicts enhancer activity from DNA sequence and enables the de novo design of synthetic
392 enhancers. *Nature Genetics*, 54(5):613–624, 2022.
- 393 [19] Thomas Desautels, Andreas Krause, and Joel W. Burdick. Parallelizing exploration-
394 exploitation tradeoffs in Gaussian process bandit optimization. *Journal of Machine Learning*
395 *Research*, 15:3873–3923, 2014.
- 396 [20] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training
397 of deep bidirectional transformers for language understanding. In *Proceedings of the 2019*
398 *Conference of the North American Chapter of the Association for Computational Linguistics:*
399 *Human Language Technologies (NAACL-HLT)*, pages 4171–4186, 2019.
- 400 [21] Kieran Didi, Sarah Alamdari, Alex X. Lu, Bruce J. Wittmann, Kadina E. Johnston, Ava P.
401 Amini, Ali Madani, Maya Czeneszew, Christian Dallago, and Kevin K. Yang. FLIP2: Ex-
402 panding protein fitness landscape benchmarks for real-world machine learning applications.
403 *bioRxiv*, 2026.
- 404 [22] Júlia Domingo, Guillaume Diss, and Ben Lehner. Pairwise and higher-order genetic interac-
405 tions during the evolution of a tRNA. *Nature*, 558(7708):117–121, 2018.
- 406 [23] Michael B. Doud and Jesse D. Bloom. Accurate measurement of the effects of all amino-acid
407 mutations on influenza hemagglutinin. *Viruses*, 8(6):155, 2016.
- 408 [24] EvolutionaryScale Team. ESM Cambrian (ESM-C): A new family of efficient protein language
409 models. EvolutionaryScale, 2024.
- 410 [25] Maximilian Gantz, Simon V. Mathis, Friederike E. H. Nintzel, Maximilian Penner, Paul J.
411 Zurek, Tanja Knaus, Vasiliki Tseliou, Elie Patel, Daniel Boros, Friedrich-Maximilian We-
412 berling, Matthew R. A. Kenneth, Oskar J. Klein, Elliot J. Medcalf, Jacob Moss, Michael
413 Herger, Tomasz S. Kaminski, Francesco G. Mutti, Pietro Lió, and Florian Hollfelder. Micro-
414 droplet screening rapidly profiles a biocatalyst to enable its AI-assisted engineering. *bioRxiv*,
415 2024. Source paper for the FLIP2 imine reductase (IRED) DMS dataset; accepted for journal
416 publication 2025; published-version DOI not yet indexed as of 2026-05.

- 417 [26] Miguel García-Ortegón, Gregor N. C. Simm, Austin J. Tripp, José Miguel Hernández-Lobato,
418 Andreas Bender, and Sergio Bacallado. DOCKSTRING: Easy molecular docking yields better
419 benchmarks for ligand design. *Journal of Chemical Information and Modeling*, 62(15):3486–
420 3502, 2022.
- 421 [27] Andrew O. Giacomelli, Xiaoping Yang, Robert E. Lintner, James M. McFarland, Marc Duby,
422 Jaegil Kim, Thomas P. Howard, David Y. Takeda, Seav Huong Ly, Eejung Kim, Hugh S.
423 Gannon, Brian Hurhula, Ted Sharpe, Amy Goodale, Briana Fritchman, Scott Steelman,
424 Francisca Vazquez, Aviad Tsherniak, Andrew J. Aguirre, John G. Doench, Federica Piccioni,
425 Charles W. M. Roberts, Matthew Meyerson, Gad Getz, Cory M. Johannessen, David E. Root,
426 and William C. Hahn. Mutational processes shape the landscape of TP53 mutations in human
427 cancer. *Nature Genetics*, 50(10):1381–1387, 2018.
- 428 [28] David Ginsbourger, Rodolphe Le Riche, and Laurent Carraro. Kriging is well-suited to paral-
429 lelize optimization. In Yoel Tenne and Chi-Keong Goh, editors, *Computational Intelligence in*
430 *Expensive Optimization Problems*, volume 2 of *Adaptation Learning and Optimization*, pages
431 131–162. Springer, 2010. Earlier 2008 HAL working paper of the same title; cite key retains
432 “2008” for continuity with prior drafts.
- 433 [29] Javier González, Zhenwen Dai, Philipp Hennig, and Neil D. Lawrence. Batch Bayesian
434 optimization via local penalization. In *Artificial Intelligence and Statistics (AISTATS)*, 2016.
- 435 [30] Miguel González-Duque, Richard Michael, Simon Bartels, Yevgen Zainchkovskyy, Søren
436 Hauberg, and Wouter Boomsma. A survey and benchmark of high-dimensional Bayesian
437 optimization of discrete sequences. In *Advances in Neural Information Processing Systems*
438 *(NeurIPS) Datasets and Benchmarks Track*, volume 37, 2024.
- 439 [31] Sager J. Gosai, Rodrigo I. Castro, Natalia Fuentes, John C. Butts, Kousuke Mouri, Michael
440 Alasoadura, Susan Kales, Thanh Thanh L. Nguyen, Ramil R. Noche, Arya S. Rao, Mary T.
441 Joy, Pardis C. Sabeti, Steven K. Reilly, and Ryan Tewhey. Machine-guided design of cell-type-
442 targeting cis-regulatory elements. *Nature*, 634(8036):1211–1220, 2024.
- 443 [32] Rushil Gupta, Jason Hartford, and Bang Liu. LLMs for Bayesian optimization in scientific
444 domains: Are we there yet? In *Findings of the Association for Computational Linguistics:*
445 *EMNLP 2025*, pages 15482–15510. Association for Computational Linguistics, 2025.
- 446 [33] Michael P. Guy, David L. Young, Matthew J. Payea, Xiaojun Zhang, Yoshiko Kon, Kimberly M.
447 Dean, Elizabeth J. Grayhack, David H. Mathews, Stanley Fields, and Eric M. Phizicky.
448 Identification of the determinants of tRNA function and susceptibility to rapid tRNA decay by
449 high-throughput in vivo analysis. *Genes & Development*, 28(15):1721–1732, 2014.
- 450 [34] Johannes Hachmann, Roberto Olivares-Amaya, Sule Atahan-Evrenk, Carlos Amador-Bedolla,
451 Roel S. Sánchez-Carrera, Aryeh Gold-Parker, Leslie Vogt, Anna M. Brockway, and Alán
452 Aspuru-Guzik. The Harvard Clean Energy Project: Large-scale computational screening and
453 design of organic photovoltaics on the World Community Grid. *The Journal of Physical*
454 *Chemistry Letters*, 2(17):2241–2251, 2011.
- 455 [35] Hugh K. Haddox, Adam S. Dingens, Sarah K. Hilton, Julie Overbaugh, and Jesse D. Bloom.
456 Mapping mutational effects along the evolutionary landscape of HIV envelope. *eLife*, 7:e34420,
457 2018.
- 458 [36] James Harrison, John Willes, and Jasper Snoek. Variational Bayesian last layers. In *Interna-*
459 *tional Conference on Learning Representations (ICLR)*, 2024.
- 460 [37] Erik Hartman, Jonas Wallin, Johan Malmström, and Jimmy Olsson. Controllable protein
461 design with particle-based Feynman-Kac steering. *arXiv preprint arXiv:2511.09216*, 2025.
- 462 [38] José Miguel Hernández-Lobato, James Requeima, Edward O. Pyzer-Knapp, and Alán Aspuru-
463 Guzik. Parallel and distributed Thompson sampling for large-scale accelerated exploration of
464 chemical space. In *International Conference on Machine Learning (ICML)*, 2017. Cite key
465 inherited from earlier draft uses “PES” suffix; this entry is the 2017 parallel-TS paper that
466 matches the GP-Thompson context in App. C.1, not the 2014 Predictive Entropy Search paper.
467 Rename the key if you want a clearer label.

- 468 [39] Brian L. Hie, Varun R. Shanker, Duo Xu, Theodora U. J. Bruun, Payton A. Weidenbacher,
469 Shaogeng Tang, Wesley Wu, John E. Pak, and Peter S. Kim. Efficient evolution of human
470 antibodies from general protein language models. *Nature Biotechnology*, 42(2):275–283, 2024.
- 471 [40] Tomoyuki Hino, Satoshi N. Omura, Ryoya Nakagawa, Tomoaki Togashi, Satoshi N. Takeda,
472 Takafumi Hiramoto, Satoru Tasaka, Hisato Hirano, Takeshi Tokuyama, Hideki Uosaki, Soh
473 Ishiguro, Madina Kagieva, Hiroto Yamano, Yuto Ozaki, Daisuke Motooka, Hideto Mori, Yuhei
474 Kirita, Yoshiaki Kise, Yuzuru Itoh, Shogo Matoba, Hiroyuki Aburatani, Nozomu Yachie,
475 Tautvydas Karvelis, Virginijus Siksnys, Tsukasa Ohmori, Akihiro Hoshino, and Osamu Nureki.
476 An AsCas12f-based compact genome-editing tool derived by deep mutational scanning and
477 structural analysis. *Cell*, 186(22):4920–4935.e23, 2023.
- 478 [41] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang
479 Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *Iclr*, 1(2):3,
480 2022.
- 481 [42] Anna Jarzab, Nils Kurzawa, Thomas Hopf, Matthias Moerch, Jana Zecha, Niels Leijten,
482 Yangyang Bian, Eva Musiol, Melanie Maschberger, Gabriele Stoehr, Isabelle Becher, Charlotte
483 Daly, Patroklos Samaras, Julia Mergner, Britta Spanier, Angel Angelov, Thilo Werner, Marcus
484 Bantscheff, Mathias Wilhelm, Martin Klingenspor, Simone Lemeer, Wolfgang Liebl, Hannes
485 Hahne, Mikhail M. Savitski, and Bernhard Kuster. Meltome atlas—thermal proteome stability
486 across the tree of life. *Nature Methods*, 17(5):495–503, 2020.
- 487 [43] Kaiyi Jiang, Zhaoqing Yan, Matteo Di Bernardo, Samantha R. Sgrizzi, Lukas Villiger, Alisan
488 Kayabolen, Byungji Kim, Josephine K. Carscadden, Masahiro Hiraizumi, Hiroshi Nishimasu,
489 Jonathan S. Gootenberg, and Omar O. Abudayyeh. Rapid in silico directed evolution by a
490 protein language model with EVOLVEpro. *Science*, 387(6731):eadr6006, 2024.
- 491 [44] Kadina E. Johnston, Patrick J. Almhjell, Ella J. Watkins-Dulaney, Grace Liu, Nicholas J. Porter,
492 Jason Yang, and Frances H. Arnold. A combinatorially complete epistatic fitness landscape in
493 an enzyme active site. *Proceedings of the National Academy of Sciences*, 121(32):e2400439121,
494 2024.
- 495 [45] Eric M. Jones, Nathan B. Lubock, A. J. Venkatakrishnan, Jeffrey Wang, Alex M. Tseng,
496 Joseph M. Paggi, Naomi R. Latorraca, Daniel Cancilla, Megan Satyadi, Jessica E. Davis,
497 M. Madan Babu, Ron O. Dror, and Sriram Kosuri. Structural and functional characterization
498 of G protein-coupled receptors with deep mutational scanning. *eLife*, 9:e54895, 2020.
- 499 [46] Adam Tauman Kalai, Ofir Nachum, Santosh S. Vempala, and Edwin Zhang. Why language
500 models hallucinate. *arXiv preprint arXiv:2509.04664*, 2025.
- 501 [47] Shengdong Ke, Vincent Anquetil, Jorge Rojas Zamalloa, Arpit Maity, Anthony Yang, Mauri-
502 cicio A. Arias, Sergey Kalachikov, James J. Russo, Jingyue Ju, and Lawrence A. Chasin.
503 Saturation mutagenesis reveals manifold determinants of exon definition. *Genome Research*,
504 28(1):11–24, 2018.
- 505 [48] Leo Klarner, Tim G. J. Rudner, Garrett M. Morris, Charlotte M. Deane, and Yee Whye Teh.
506 Context-guided diffusion for out-of-distribution molecular and protein design. In *International
507 Conference on Machine Learning (ICML)*, 2024. arXiv:2407.11942.
- 508 [49] Shungo Kobori, Kazuki Takahashi, and Yohei Yokobayashi. Deep sequencing analysis of
509 aptazyme variants based on a pistol ribozyme. *ACS Synthetic Biology*, 6(7):1283–1288, 2017.
- 510 [50] Shungo Kobori and Yohei Yokobayashi. High-throughput mutational analysis of a twister
511 ribozyme. *Angewandte Chemie International Edition*, 55(35):10354–10357, 2016.
- 512 [51] Agustinus Kristiadi, Felix Strieth-Kalthoff, Marta Skreta, Pascal Poupart, Alán Aspuru-Guzik,
513 and Geoff Pleiss. A sober look at LLMs for material discovery: Are they actually good for
514 Bayesian optimization over molecules? In *International Conference on Machine Learning
515 (ICML)*, 2024.
- 516 [52] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable
517 predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information
518 Processing Systems (NeurIPS)*, 2017.

- 519 [53] Juhye M. Lee, John Huddleston, Michael B. Doud, Kathryn A. Hooper, Nicholas C. Wu,
520 Trevor Bedford, and Jesse D. Bloom. Deep mutational scanning of hemagglutinin helps predict
521 evolutionary fates of human H3N2 influenza variants. *Proceedings of the National Academy*
522 *of Sciences*, 115(35):E8276–E8285, 2018.
- 523 [54] Chuan Li, Wenfeng Qian, Calum J. Maclean, and Jianzhi Zhang. The fitness landscape of a
524 tRNA gene. *Science*, 352(6287):837–840, 2016.
- 525 [55] Tennison Liu, Nicolás Astorga, Nabeel Seedat, and Mihaela van der Schaar. Large lan-
526 guage models to enhance Bayesian optimization. In *International Conference on Learning*
527 *Representations (ICLR)*, 2024. LLAMBO.
- 528 [56] Dougal Maclaurin, David Duvenaud, and Ryan P Adams. Early stopping is nonparametric
529 variational inference. *arXiv preprint arXiv:1504.01344*, 2015.
- 530 [57] Ewan K. S. McRae, Christopher J. K. Wan, Emil L. Kristoffersen, Kasper Hansen, Emanuele
531 Gianni, Isabel Gallego, Joseph F. Curran, James Attwater, Philipp Holliger, and Ebbe S.
532 Andersen. Cryo-EM structure and functional landscape of an RNA polymerase ribozyme.
533 *Proceedings of the National Academy of Sciences*, 121(3):e2313332121, 2024.
- 534 [58] Joshua Meier, Roshan Rao, Robert Verkuil, Jason Liu, Tom Sercu, and Alexander Rives.
535 Language models enable zero-shot prediction of the effects of mutations on protein function.
536 In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. ESM-1v; demon-
537 strates that protein language models capture mutation effects zero-shot and that fine-tuning on
538 sequences from the target protein family further improves variant-effect prediction.
- 539 [59] Aliakbar Nafar, Kristen Brent Venable, and Parisa Kordjamshidi. Learning vs retrieval: The
540 role of in-context examples in regression with large language models. In *Proceedings of*
541 *the 2025 Conference of the North American Chapter of the Association for Computational*
542 *Linguistics: Human Language Technologies (NAACL)*, 2025. arXiv:2409.04318 (2024).
- 543 [60] Giang Ngo, Dat Phan Trong, Dang Nguyen, Sunil Gupta, and Svetha Venkatesh. Adaptive
544 acquisition selection for bayesian optimization with large language models. *arXiv preprint*
545 *arXiv:2602.07904*, 2026.
- 546 [61] Pascal Notin, Aaron W. Kollasch, Daniel Ritter, Lood van Niekerk, Steffanie Paul, Hansen
547 Spinner, Nathan Rollins, Ada Shaw, Rose Orenbuch, Ruben Weitzman, Jonathan Frazer,
548 Mafalda Dias, Dinko Franceschi, Yarin Gal, and Debora S. Marks. ProteinGym: Large-scale
549 benchmarks for protein fitness prediction and design. In *Advances in Neural Information*
550 *Processing Systems (NeurIPS) Datasets and Benchmarks*, 2023.
- 551 [62] Sebastian W. Ober, Carl E. Rasmussen, and Mark van der Wilk. The promises and pitfalls
552 of deep kernel learning. In *Proceedings of the 37th Conference on Uncertainty in Artificial*
553 *Intelligence (UAI)*, 2021.
- 554 [63] Rafael Josip Penić, Tin Vlašić, Roland G. Huber, Yue Wan, and Mile Šikić. RiNALMo:
555 General-purpose RNA language models can generalize well on structure prediction tasks.
556 *Nature Communications*, 16:5671, 2025.
- 557 [64] Gianluca Peri, Clemence Gibard, Nicholas H. Shults, Kelvin Crossin, and Eric J. Hayden.
558 Dynamic RNA fitness landscapes of a group I ribozyme during changes to the experimental
559 environment. *Molecular Biology and Evolution*, 39(3):msab373, 2022.
- 560 [65] Anna I. Podgornaia and Michael T. Laub. Pervasive degeneracy and epistasis in a protein-
561 protein interface. *Science*, 347(6222):673–677, 2015.
- 562 [66] Victoria O. Pokusaeva, Dinara R. Usmanova, Ekaterina V. Putintseva, Lorena Espinar, Karen S.
563 Sarkisyan, Alexander S. Mishin, Natalya S. Bogatyreva, Dmitry N. Ivankov, Arseniy V.
564 Akopyan, Sergey Ya. Avvakumov, Inna S. Povolotskaya, Guillaume J. Filion, Lucas B. Carey,
565 and Fyodor A. Kondrashov. An experimental assay of the interactions of amino acids from
566 orthologous sequences shaping a complex fitness landscape. *PLOS Genetics*, 15(4):e1008079,
567 2019.

- 568 [67] Michael Poli, Stefano Massaroli, Eric Nguyen, Daniel Y. Fu, Tri Dao, Stephen Baccus, Yoshua
569 Bengio, Stefano Ermon, and Christopher Ré. Hyena hierarchy: Towards larger convolutional
570 language models. In *International Conference on Machine Learning (ICML)*, 2023.
- 571 [68] Robert Pollice, Gabriel dos Passos Gomes, Matteo Aldeghi, Riley J. Hickman, Mario Krenn,
572 Cyrille Lavigne, Michael Lindner-D’Addario, AkshatKumar Nigam, Cher Tian Ser, Zhenpeng
573 Yao, and Alán Aspuru-Guzik. Data-driven strategies for accelerated materials design. *Accounts*
574 *of Chemical Research*, 54(4):849–860, 2021.
- 575 [69] Qwen Team. Qwen3.6-35B-A3B: Agentic coding power, now open to all, April 2026.
- 576 [70] Abdul Muntakim Rafi, Daria Nogina, Dmitry Penzar, Dohoon Lee, Danyeong Lee, Nayeon
577 Kim, et al. A community effort to optimize sequence-based deep learning models of gene
578 regulation. *Nature Biotechnology*, 43:412–423, 2025.
- 579 [71] Mayk Caldas Ramos, Shane S. Michtavy, Marc D. Porosoff, and Andrew D. White. Bayesian
580 optimization of catalysis with in-context learning. *ACS Central Science*, 2026. Articles ASAP,
581 online April 2026; preprint arXiv:2304.05341 (2023).
- 582 [72] Bojana Ranković, Ryan-Rhys Griffiths, and Philippe Schwaller. Large language models as
583 uncertainty-calibrated optimizers for experimental discovery. *arXiv preprint arXiv:2504.06265*,
584 2025.
- 585 [73] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine*
586 *Learning*. MIT Press, 2006.
- 587 [74] Jarrid Rector-Brooks, Théophile Lambert, Marta Skreta, Daniel Roth, Yueming Long, Zi-Qi
588 Li, Xi Zhang, Miruna Cretu, Francesca-Zhoufan Li, Tanvi Ganapathy, Emily Jin, Avishek Joey
589 Bose, Jason Yang, Kirill Neklyudov, Yoshua Bengio, Alexander Tong, Frances H. Arnold,
590 and Cheng-Hao Liu. General multimodal protein design enables DNA-encoding of chemistry.
591 *arXiv preprint arXiv:2604.05181*, 2026. DISCO.
- 592 [75] Carlos Riquelme, George Tucker, and Jasper Snoek. Deep Bayesian bandits showdown: An
593 empirical comparison of Bayesian deep networks for Thompson Sampling. In *International*
594 *Conference on Learning Representations (ICLR)*, 2018.
- 595 [76] James M. Roberts, Jacob D. Beck, Tanner B. Pollock, Devin P. Bendixsen, and Eric J. Hayden.
596 RNA sequence to structure analysis from comprehensive pairwise mutagenesis of multiple
597 self-cleaving ribozymes. *eLife*, 12:e80360, 2023.
- 598 [77] Philip A. Romero, Andreas Krause, and Frances H. Arnold. Navigating the protein fit-
599 ness landscape with Gaussian processes. *Proceedings of the National Academy of Sciences*,
600 110(3):E193–E201, 2013.
- 601 [78] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks
602 for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted*
603 *Intervention (MICCAI)*, pages 234–241, 2015.
- 604 [79] Daniel Russo and Benjamin Van Roy. Learning to optimize via posterior sampling. *Mathemat-*
605 *ics of Operations Research*, 39(4):1221–1243, 2014. arXiv:1301.2609.
- 606 [80] Adam Rychert, Gasper Spagnolo, and Evgenii Posashkov. Reproducibility study of large
607 language model Bayesian optimization. arXiv:2511.18891, 2025. arXiv preprint.
- 608 [81] Julian Salazar, Davis Liang, Toan Q. Nguyen, and Katrin Kirchhoff. Masked language model
609 scoring. In *Proceedings of the 58th Annual Meeting of the Association for Computational*
610 *Linguistics (ACL)*, pages 2699–2712, 2020.
- 611 [82] Benjamin J. Shields, Jason Stevens, Jun Li, Marvin Parasram, Farhan Damani, Jesus I. Mar-
612 tinez Alvarado, Jacob M. Janey, Ryan P. Adams, and Abigail G. Doyle. Bayesian reaction
613 optimization as a tool for chemical synthesis. *Nature*, 590(7844):89–96, 2021.
- 614 [83] Eduardo Soares, Emilio Vital Brazil, Victor Shirasuna, Dmitry Zubarev, Renato Cerqueira,
615 and Kristin Schmidt. An open-source family of large encoder-decoder foundation models for
616 chemistry. *Communications Chemistry*, 8:193, 2025.

- 617 [84] Valerie W. C. Soo, Jacob B. Swadling, Andre J. Faure, and Tobias Warnecke. Fitness landscape
618 of a dynamic RNA structure. *PLOS Genetics*, 17(2):e1009353, 2021.
- 619 [85] Marion Sourisseau, Daniel J. P. Lawrence, Megan C. Schwarz, Carina H. Storrs, Ethan C. Veit,
620 Jesse D. Bloom, and Matthew J. Evans. Deep mutational scanning comprehensively maps how
621 Zika envelope protein mutations affect viral growth and antibody escape. *Journal of Virology*,
622 93(23):e01291–19, 2019.
- 623 [86] Niranjana Srinivas, Andreas Krause, Sham M. Kakade, and Matthias W. Seeger. Gaussian
624 Process optimization in the bandit setting: No regret and experimental design. In *International
625 Conference on Machine Learning (ICML)*, 2010.
- 626 [87] Samuel Stanton, Wesley J. Maddox, Nate Gruver, Phillip Maffettone, Emily Delaney, Peyton
627 Greenside, and Andrew Gordon Wilson. Accelerating Bayesian optimization for biological
628 sequence design with denoising autoencoders. In *International Conference on Machine
629 Learning (ICML)*, 2022. LaMBO.
- 630 [88] Michael A. Stiffler, Doeke R. Hekstra, and Rama Ranganathan. Evolvability as a function of
631 purifying selection in TEM-1 β -lactamase. *Cell*, 160(5):882–892, 2015.
- 632 [89] Jin Su, Chenchen Han, Yuyang Zhou, Junjie Shan, Xibin Zhou, and Fajie Yuan. SaProt:
633 Protein language modeling with structure-aware vocabulary. In *International Conference on
634 Learning Representations (ICLR)*, 2024.
- 635 [90] Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao,
636 Chelsea Finn, and Christopher D. Manning. Just ask for calibration: Strategies for eliciting
637 calibrated confidence scores from language models fine-tuned with human feedback. In
638 *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023.
- 639 [91] Gary Tom, Stefan P. Schmid, Sterling G. Baird, Yang Cao, Kourosh Darvish, Han Hao, Sheng
640 Lo, Sergio Pablo-García, Ella M. Rajaonson, Marta Skreta, Naruki Yoshikawa, Samantha
641 Corapi, Gun Deniz Akkoc, Felix Strieth-Kalthoff, Martin Seifrid, and Alán Aspuru-Guzik.
642 Self-driving laboratories for chemistry and materials science. *Chemical Reviews*, 124(16):9633–
643 9732, 2024.
- 644 [92] Jacob M. Tome, Abdullah Ozer, John M. Pagano, Dan Gheba, Gary P. Schroth, and John T. Lis.
645 Comprehensive analysis of RNA-protein interactions by high-throughput sequencing-RNA
646 affinity profiling. *Nature Methods*, 11(6):683–688, 2014.
- 647 [93] Floris van der Flier, Dave Estell, Sina Pricelius, Lydia Dankmeyer, Sander van Stigt Thans,
648 Harm Mulder, Rei Otsuka, Frits Goedegebuur, Laurens Lammerts, Diego Staphorst, Aalt D. J.
649 van Dijk, Dick de Ridder, and Henning Redestig. Enzyme structure correlates with variant
650 effect predictability. *Computational and Structural Biotechnology Journal*, 23:3489–3497,
651 2024.
- 652 [94] Derek van Tilborg, Alisa Alenicheva, and Francesca Grisoni. Exposing the limitations of
653 molecular machine learning with activity cliffs. *Journal of Chemical Information and Modeling*,
654 62(23):5938–5951, 2022.
- 655 [95] Alex Wang and Kyunghyun Cho. BERT has a mouth, and it must speak: BERT as a Markov
656 random field language model. In *Proceedings of the Workshop on Methods for Optimizing and
657 Evaluating Neural Language Generation (NeuralGen)*, pages 30–36, 2019.
- 658 [96] Chenyu Wang, Masatoshi Uehara, Yichun He, Amy Wang, Tommaso Biancalani, Avantika
659 Lal, Tommi Jaakkola, Sergey Levine, Hanchen Wang, and Aviv Regev. Fine-tuning discrete
660 diffusion models via reward optimization with applications to DNA and protein design. *arXiv
661 preprint arXiv:2410.13643*, 2024. DRAKES.
- 662 [97] Xinyou Wang, Zaixiang Zheng, Fei Ye, Dongyu Xue, Shujian Huang, and Quanquan Gu. Dif-
663 fusion language models are versatile protein learners. In *Proceedings of the 41st International
664 Conference on Machine Learning (ICML)*, volume 235 of *PMLR*, 2024. DPLM; ByteDance
665 Research & Nanjing University; arXiv:2402.18567.

- 666 [98] James T. Wilson, Viacheslav Borovitskiy, Alexander Terenin, Peter Mostowsky, and Marc Peter
667 Deisenroth. Efficiently sampling functions from Gaussian process posteriors. In *International
668 Conference on Machine Learning (ICML)*, 2020. arXiv:2002.09309.
- 669 [99] James T. Wilson, Riccardo Moriconi, Frank Hutter, and Marc Peter Deisenroth. The reparamete-
670 rization trick for acquisition functions. In *NeurIPS Workshop on Bayesian Optimization*, 2017.
671 arXiv:1712.00424; consider the follow-up Wilson et al., NeurIPS 2018, arXiv:1805.10196 for
672 a peer-reviewed batch-BO citation.
- 673 [100] Nicholas C. Wu, Lei Dai, C. Anders Olson, James O. Lloyd-Smith, and Ren Sun. Adaptation
674 in protein fitness landscapes is facilitated by indirect paths. *eLife*, 5:e16965, 2016.
- 675 [101] Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. Can
676 LLMs express their uncertainty? An empirical evaluation of confidence elicitation in LLMs.
677 In *International Conference on Learning Representations (ICLR)*, 2024.
- 678 [102] Jason Yang, Wenda Chu, Daniel Khalil, Raul Astudillo, Bruce J. Wittmann, Frances H.
679 Arnold, and Yisong Yue. Steering generative models with experimental data for protein fitness
680 optimization. *arXiv preprint arXiv:2505.15093*, 2025. SGPO; MSA-trained generative prior +
681 classifier guidance on fitness labels.
- 682 [103] Tatsuya Yoshizawa, Shoichi Ishida, Tomohiro Sato, Masateru Ohta, Teruki Honma, and Kei
683 Terayama. A data-driven generative strategy to avoid reward hacking in multi-objective
684 molecular design. *Nature Communications*, 16, 2025. DyRAMO.
- 685 [104] Zhe Zhang, Xianggui Hong, Peng Xiong, Junfeng Wang, Yaoqi Zhou, and Jian Zhan. Minimal
686 twister sister-like self-cleaving ribozymes in the human genome revealed by deep mutational
687 scanning. *eLife*, 12:RP90254, 2024.
- 688 [105] Zhe Zhang, Peng Xiong, Tongchuan Zhang, Junfeng Wang, Jian Zhan, and Yaoqi Zhou.
689 Accurate inference of the full base-pairing structure of RNA by deep mutational scanning and
690 covariation-induced deviation of activity. *Nucleic Acids Research*, 48(3):1451–1465, 2020.

691 A Foundation models

692 The frozen main sweep fixes one primary foundation model (FM) per modality and holds it constant
 693 across every surrogate head and acquisition rule in App. ???. For each (FM, task) pair we precompute
 694 two quantities offline: a D -dimensional pooled embedding per candidate, and a length-normalized
 695 log-likelihood-style score that we call the model’s pseudo-likelihood (PLL). The BO loop never
 696 instantiates the FM: learned surrogates consume the precomputed embedding, and the Zero-Shot
 697 surrogate consumes the precomputed PLL. This decouples the comparison from FM inference
 698 throughput and lets us swap FMs on a fixed candidate pool without changing anything else.

699 A.1 Models considered

700 For every modality we use one primary FM in the 61-task sweep and pair it against one principal
 701 alternative for the backbone-sensitivity sweep (App. E). The primary FMs are domain-specific, pub-
 702 licly released, and compatible with the raw candidate representation in the corresponding benchmark.
 703 Each alternative varies the choice along the dimension most likely to matter in that modality: a
 704 structure-aware tokenizer for proteins (SaProt 650M vs. ESM-C 600M), a different RNA-encoder
 705 family at larger scale (RiNALMo-giga vs. RNA-FM), an autoregressive decoder vs. an encoder for
 706 DNA (Evo2-1B vs. NTv3-650M), and an encoder/decoder reconstruction model vs. a causal SMILES
 707 decoder for chemistry (SMI-TED vs. ChemFM-1B). Table 2 summarizes the eight models.

Table 2: **Foundation models used in this work.** Primary models are used for the 61-task surrogate sweep; alternatives are used in the paired backbone-sensitivity sweep (App. E). D is the pooled embedding dimension consumed by every learned surrogate. *PLL family* indicates which scoring procedure in §A.3 produces the cached prior score. “WT-marginal” is single-pass marginal scoring under a masked-LM head in the style of [58]; “ K -stripe masked” is a K -pass batched masking scheme used for U-Net-style nucleotide models (§A.3); “AR” is left-to-right next-token log-likelihood; “Recon.” is encoder/decoder reconstruction cross-entropy. Length normalization divides the per-sequence log-probability by the number of scored non-special tokens.

Modality	Model	Role	Architecture	Params	D	PLL family
Protein	ESM-C 600M [24]	primary	encoder, transformer (rotary, SwiGLU)	600M	1152	WT-marginal
Protein	SaProt 650M (AF2) [89]	alternative	encoder, MLM (446-vocab AA \times 3Di bigrams)	650M	1280	WT-marginal, AA-marg.
RNA	RNA-FM [15]	primary	encoder, BERT-style MLM	100M	640	WT-marginal
RNA	RiNALMo-giga [63]	alternative	encoder, transformer (RoPE, SwiGLU)	650M	1280	WT-marginal
DNA	NTv3-650M [7, 16]	primary	nucleotide encoder, U-Net + transformer	650M	1536	K -stripe masked
DNA	Evo2-1B [9]	alternative	decoder, StripedHyena [67]	1B	1920	AR
Chemistry	ChemFM-1B [13]	primary	decoder, Llama-family causal LM	970M	2048	AR
Chemistry	SMI-TED [83]	alternative	encoder/decoder	289M	768	Recon.

708 A.2 Embedding extraction protocol

709 Each FM is run in inference mode using its native tokenizer. Inputs come from the curated task tables
 710 in App. D: protein, RNA, and DNA strings are uppercased before tokenization, while molecules are
 711 kept in their curated SMILES form. Rows with non-finite targets, empty inputs, or inputs longer than
 712 the model context window are dropped from that FM’s candidate pool, and the BO comparisons within
 713 a modality use the intersection pool so that primary and alternative FMs see identical candidates.
 714 Inference uses bfloat16 mixed-precision autocast on GPU.

715 The pooled embedding is the mean of the final-layer hidden states over non-special tokens, where
 716 the special set is taken from each tokenizer’s mask and includes BOS/CLS, EOS/SEP, PAD, MASK,
 717 UNK, and model-specific equivalents. We apply the same mean-over-non-special-tokens rule to every
 718 FM in Table 2 so that pooling is not a confound in the comparison, with two exceptions dictated by
 719 the model itself. For SMI-TED we use the autoencoder bottleneck vector that the decoder is trained
 720 to reconstruct from, since that is the representation the model is designed to expose. For NTv3 we
 721 pool the hidden state at the LM-head input rather than the post-skip output, so that the embedding
 722 stays on the same surface the model uses internally; the corresponding prior score is computed under
 723 the masking scheme below to avoid a trivial-copy path through the U-Net’s skip connection. For
 724 Evo2 we additionally retain an intermediate Hyena-block representation alongside the final-layer

725 pool, because several near-final block outputs are empirically degenerate and pooling at those depths
726 would silently degrade the embedding.

727 **A.3 Pseudo-likelihood and log-likelihood scoring**

728 The cached prior is the model’s length-normalized log-probability of the candidate under its native
729 head, averaged over non-special tokens. For autoregressive decoders this is the standard one-token-
730 shift log-likelihood. For encoder masked-LMs we use the single-pass wild-type-marginal scoring
731 of [58], which approximates the true pseudo-log-likelihood of [95, 81] at one forward pass per
732 candidate and is the standard cheap surrogate for protein-LM zero-shot variant scoring [58, 39]. For
733 SMI-TED we use the model’s native reconstruction objective: the per-token log-probability under the
734 autoencoder decoder, averaged over the non-special positions. We refer to all four scalars uniformly
735 as the PLL, and the Zero-Shot surrogate ranks candidates by this scalar with no further modeling.

736 Two FMs require a non-standard construction. SaProt’s vocabulary is a 446-token product of amino-
737 acid and 3Di structural tokens, so a direct gather would conflate sequence and structure preferences;
738 we instead marginalize the bigram softmax over the 3Di axis within each input amino-acid letter, so
739 the score reflects the model’s distribution over the channel that varies in our candidates. NTV3 is a
740 U-Net encoder [78] whose full-resolution skip connection feeds a transform of the token embeddings
741 into the pre-LM-head representation, making a single unmasked forward dominated by a trivial-copy
742 identity rather than the trunk’s predictions; we instead use a K -pass masked scheme that masks $1/K$
743 of the positions on each pass and reads log-probabilities only at the masked positions, with $K=7$
744 chosen to match the $\approx 15\%$ MLM training rate of [20].

745 **B Surrogate heads**

746 All surrogates in the main sweep consume the same per-task FM embeddings, cached once per (FM,
747 task) pair as a parquet of fixed-size float vectors (Sec. 2, App. A). The only thing that varies across
748 rows of Table 1 is the surrogate head and its acquisition rule. Any gap in NBSF or Top-5% count
749 between two methods isolates the head, not the representation. This appendix records the eight
750 surrogates compared in the main results, the exact hyperparameters used for each, and the ablations
751 we ran when picking those settings.

752 **B.1 Random (Random)**

753 Uniform sampling without replacement from the heldout pool. The configuration omits both the
754 surrogate and the acquisition blocks and sets `batch_strategy=random` on the optimizer. The
755 featurizer is set to `identity` so no FM forward pass is performed.

756 **B.2 Zero-Shot (Zero_Shot)**

757 The zero-shot baseline ranks the heldout pool by the FM’s own pseudo-likelihood at each candidate,
758 with no labelled-data fit and no parameter update over the campaign. This is the natural “FM-prior-
759 only” control: it isolates how much of the optimization signal is already in the FM’s pretraining
760 likelihood, before any task-specific learning takes place. For protein and RNA assays this baseline is
761 well-known to correlate with measured fitness for many DMS landscapes [61, 58], which is what
762 makes it a meaningful anchor rather than a trivial floor.

763 The cached score column is the per-FM PLL constructed by the scoring procedures of App. A.3:
764 WT-marginal scoring for the masked-LM encoders (ESM-C 600M, RNA-FM), K -stripe masked PLL
765 for NTV3-650M, and autoregressive log-likelihood for ChemFM-1B. All three are length-normalised
766 so candidates of different sizes sit on the same scale and higher always means more likely under the
767 FM. Acquisition is `ArgMax` on this scalar. Because the score is fixed at parquet-creation time and
768 the surrogate has no parameters, the ranking of the heldout pool never changes between BO rounds;
769 the loop simply removes each round’s pick from the heldout set and selects the next-highest-scoring
770 candidate. This is the sense in which `zero_shot` is open-loop: measured labels are recorded but
771 ignored.

772 **B.3 Random Forest** (RF_Greedy, RF_LogEI)

773 Sklearn’s RandomForestRegressor with 100 trees and standardized targets, fed the raw FM
774 embedding as a fixed-size feature vector. Tree-level hyperparameters are sklearn’s defaults:
775 `criterion=friedman_mse`, unbounded depth, `min_samples_leaf=1`, and `max_features=1.0`
776 (every split considers all features). The random-state seed is fixed at 42 so tree splits are deterministic
777 across methods and seeds; the only stochasticity in the baseline is the BO loop. We use the predictive
778 mean of the forest (`rf.predict`) as the surrogate mean and the variance of the per-tree predictions
779 as a heteroscedastic uncertainty estimate.

780 This recipe matches the EvolvePro setup [43]: a frozen protein-FM embedding, an unconstrained
781 sklearn random forest with 100 estimators, and a greedy step that picks the candidate with the largest
782 forest mean. We also report the LogEI-on-tree-variance variant in the same row of Table 1, which
783 uses the tree variance as a posterior variance under BoTorch’s LogExpectedImprovement [1] with
784 f^* set to the running maximum on the labelled set. LogEI adds an exploration term but does not
785 change the surrogate; the comparison between the two acquisitions inside this row isolates the effect
786 of variance-aware exploration on top of an EvolvePro-style head.

787 **B.4 Linear** (Linear)

788 A single linear layer on the raw FM embedding, $f(x) = w^\top z + b$ with $z \in \mathbb{R}^D$, trained end-to-end
789 with SGD rather than the closed-form ridge solution. Training is identical to the mlp head described
790 next: AdamW (`lr=3e-4`, `wd=1e-3`), batch size 32, gradient clip 1.0, MSE loss against standardized
791 targets, up to 3000 epochs per BO round with patience-100 early stopping on the per-epoch training
792 loss. The head is reinitialized at every BO round before retraining, so nothing carries between rounds
793 beyond the labelled set. Acquisition is ArgMax.

794 We use this SGD-trained linear head rather than a closed-form ridge solution so the only architectural
795 difference between `linear` and `mlp` is the head depth, with optimizer, loss, schedule, and reinitial-
796 ization protocol all matched. This isolates the effect of the nonlinearity in the MLP from the effect of
797 the optimization recipe.

798 **B.5 MLP** (MLP_Greedy)

799 A 2-layer MLP head on the raw FM embedding, with `hidden_features=128`, `num_layers=1`,
800 `activation=elu`. Training uses AdamW (`lr=3e-4`, `wd=1e-3`), mini-batch 32, gradient clip 1.0,
801 MSE on standardized targets, up to 3000 epochs with patience-100 early stopping on the per-epoch
802 training loss. Acquisition is ArgMax on the predictive mean.

803 The head is reinitialized at every BO round before retraining; the model weights, the optimizer
804 moments, and any learning-rate state are all dropped between rounds. This matters for the result in
805 Sec. 3.1: the converged weights of an SGD-trained MLP from a fresh init can be intuitively read
806 as an approximate draw from the implicit posterior over heads induced by the init distribution and
807 the optimizer trajectory on the current labelled set, so an ArgMax on those weights is effectively a
808 single-sample Thompson rule with the sampler hidden in the optimizer.

809 **B.6 MLP Ensemble** (MLP_Ensemble)

810 Five MLP heads with the same architecture and optimizer recipe as `mlp`, trained jointly on the labelled
811 set with summed MSE loss. The diversity comes from initialization, not from data: each head’s linear
812 weights are Kaiming-uniform initialized under a per-head seed (`torch.manual_seed(k * 1337)` for
813 the k -th head), so the five heads start at different points in weight space and converge to different
814 solutions under the same SGD trajectory. We do not bootstrap or subsample the labelled data. All
815 five heads are reinitialized at the start of every BO round.

816 The predictive mean is the average of the head outputs and the predictive variance is the em-
817 pirical variance across the five outputs, both de-standardized before being passed to BoTorch’s
818 LogExpectedImprovement. We performed an ablation for this surrogate with LogEI and Thomp-
819 son Sampling and found that Thompson Sampling performs far better. For batch BO the ensemble
820 exposes a Thompson rule: each batch slot draws one head index uniformly and selects the candidate
821 with the largest predicted mean under that head.

822 **B.7 Gaussian Process (Gaussian Process)**

823 An exact GP from BoTorch with a constant mean, a scaled Matérn-5/2 ARD kernel directly on the raw
 824 FM embedding, and a Gaussian likelihood, fit by `fit_gpytorch_mll` [4]. Inputs are mapped to the
 825 unit cube (`Normalize`) and targets are standardized (`Standardize`). The kernel is initialized with
 826 output scale 1.0, lengthscale 1.0, and noise variance 1.0 under the constraint $\sigma_y^2 \geq 10^{-3}$, with the
 827 L-BFGS optimizer using a learning rate of 0.2 on the GP hyperparameters. Acquisition is BoTorch’s
 828 `LogExpectedImprovement` [1] with f^* set to the running maximum.

829 The default `Gaussian Process` runs the kernel on the full D -dimensional FM embedding (e.g.
 830 $D=1152$ for ESM-C 600M). We also tested a shallow deep-kernel variant (`gp_bottleneck`) that
 831 prepends a trainable `Linear` ($D \rightarrow 128$) + ELU projector before the kernel and co-trains the projector
 832 with the GP hyperparameters by marginal likelihood, using a separate AdamW parameter group
 833 at `lr=1e-3` and `wd=1e-3` (the kernel still uses `lr=0.2`). The bottleneck shape mirrors the VBLL
 834 projection exactly so any GP-vs-VBLL difference attributable to the projector is held fixed. Across the
 835 cohort the bottleneck variant performed worse than the no-projection `Gaussian Process`, consistent
 836 with the well-known DKL overfitting failure mode at small labelled-set sizes [62], so we report the
 837 no-projection GP as our headline GP baseline.

838 **B.8 Variational Bayesian Last Layer (VBLL)**

839 **Generative model.** A Bayesian linear regression head sits on top of a learned `Linear` ($D \rightarrow d$) + ELU
 840 projection $\phi_\theta : \mathbb{R}^D \rightarrow \mathbb{R}^d$ of the FM embedding z with $d = 128$ [36]. Writing $\phi_i = \phi_\theta(z_i)$ for
 841 brevity, the joint generative model on the labelled set $\mathcal{D}_t = \{(z_i, y_i)\}_{i=1}^{|\mathcal{D}_t|}$ is

$$y_i | \phi_i, w, \sigma_y^2 \sim \mathcal{N}(w^\top \phi_i, \sigma_y^2) \quad (\text{likelihood}), \quad (1)$$

$$w \sim \mathcal{N}(0, s_w I_d) \quad (\text{weight prior}), \quad (2)$$

$$\sigma_y^2 \sim \mathcal{W}^{-1}(\nu, s_\Sigma) \quad (\text{noise prior}), \quad (3)$$

842 where the scalar Inverse-Wishart on σ_y^2 reduces to an Inverse-Gamma in our single-output regression
 843 setting. The variational posterior factorizes as $q(w, \sigma_y^2) = q(w) q(\sigma_y^2)$ with a Gaussian $q(w) =$
 844 $\mathcal{N}(\mu, \Sigma)$ on the weights (full $d \times d$ covariance) and a deterministic $q(\sigma_y^2)$ parameterized by a single
 845 learned log-scale; the latter behaves as a MAP estimate of the noise variance under the Inverse-Wishart
 846 prior.

847 **ELBO.** Under this conjugate structure both the data-term expectation and $\text{KL}(q||p)$ are available in
 848 closed form, giving a deterministic per-batch ELBO:

$$\begin{aligned} \mathcal{L}(\theta, q) = \frac{1}{|\mathcal{D}_t|} \sum_{(z_i, y_i) \in \mathcal{D}_t} & \left[\underbrace{\log \mathcal{N}(y_i; \mu^\top \phi_i, \sigma_y^2)}_{\text{predictive log-likelihood}} - \underbrace{\frac{1}{2} \phi_i^\top \Sigma \phi_i / \sigma_y^2}_{\text{weight-uncertainty penalty}} \right] \\ & + \lambda_t \cdot \left[\underbrace{\nu \log \sigma_y^{-2} - \frac{1}{2} s_\Sigma \sigma_y^{-2}}_{\text{Inverse-Wishart prior log-density on } \sigma_y^2} - \underbrace{\text{KL}(\mathcal{N}(\mu, \Sigma) || \mathcal{N}(0, s_w I_d))}_{\text{weight KL}} \right]. \end{aligned} \quad (4)$$

849 The data term is the closed-form expectation $\mathbb{E}_{q(w)}[\log \mathcal{N}(y_i; w^\top \phi_i, \sigma_y^2)]$, which decomposes into
 850 $\log \mathcal{N}(y_i; \mu^\top \phi_i, \sigma_y^2)$ minus the weight-uncertainty penalty $\frac{1}{2} \phi_i^\top \Sigma \phi_i / \sigma_y^2$. The Inverse-Wishart term
 851 acts as a regulariser on the noise variance under the deterministic $q(\sigma_y^2)$, preventing collapse to
 852 $\sigma_y^2 \rightarrow 0$. σ_y^2 inside the ELBO denotes the current variational point estimate.

853 **Predictive distribution.** For a candidate x with embedding $z(x)$, the posterior predictive over the
 854 noiseless function value $f(x) = w^\top \phi_\theta(z(x))$ is closed-form Gaussian:

$$p(f(x) | \mathcal{D}_t) = \mathcal{N} \left(\underbrace{\mu^\top \phi_\theta(z(x))}_{\text{predictive mean}}, \underbrace{\phi_\theta(z(x))^\top \Sigma \phi_\theta(z(x))}_{\text{predictive variance}} \right). \quad (5)$$

855 The mean is the linear readout under the variational mean μ , and the variance is the quadratic form
 856 of the projected feature against the posterior covariance Σ (with the noise variance σ_y^2 added if a

857 predictive over y is required). This gives any variance-aware acquisition rule a mean and variance per
 858 candidate without Monte Carlo sampling.

859 **Hyperparameters.** Table 3 lists the head’s hyperparameters, grouped by role.

Table 3: **VBL head hyperparameters.** Grouped by role. The same settings are used across every task, modality, and FM in the main sweep; the only quantity that changes between rounds is the regularisation weight λ_t , which is rescaled by the current dataset size.

Group	Hyperparameter	Value
Architecture	Projection width d	128
	Activation	ELU
	Covariance parameterisation of $q(w)$	dense ($d \times d$)
Priors	Weight prior scale s_w	$1/d$
	Noise prior scale s_Σ	10
	Degrees of freedom ν	10
ELBO weighting	Regularisation weight λ_t	$1/ \mathcal{D}_t $ (per round)
Optimisation	Optimiser	AdamW
	Learning rate	10^{-3}
	Weight decay	10^{-3}
	Mini-batch size	64
Schedule	Gradient clip (max norm)	1.0
	Warmup	5 epochs, linear
	Decay	cosine to zero
	Max epochs / round	3000 (patience 100, smoothing 0.5)
Per-round reset	Target standardisation	yes
	FM-embedding normalisation	no
	Head / projector reinit each round	yes

860 C Acquisition functions

861 The acquisition rule turns the surrogate’s predictive distribution at each candidate into a single
 862 score and selects the candidate (sequentially) or batch of candidates (under batched selection) that
 863 maximizes it. Three rules cover every method in Table 1: *Greedy* on the predictive mean, *LogEI* on
 864 the predictive mean and variance, and *Thompson sampling* on a posterior draw. This appendix gives
 865 the sequential form of each rule and the batch construction we pair with it.

866 C.1 Sequential acquisition rules

867 Let $\mu_t(x)$ and $\sigma_t^2(x)$ denote the posterior predictive mean and variance produced by the surrogate at
 868 round t . The sequential acquisition selects $x_t = \arg \max_{x \in \mathcal{X} \setminus \mathcal{D}_{t-1}} \alpha_t(x)$, where α_t is one of:

869 **Greedy.** $\alpha_t(x) = \mu_t(x)$, which selects the candidate with the largest predicted mean. For Zero-Shot
 870 this reduces to the FM’s pseudo-likelihood, since the surrogate is the precomputed PLL itself; for the
 871 deterministic heads (Linear, MLP, RF_Greedy) it is the predicted scalar with no variance term. This
 872 is the only rule available to surrogates without an uncertainty estimate.

873 **LogEI.** $\alpha_t(x) = \log \mathbb{E}[(f(x) - f^*)_+]$, the logarithm of the expected improvement over
 874 the running maximum $f^* = \max_{s \leq t-1} y_s$. We use BoTorch’s numerically stabilised
 875 LogExpectedImprovement [1] under the surrogate’s predictive Gaussian, with $\mu_t(x)$ and $\sigma_t^2(x)$
 876 supplied by the head: the marginal predictive of the GP [73], the empirical mean and across-head
 877 variance of the MLP ensemble (de-standardized before being passed to the acquisition), and the
 878 predictive mean and tree-variance of the random forest for RF_LogEI. LogEI converts to expected
 879 improvement under a strict monotone transform, so the argmax of LogEI matches that of EI; the
 880 log form simply avoids the floating-point underflow that EI suffers when f^* is far above the current
 881 posterior mode [1].

882 **Thompson sampling.** $\alpha_t(x) = \tilde{f}_t(x)$, where \tilde{f}_t is one draw from the surrogate’s posterior over
 883 functions, and $x_t = \arg \max_x \tilde{f}_t(x)$ is the maximizer of that single draw [79]. This matches the
 884 Thompson rule’s regret guarantees and yields exploration without an explicit acquisition trade-off
 885 term: a candidate is selected because it is the optimum of *some* plausible function under the posterior,
 886 not because it has high mean or high variance per se.

887 For VBLL, the predictive in Eq. 5 is closed-form Gaussian, and a function draw is obtained by drawing
 888 the last-layer weights once: $\tilde{w} \sim \mathcal{N}(\mu, \Sigma)$ via the Cholesky factor of Σ , with $\tilde{f}_t(x) = \tilde{w}^\top \phi_\theta(z(x))$
 889 scored over the pool by a single matrix-vector product. For the MLP ensemble, a function draw is
 890 obtained by sampling one head index $k \sim \text{Uniform}\{1, \dots, M\}$ and taking $\tilde{f}_t(x) = m_k(x)$, where
 891 m_k is the k -th head’s prediction. The two constructions are not interchangeable: VBLL exposes an
 892 explicit weight posterior so each call is a fresh continuous draw, whereas the ensemble draws from a
 893 discrete posterior over M trained heads.

894 Thompson sampling is in principle also available for the GP, by drawing a joint posterior sample over
 895 the candidate pool from the GP’s N -dimensional predictive Gaussian and taking its argmax [38, 98].
 896 We do not use this rule in the main results because each draw requires a Cholesky factorization
 897 of an $N \times N$ kernel matrix at every BO round, which is $O(N^3)$ in the candidate pool size and
 898 prohibitive at the pool sizes considered here (N up to $\sim 10K$) when run across 61 tasks, 5 seeds, and
 899 $T = 100$ sequential rounds; the GP is therefore paired with LogEI throughout. The deterministic
 900 heads (Linear, MLP, RF_Greedy) and the random forest expose neither a weight posterior nor an
 901 ensemble of trained functions, so they are paired with ArgMax or LogEI rather than Thompson.

902 C.2 Batch construction

903 Batch BO at size k requires the surrogate to commit to k candidates without intermediate observations.
 904 The challenge is that the round-start posterior is the only information available for every member
 905 of the batch, so the natural single-step argmax of any acquisition concentrates the batch on a single
 906 mode of the posterior unless the batching rule explicitly diversifies.

907 We use the natural batch construction for each acquisition family. For Thompson sampling on VBLL
 908 and the MLP ensemble, batch members are independent posterior draws and their argmaxes; for
 909 VBLL we additionally apply a *projection-residual* diversity rule (App. C.3) that rejects redundant
 910 draws in covariance-weighted feature space. For the LogEI methods (GP, MLP ensemble paired
 911 with LogEI, RF paired with LogEI), the batch is the top- k candidates by the round-start LogEI score
 912 (App. C.4), since the surrogate’s hyperparameters and f^* are not refit between picks within a single
 913 batch. For greedy methods the batch is the top- k candidates by predictive mean, and for the random
 914 baseline it is a uniform sample without replacement.

915 C.3 VBLL: Thompson sampling with projection-residual diversity

916 Sequential Thompson sampling under VBLL is exact (App. B.8): a single Cholesky factorization
 917 of Σ followed by an inner product over the candidate pool. The natural batch extension at size k
 918 draws k independent posterior samples $\tilde{w}^{(1)}, \dots, \tilde{w}^{(k)} \sim q(w)$ and selects the argmax of each over
 919 the remaining heldout pool. This is correct when the posterior is broad, but for a sharp posterior
 920 the k argmaxes can concentrate on the same region of design space because every draw lies near
 921 the posterior mean. We layer a *projection-residual* diversity rule on top of the independent-sample
 922 construction to mitigate this collapse without changing the underlying Thompson semantics.

923 The rule operates in covariance-weighted feature space. For a candidate x , define $z(x) = L^\top \phi_\theta(x)$,
 924 where L is the Cholesky factor of the posterior covariance Σ (so $\|z(x)\|^2 = \phi_\theta(x)^\top \Sigma \phi_\theta(x)$ is the
 925 head’s predictive variance at x , up to noise). After committing batch members $X_j = \{x_{(1)}, \dots, x_{(j)}\}$,
 926 we maintain an orthonormal basis $Q \in \mathbb{R}^{r \times d}$ for $\text{span}\{z(x_{(1)}), \dots, z(x_{(j)})\}$ and accept the $(j+1)$ -th
 927 proposal x^* iff its residual-energy ratio

$$\rho(x^*) = \frac{\|z(x^*)\|^2 - \|Qz(x^*)\|^2}{\|z(x^*)\|^2 + \varepsilon} \quad (6)$$

928 exceeds a fixed threshold δ . The numerator is the squared length of $z(x^*)$ orthogonal to the span of
 929 already-committed members; $\rho \rightarrow 1$ when $z(x^*)$ is fully orthogonal to that span (the candidate carries
 930 information about a new direction in the posterior covariance), and $\rho \rightarrow 0$ when it is essentially

Algorithm 1 VBLL Batch Thompson Sampling with Projection-Residual Diversity

Require: Posterior $q(w) = \mathcal{N}(\mu, \Sigma)$, projection ϕ_θ , candidate pool \mathcal{X} , batch size k , max resamples R , threshold δ , tolerance ε

- 1: $L \leftarrow \text{Cholesky}(\Sigma)$
- 2: $X_0 \leftarrow \emptyset, Q \leftarrow \emptyset$
- 3: **for** $j = 0, 1, \dots, k - 1$ **do**
- 4: **for** attempt = 0, 1, \dots, R **do**
- 5: Draw $\tilde{w} \sim \mathcal{N}(\mu, \Sigma)$
- 6: $x^* \leftarrow \arg \max_{x \in \mathcal{X} \setminus X_j} \tilde{w}^\top \phi_\theta(x)$
- 7: $z \leftarrow L^\top \phi_\theta(x^*)$
- 8: **if** $Q = \emptyset$ **or** $\|z\|^2 \leq \varepsilon$ **or** attempt = R **then**
- 9: accept \leftarrow **true**
- 10: **else**
- 11: $\rho \leftarrow (\|z\|^2 - \|Qz\|^2) / (\|z\|^2 + \varepsilon)$
- 12: accept $\leftarrow (\rho \geq \delta)$
- 13: **end if**
- 14: **if** accept **then**
- 15: $X_{j+1} \leftarrow X_j \cup \{x^*\}$
- 16: Append the unit-normalized residual of z against Q to Q (Gram–Schmidt)
- 17: **break**
- 18: **end if**
- 19: **end for**
- 20: **end for**
- 21: **return** X_k

931 redundant. If $\rho(x^*) < \delta$ we draw a fresh posterior sample and a fresh argmax, up to R resamples,
932 after which we accept the last candidate as a fallback so the batch budget is always filled. On
933 accept, Q is extended by Gram–Schmidt orthogonalization of $z(x^*)$ against the current basis. The
934 numerical tolerance ε avoids division by zero; if $\|z(x^*)\|^2 \leq \varepsilon$ the head reports essentially no
935 posterior uncertainty at x^* and the candidate is accepted unconditionally so that diversity filtering
936 does not block batch completion. We use $\delta = 10^{-3}$, $R = 5$, and $\varepsilon = 10^{-8}$ in all experiments.

937 The per-batch cost is $O(k(d^2 + Nd) + k^2d)$: a single Cholesky reused across all draws, k posterior
938 draws plus pool scorings, and a Gram–Schmidt update that grows linearly in the basis size. This
939 is independent of $|\mathcal{D}_{t-1}|$ and of the FM size, since the FM enters only through the precomputed
940 embedding z .

941 **Ablation against plain Thompson sampling.** We compared this rule against the plain independent-
942 sample construction (the same algorithm with $\delta = 0$, equivalent to skipping the residual check) at
943 every batch size $k \in \{24, 48, 96, 192, 384\}$ used in the main batch sweep. Across the cohort, the
944 projection-residual variant slightly improved both NBSF and Top-5% count over plain Thompson
945 sampling, with the gap most visible at the largest batch sizes where the posterior is sharpest and the
946 redundancy mode is most pronounced. The improvement is small enough on aggregate that we report
947 only the projection-residual variant in the main results, but the ablation justifies it as the default and
948 is consistent with prior reports that Thompson batches benefit from explicit redundancy filtering at
949 large k [10].

950 C.4 LogEI surrogates: top- k acquisition argmax

951 For LogEI on the GP, the LogEI variant of the MLP ensemble, and RF_LogEI, the batch is the top- k
952 candidates by the round-start LogEI score over the heldout pool. Within a single batch we do not refit
953 the surrogate or update f^* between picks; the predictive mean, predictive variance, and improvement
954 target are all held at their round-start values. This is by construction equivalent to the constant-liar
955 batch rule with the model’s posterior mean as the lie [28, 99]: the lie does not change the LogEI score
956 at the remaining points because the surrogate is not refit, so sequentially appending $(x^{(j)}, \mu_t(x^{(j)}))$
957 and recomputing the argmax produces the same ordering as a single sort by the original LogEI values.
958 We use this top- k form in the main results because the equivalence is exact and the implementation is
959 $O(N \log k)$ instead of $O(kN)$.

960 This batch rule does not encode any explicit diversity: at large k all k picks come from a single
961 round-start posterior, and they tend to cluster around the same posterior maximum. The collapse is
962 most visible on the GP, whose predictive variance is small in the well-explored part of the pool, and
963 is the failure mode underlying the GP’s poor scaling at large batch in Sec. 3.2.

964 **Sequential kriging-believer ablation.** We additionally tested a true sequential kriging-believer
965 variant for the GP [28], in which after each pick the GP is conditioned on the pseudo-observation
966 $(x^{(j)}, \mu_t(x^{(j)}))$ via Gaussian conditioning (no hyperparameter refit) and the LogEI is recomputed on
967 the remaining pool. Because conditioning lowers the predictive variance near $x^{(j)}$, this rule diversifies
968 the batch at the cost of k posterior conditionings per round. On our cohort the sequential variant
969 actually performed slightly worse than top- k LogEI, and the runtime was substantially larger. We
970 therefore report top- k LogEI as the headline GP batch rule.

971 C.5 Greedy and random surrogates

972 The greedy heads (Linear, MLP, RF_Greedy, Zero-Shot) have no per-candidate uncertainty and no
973 posterior to sample from. Their batch construction is top- k on the predictive mean (ArgMax on μ_t),
974 with no diversity term. This matches their sequential rule one-to-one: nothing about the surrogate
975 changes between members of the batch, so the k best candidates under the round-start posterior are
976 exactly the k candidates the rule would select if run sequentially without intermediate observations.
977 The Random baseline samples k candidates uniformly without replacement from the heldout pool,
978 independently of the surrogate.

979 D Datasets

980 The main benchmark contains 61 fixed-pool scalar optimization tasks: 18 protein, 12 chemistry, 11
981 DNA, and 20 RNA. The broader benchmark collection contains additional audit, ablation, and legacy
982 tasks; the selected cohort below is the one used for the paper’s aggregate figures. Each task is treated
983 as a finite candidate pool with a measured or computed scalar objective, and all methods compare on
984 the same candidate set for that task.

985 **Selection criteria.** We include tasks that satisfy four constraints. First, the input is a discrete
986 scientific object that can be passed to the modality FM without changing the representation: amino-
987 acid sequence, DNA sequence, RNA sequence, or SMILES. Second, the target is a scalar regression
988 objective with a well-defined optimization direction; ordinal or classification-only datasets such
989 as `nucb` and `vapdiff` are excluded. Third, the deduplicated candidate pool is large enough for a
990 96-candidate initialization plus a 100-round sequential campaign without exhausting the landscape.
991 Fourth, the source task is from an established benchmark or primary assay collection with enough
992 provenance to reproduce the curated task. The selected set therefore emphasizes real biological assays
993 and standard molecular-property benchmarks, while excluding chemistry reaction-procedure tasks
994 whose natural-language inputs are not comparable to the SMILES-only chemistry FM pipeline, very
995 small panels, deprecated duplicate tasks, and context-length edge cases.

996 **Preprocessing and target convention.** Preprocessing removes rows with missing inputs or missing
997 labels, canonicalizes the input representation, drops exact duplicate candidates when the source
998 permits, and keeps one scalar target per task. All tasks are maximization tasks except `enamine10k`,
999 where lower docking score is better. For each seed, the initial 96 candidates are drawn from the below-
1000 median half of the task’s target distribution, forcing every method to improve from a deliberately
1001 weak starting set.

1002 **Subsampling policy.** Subsampling is deterministic and task-level, never method-level. When a
1003 source landscape is much larger than the campaign budget, the runtime CSV is a fixed subsample,
1004 usually stored as `<task>_10k.csv` and generated with seed 42; the full source files remain on
1005 disk for ablations when available. Other selected tasks use the full curated assay table because it is
1006 already a finite benchmark pool and embeddings were cached for the full CSV; the exact runtime
1007 N is reported in the tables. For the SGPO comparison, we use the full combinatorially complete
1008 landscapes for `gb1`, `phoq`, and `trpb_4site`, so every sequence proposed by either method can be
1009 scored against the complete experimental map. Separate LLM-context subsamples, such as `gb1_1500`
1010 and `gosai_skns_h_2100`, are not part of the 61-task BO cohort.

1011 **Protein tasks.** Protein tasks combine FLIP [17], FLIP2 [21], EvolvePro / ProteinGym [61], and
 1012 complete combinatorial landscapes used for the SGPO comparison. The optimized properties
 1013 cover viral packaging, binding, enzymatic activity, viral fitness, protein stability, and genetic-circuit
 1014 signaling.

Table 4: **Protein benchmark tasks.** N_{orig} is the original curated dataset size before any campaign subsampling.

Task	Benchmark Suite	N_{orig}	Dataset description	Original citation
aav	FLIP	283,775	AAV2 capsid VP1-region DMS; maximize viral packaging viability.	Bryant et al., 2021 [11]
alpha_amylase	FLIP2	7,467	Single-mutant <i>Bacillus</i> alpha-amylase assay; maximize DP3 substrate-hydrolysis activity.	van der Flier et al., 2024 [93]
brenan	EvolvePro ProteinGym	6,810	Human MAPK1 / ERK2 kinase DMS; maximize SCH-condition kinase signaling activity.	Brenan et al., 2016 [8]
cas12f	EvolvePro	7,942	AsCas12f1 miniature CRISPR nuclease DMS; maximize genome-editing / DNA-cleavage efficiency.	Hino et al., 2023 [40]
doud	EvolvePro ProteinGym	10,716	H1N1 influenza hemagglutinin DMS; maximize viral replicative fitness.	Doud and Bloom, 2016 [23]
gb1	FLIP ProteinGym	149,361	Four-position GB1 combinatorial DMS; maximize IgG-Fc binding fitness.	Wu et al., 2016 [100]
giacomelli	EvolvePro ProteinGym	7,448	Human TP53 DMS under etoposide selection; maximize tumor-suppressor function.	Giacomelli et al., 2018 [27]
haddox	EvolvePro ProteinGym	12,863	HIV-1 Env DMS; maximize viral infectivity / replicative fitness.	Haddox et al., 2018 [35]
hydro	FLIP2	24,935	HIS3 ortholog combinatorial DMS; maximize fitness across hydrophobic-core mutations.	Pokusaeva et al., 2019 [66]
ired	FLIP2	10,905	Imine reductase DMS; maximize catalytic fitness on the active substrate.	Gantz et al., 2024 [25]
jones	EvolvePro ProteinGym	7,800	Human beta-2 adrenergic receptor DMS; maximize agonist-dependent signaling activity.	Jones et al., 2020 [45]
lee	EvolvePro ProteinGym	10,754	H3N2 influenza hemagglutinin DMS; maximize viral replication capacity.	Lee et al., 2018 [53]
meltome_human	FLIP	9,792	Human proteome melting-temperature dataset; maximize protein thermostability (T_m).	Jarzab et al., 2020 [42]
meltome_mixed	FLIP	23,340	Mixed-species proteome melting-temperature dataset; maximize protein thermostability (T_m).	Jarzab et al., 2020 [42]
phoq	PhoQ landscape	140,517	PhoQ four-site histidine-kinase landscape; maximize signaling fitness.	Podgornaia and Laub, 2015 [65]
stiffler	EvolvePro ProteinGym	4,978	TEM-1 beta-lactamase DMS; maximize ampicillin-resistance fitness.	Stiffler et al., 2015 [88]
trpb_4site	FLIP2	159,129	Four-site PfTrpB combinatorial landscape; maximize beta-replacement reaction fitness.	Johnston et al., 2024 [44]
zikv_E	EvolvePro ProteinGym	9,577	Zika-virus envelope-protein DMS; maximize viral growth / replication fitness.	Sourisseau et al., 2019 [85]

1015 **Chemistry tasks.** The chemistry cohort is SMILES-only. It includes three GOLLuM molecular-
 1016 property tasks [72] and nine MoleculeACE activity-cliff panels [94]. Reaction-yield tasks with
 1017 procedure text are intentionally excluded from the main cohort because they use a different input
 1018 representation.

Table 5: **Chemistry benchmark tasks.**

Task	Benchmark Suite	N_{orig}	Dataset description	Original citation
d3r_dopamine	MoleculeACE	3,651	Dopamine D3 receptor ligands; maximize pChEMBL / pK_i binding affinity.	van Tilborg et al., 2022 [94]
delta_opioid	MoleculeACE	2,598	Delta-opioid receptor ligands; maximize pChEMBL / pK_i binding affinity.	van Tilborg et al., 2022 [94]

Task	Benchmark Suite	N_{orig}	Dataset description	Original citation
enamine10k	GOLLuM DOCKSTRING	10,449	Enamine REAL molecules docked to a target receptor; minimize docking score.	García-Ortegón et al., 2022 [26]
f2_thrombin	MoleculeACE	2,750	Thrombin / coagulation factor II inhibitors; maximize pChEMBL potency.	van Tilborg et al., 2022 [94]
fx_factor_x	MoleculeACE	3,093	Factor Xa inhibitors; maximize pChEMBL potency.	van Tilborg et al., 2022 [94]
hrh3_histamine	MoleculeACE	2,862	Histamine H3 receptor ligands; maximize pChEMBL / pK_i binding affinity.	van Tilborg et al., 2022 [94]
kappa_opioid	MoleculeACE	2,603	Kappa-opioid receptor ligands; maximize pChEMBL / pK_i binding affinity.	van Tilborg et al., 2022 [94]
laser_multi10k	GOLLuM	10,000	Laser / photonic candidate molecules; maximize computed fluorescence oscillator strength.	Pollice et al., 2021 [68]
mor_mu_opioid	MoleculeACE	3,142	Mu-opioid receptor ligands; maximize pChEMBL / pK_i binding affinity.	van Tilborg et al., 2022 [94]
photovoltaics_pce10k	GOLLuM CEP	2,322,849	Organic photovoltaic donor molecules; maximize simulated power-conversion efficiency.	Hachmann et al., 2011 [34]
pparg	MoleculeACE	2,349	PPAR-gamma ligands; maximize pChEMBL activity / potency.	van Tilborg et al., 2022 [94]
serotonin_5ht1a	MoleculeACE	3,317	Serotonin 5-HT1A receptor ligands; maximize pChEMBL / pK_i binding affinity.	van Tilborg et al., 2022 [94]

1019 **DNA tasks.** DNA tasks are MPRA or promoter-expression datasets. They are grouped into random/synthetic promoter benchmarks, DeepSTARR enhancer activity, Gosai human cis-regulatory
1020 element activity, and PARM / Barbadilla promoter MPRA tasks. The five `barbadilla_*` tasks are
1021 derived from the same promoter-fragment library and differ only in the expression or perturbation
1022 target.
1023

Table 6: **DNA benchmark tasks.**

Task	Benchmark Suite	N_{orig}	Dataset description	Original citation
barbadilla_hepg2	PARM MPRA	4,449,234	Human promoter-fragment MPRA in HepG2 cells; maximize basal expression.	Barbadilla-Martínez et al., 2026 [5]
barbadilla_hepg2_nutlin3a_lfc	PARM MPRA	4,449,234	Same promoter-fragment library in HepG2; maximize Nutlin-3a perturbation log-fold change.	Barbadilla-Martínez et al., 2026 [5]
barbadilla_k562	PARM MPRA	4,449,234	Human promoter-fragment MPRA in K562 cells; maximize basal expression.	Barbadilla-Martínez et al., 2026 [5]
barbadilla_k562_heatshock_lfc	PARM MPRA	4,449,234	Same promoter-fragment library in K562; maximize heat-shock perturbation log-fold change.	Barbadilla-Martínez et al., 2026 [5]
barbadilla_k562_pma_lfc	PARM MPRA	4,449,234	Same promoter-fragment library in K562; maximize PMA perturbation log-fold change.	Barbadilla-Martínez et al., 2026 [5]
deepstarr_dev	DeepSTARR	484,052	Drosophila STARR-seq enhancer library; maximize developmental-enhancer activity.	de Almeida et al., 2022 [18]
deepstarr_hk	DeepSTARR	484,052	Same enhancer library; maximize housekeeping-enhancer activity.	de Almeida et al., 2022 [18]
dream	DREAM	7,413,183	Random synthetic yeast promoters; maximize measured promoter expression.	Rafi et al., 2025 [70]
gosai_hepg2	Gosai MPRA	CRE 798,064	Human cis-regulatory elements assayed in HepG2 cells; maximize regulatory activity.	Gosai et al., 2024 [31]
gosai_k562	Gosai MPRA	CRE 798,064	Human cis-regulatory elements assayed in K562 cells; maximize regulatory activity.	Gosai et al., 2024 [31]

Task	Benchmark Suite	N_{orig}	Dataset description	Original citation
gosai_sknsh	Gosai MPRA	CRE 798,064	Human cis-regulatory elements assayed in SK-N-SH cells; maximize regulatory activity.	Gosai et al., 2024 [31]

1024 **RNA tasks.** The RNA cohort is dominated by RNAgym [3] ribozyme and tRNA fitness landscapes,
1025 with tasks spanning self-cleavage, splicing, aptamer binding, mRNA expression, and tRNA suppression.
1026 Several sources contain multiple assays on related ribozyme families; these are kept as separate
1027 tasks when they have distinct candidate pools or distinct measured targets.

Table 7: RNA benchmark tasks.

Task	Benchmark Suite	N_{orig}	Dataset description	Original citation
andreasson_ribozyme	RNAgym	7,343	Cofactor-dependent <i>glmS</i> self-cleaving ribozyme variants; maximize cleavage activity.	Andreasson et al., 2020 [2]
cpeb3_ribozyme	RNAgym	111,417	Human CPEB3 HDV-like ribozyme DMS; maximize self-cleavage activity.	Zhang et al., 2020 [105]
domingo_trna	RNAgym	4,175	Yeast tRNA fitness landscape; maximize translation / aminoacylation fitness.	Domingo et al., 2018 [22]
hairpin_ribozyme	RNAgym	22,578	Hairpin self-cleaving ribozyme DMS; maximize cleavage activity.	Roberts et al., 2023 [76]
hammerhead_ribozyme	RNAgym	9,046	Type-III hammerhead ribozyme mutational library; maximize self-cleavage activity.	Roberts et al., 2023 [76]
hdv_ribozyme	RNAgym	33,930	Hepatitis-delta-virus ribozyme DMS; maximize self-cleavage activity.	Roberts et al., 2023 [76]
ke_mrna	RNAgym	5,559	mRNA exon-definition saturation mutagenesis; maximize splicing-reporter expression.	Ke et al., 2018 [47]
mcrae_5tu_ribozyme	RNAgym	74,942	5TU catalytic subunit of a triplet polymerase ribozyme (TPR); maximize polymerase activity.	McRae et al., 2024 [57]
mcrae_t1_ribozyme	RNAgym	47,503	t1 scaffolding subunit of a triplet polymerase ribozyme (TPR); maximize polymerase activity.	McRae et al., 2024 [57]
nelfe_aptamer	RNAgym	2,652	RNA aptamer variants selected against NELF-E; maximize protein-binding activity.	Tome et al., 2014 [92]
osa_twister_ribozyme	RNAgym	10,296	<i>Oryza sativa</i> twister-class ribozyme variants; maximize self-cleavage activity.	Kobori and Yokobayashi, 2016 [50]
peri_ribozyme	RNAgym	16,383	Azoarcus group-I ribozyme activity landscape; maximize activity score.	Peri et al., 2022 [64]
pistol_aptazyme	RNAgym	16,383	Pistol aptazyme saturation library; maximize log-enrichment / cleavage activity.	Kobori et al., 2017 [49]
soo_ribozyme	RNAgym	63,430	Tetrahymena group-I intron self-splicing ribozyme fitness landscape; maximize splicing activity.	Soo et al., 2021 [84]
sup4_trna_tyr	RNAgym	25,491	SUP4-oc tRNA-Tyr DMS; maximize amber-suppression / RTD-survival function.	Guy et al., 2014 [33]
trna_arg	RNAgym	65,537	Yeast SUP4-o tRNA-Arg(UCU) saturation landscape; maximize suppression-based fitness.	Li et al., 2016 [54]
twister_ribozyme	RNAgym	10,296	Twister self-cleaving ribozyme DMS; maximize cleavage activity.	Roberts et al., 2023 [76]
zhang_line1_full_ribozyme	RNAgym	69,583	Full LINE1-associated twister-sister-like self-cleaving ribozyme library; maximize cleavage activity.	Zhang et al., 2024 [104]
zhang_line1_mini_ribozyme	RNAgym	149,710	Minimized LINE1-associated twister-sister-like ribozyme library; maximize cleavage activity.	Zhang et al., 2024 [104]
zhang_or4k15_ribozyme	RNAgym	61,393	OR4K15-associated twister-sister-like self-cleaving ribozyme library; maximize cleavage activity.	Zhang et al., 2024 [104]

1028 E Foundation-model comparison per modality

1029 To isolate the effect of the foundation-model backbone, we rerun the entire surrogate sweep with the
 1030 alternative FM swapped in for each modality (Table 2) while holding the task list, the BO loop, the
 1031 seeds, and every surrogate hyperparameter fixed. The pairings are ESM-C 600M vs. SaProt 650M
 1032 (protein), RNA-FM vs. RiNALMo-giga (RNA), NTv3-650M vs. Evo2-1B (DNA), and ChemFM-1B
 1033 vs. SMI-TED (chemistry). Each modality is evaluated on the intersection of candidate pools that both
 1034 FMs admit so that any difference is attributable to the representation rather than to coverage. For
 1035 every (FM, surrogate, task, seed) tuple we record two BO outcomes: the normalized best-so-far at the
 1036 end of the budget, and the cumulative count of acquired candidates whose true objective lies above
 1037 the 95th percentile of the pool.

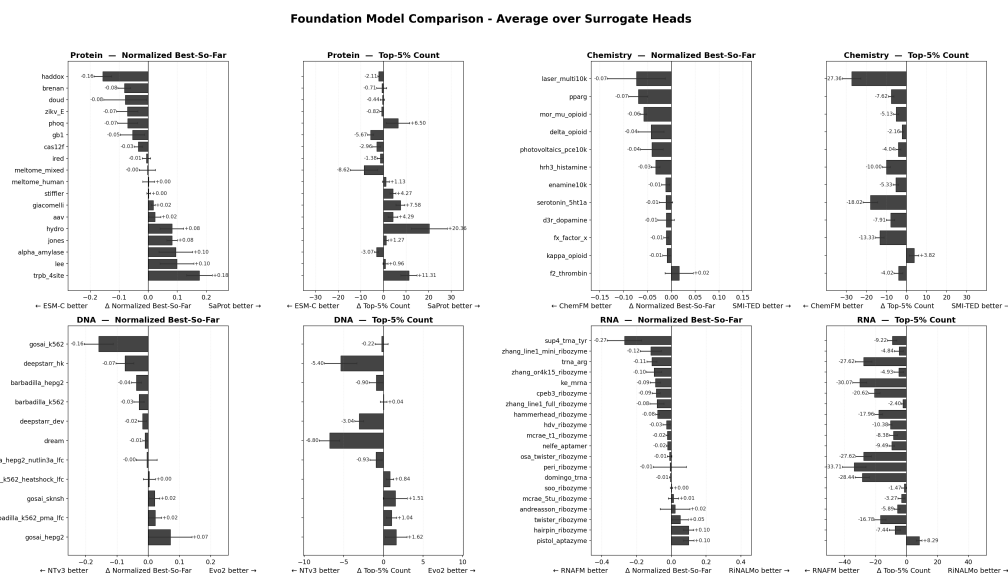


Figure 6: **Backbone sensitivity per modality.** Two panels per modality (normalized best-so-far on the left, top-5% acquired count on the right) report a per-task signed difference between the primary FM and its principal alternative (Table 2), averaged over surrogate heads and seeds. Each horizontal bar corresponds to one task; the sign indicates which FM is preferred (the labels under each axis name the winning side), and the magnitude reports the size of the gap on that metric. ESM-C 600M and SaProt 650M trade wins task by task and end up roughly on par overall, indicating that the structure-aware 3Di vocabulary helps on some landscapes but does not displace the sequence-only encoder as a default. Chemistry and RNA show clearer separations in favor of the primaries (ChemFM over SMI-TED, RNA-FM over RiNALMo-giga), with RNA exhibiting the largest and most uniform per-task gap despite RiNALMo-giga’s higher parameter count. DNA is consistent with the rest: NTv3-650M is preferred over Evo2-1B across most tasks on both metrics, validating the encoder choice for sequences whose downstream signal is dominated by short-range regulatory motifs. The takeaway is that scale alone does not predict suitability for BO; the encoder vs. decoder distinction and the alignment between the pretraining objective and the candidate space matter at least as much.

1038 F Fine-Tuning

1039 The main results in Section 3 freeze the foundation model and train only the surrogate head on cached
 1040 embeddings. This appendix reports the two regimes that update the FM itself: *supervised* LoRA
 1041 fine-tuning on the labelled BO set, and *self-supervised* LoRA pretraining on a target-family MSA.
 1042 Both are evaluated on protein tasks; the SSL regime requires a wild-type sequence and an alignable
 1043 family, which has no clean equivalent in the other modalities.

1044 **F.1 Supervised Fine-Tuning**

1045 In the Supervised regime, LoRA adapters [41] are co-trained with the surrogate head on \mathcal{D}_{t-1}
 1046 each round; adapters and head are reinitialized between rounds. We test three representative heads:
 1047 MLP_Greedy, VBLL, and GP_LogEI, the last reproducing the deep-kernel-GP-with-LoRA recipe from
 1048 GoLLuM [72] with FM, LoRA rank, and acquisition matched to the frozen baseline.

1049 Figure 7 shows that gains are head-dependent: MLP_Greedy and VBLL both improve with co-training,
 1050 especially on Top-5%. GP_LogEI does not run to completion (cross marker, GP panel): backpropa-
 1051 gating the marginal likelihood through both the kernel matrix and the adapter exhausts GPU memory
 1052 partway through every protein task. This is intrinsic to the GoLLuM-style recipe at our FM scale, not
 1053 a tuning issue, and motivates our focus on variational and deterministic heads for SFT.

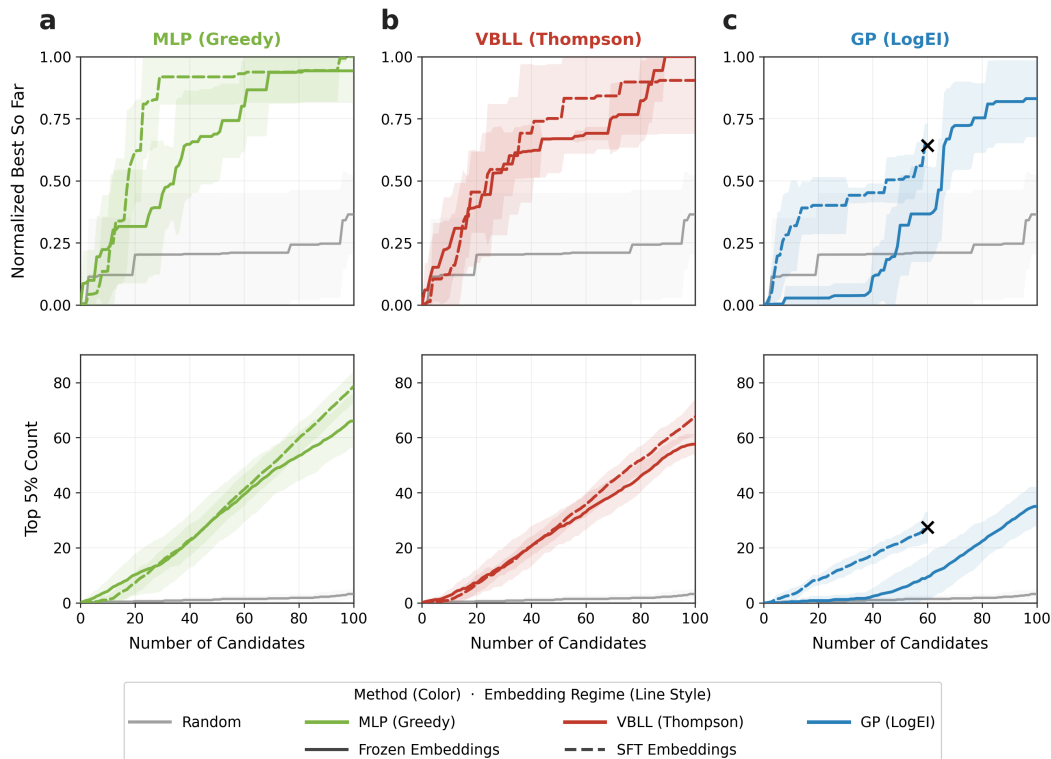


Figure 7: **Supervised LoRA fine-tuning on representative protein tasks.** NBSF (top) and Top-5% count (bottom) over $T=100$ sequential rounds for MLP_Greedy, VBLL, and GP_LogEI. Solid lines: frozen embeddings. Dashed lines: supervised LoRA. Bands are ± 1 SEM across tasks and seeds. Gains are largest for MLP_Greedy and VBLL; GP_LogEI ends up consistently breaking at that step due to memory issues.

1054 **F.2 Self-Supervised Fine-Tuning**

1055 In the Self-Supervised regime, a single LoRA adapter is pretrained once *before* the BO loop using a
 1056 masked-language-model objective on a target-family MSA, in the style of ESM-1v [58]. For each task,
 1057 the wild-type sequence is used as a query for a sequence-database search, hits are clustered at 80%
 1058 identity, and training samples are weighted by inverse cluster frequency to prevent over-representation
 1059 of dense regions of sequence space. The pretrained adapter is then loaded in place of a random init
 1060 and frozen, so SSL acts as a *target-aware feature-extractor delta* on top of the base FM. We evaluate
 1061 this regime on 15 of the 18 protein tasks; the remaining three (combinatorial-library or ortholog-panel
 1062 datasets) lack a single wild-type query sequence, so the family-MSA construction step does not apply.

1063 Figures 8 and 9 report per-task Δ NBSF and Δ Top-5% for each (task, head) pair. The picture is
 1064 strongly heterogeneous: SSL is clearly beneficial on a subset of tasks (large green bars) and clearly
 1065 harmful on another (large red bars), with the sign and magnitude depending on both task and head.

1066 Aggregating across tasks, mean NBSF changes are small and within 1 SEM for every method,
 1067 while Top-5% coverage tends to improve for the simpler greedy heads (Zero_Shot, RF_Argmax,
 1068 Linear_Argmax, MLP_Argmax, RF_LogEI) and stays roughly flat for the stronger uncertainty-aware
 1069 heads (MLP_Ensemble, GP, VBLL). Tasks where the wild type is already a high-fitness reference, and
 1070 where the family MSA is dense and well-aligned, are the most consistent winners. We therefore
 1071 do *not* adopt SSL as a default in the main results; it is a useful task-conditional lever rather than a
 1072 uniform improvement.



Figure 8: **Per-task Δ NBSF (SSL – original), 15 protein tasks.** One panel per task; bars are method-level differences, green for SSL improving, red for SSL degrading. Error bars are ± 1 SEM across seeds. Aggregate NBSF changes are within 1 SEM, but per-task swings are large and bidirectional.

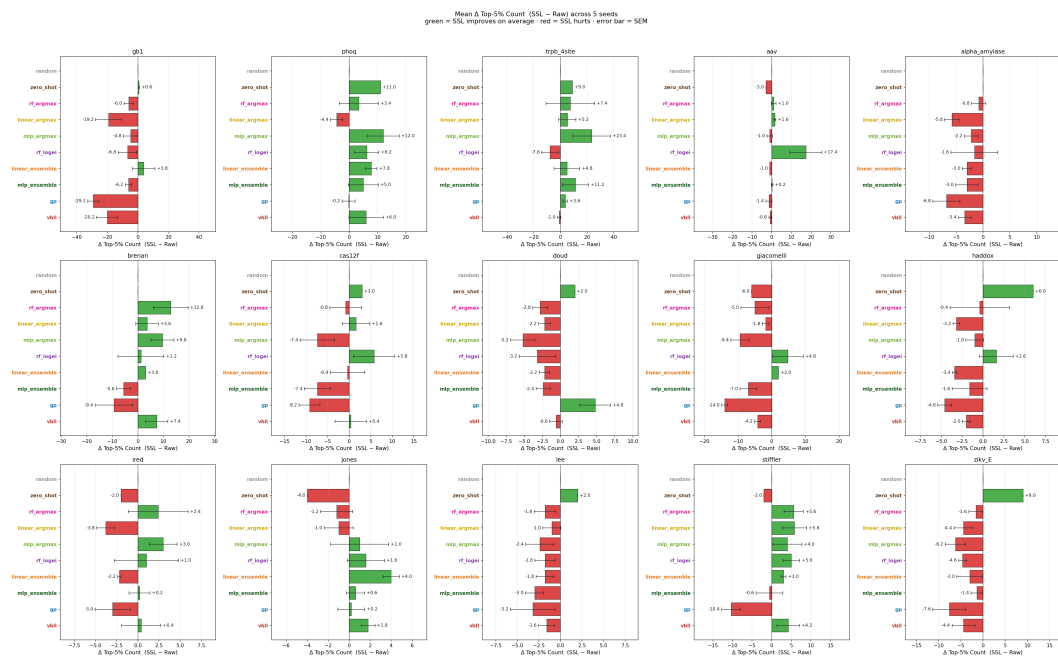


Figure 9: **Per-task Δ Top-5% (SSL - original), 15 protein tasks.** Same layout as Figure 8 for elite-candidate coverage. Greedy heads gain most often; uncertainty-aware heads are roughly unchanged on aggregate.

1073 **G Sequential Bayesian Optimization**

1074 This appendix expands on the sequential results in §3.1. We first show the across-task Pareto frontier
 1075 and the round-by-round learning curves that produce it, then turn to the variance side via expected
 1076 calibration error (ECE) for the uncertainty-aware heads.

1077 **G.1 Detailed Results**

1078 Figure 10 pools all 61 tasks into a single endpoint summary per method, with both metrics evaluated
 1079 at the end of the $T=100$ -round budget. MLP Ensemble (Thompson) and VBLL (Thompson) sit
 1080 at the upper-right frontier, MLP (Greedy) is essentially tied with them on best-so-far and only
 1081 marginally behind on top-5% count, and Gaussian Process (LogEI) matches the leaders on best-
 1082 so-far but acquires meaningfully fewer top-5% candidates. Linear (Greedy), Random Forest
 1083 (LogEI), and Random Forest (Greedy) form the middle tier, while Zero-Shot (Likelihood)
 1084 and Random are dominated on both axes. Figure 11 unrolls those endpoints over rounds and splits by
 1085 modality: rows are protein, chemistry, RNA, and DNA, the left column is normalized best-so-far,
 1086 and the right column is top-5% count. The two columns behave differently in time. Best-so-far gaps
 1087 narrow late as most methods eventually find a near-optimum, while top-5% gaps keep widening for
 1088 the remainder of the budget, because that metric rewards continuing to acquire high-value designs
 1089 after the first one is in hand. Random and Zero-Shot (Likelihood) stay roughly linear throughout,
 1090 consistent with non-adaptive sampling.

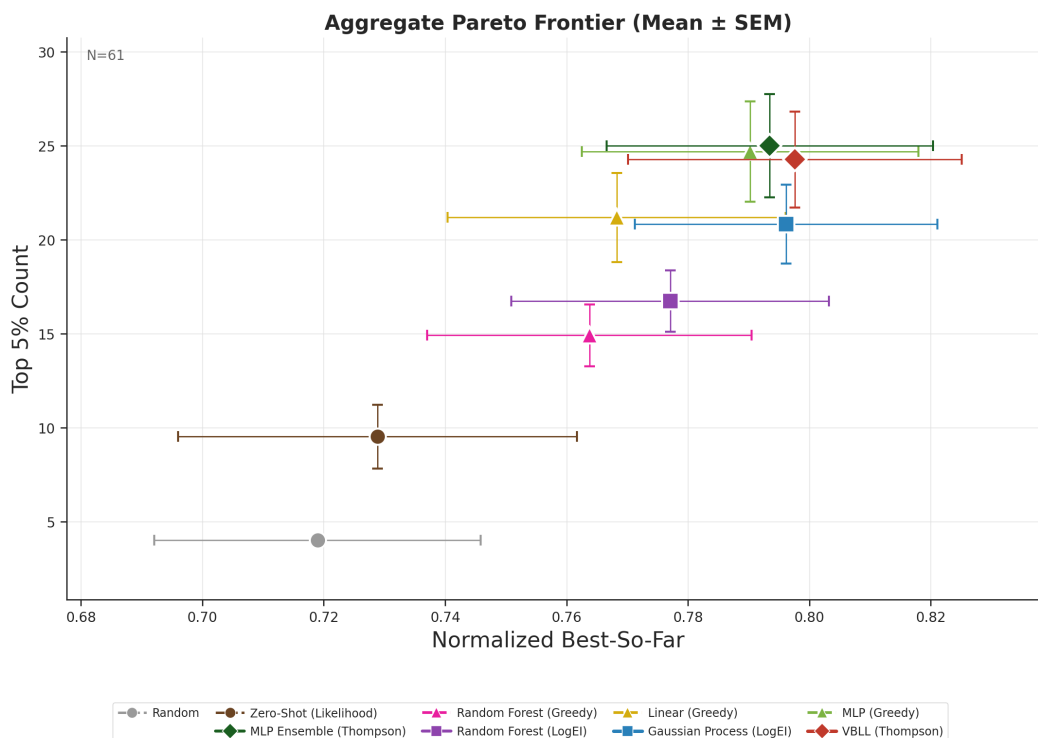


Figure 10: **Aggregate Pareto frontier of normalized best-so-far vs. top-5% count, pooled across all 61 sequential tasks ($T=100, k=1$).** Each marker is one method aggregated over tasks and seeds; error bars are ± 1 SEM. The upper-right corner is the joint optimum.

Sequential BO — aggregate by modality (\pm SEM)

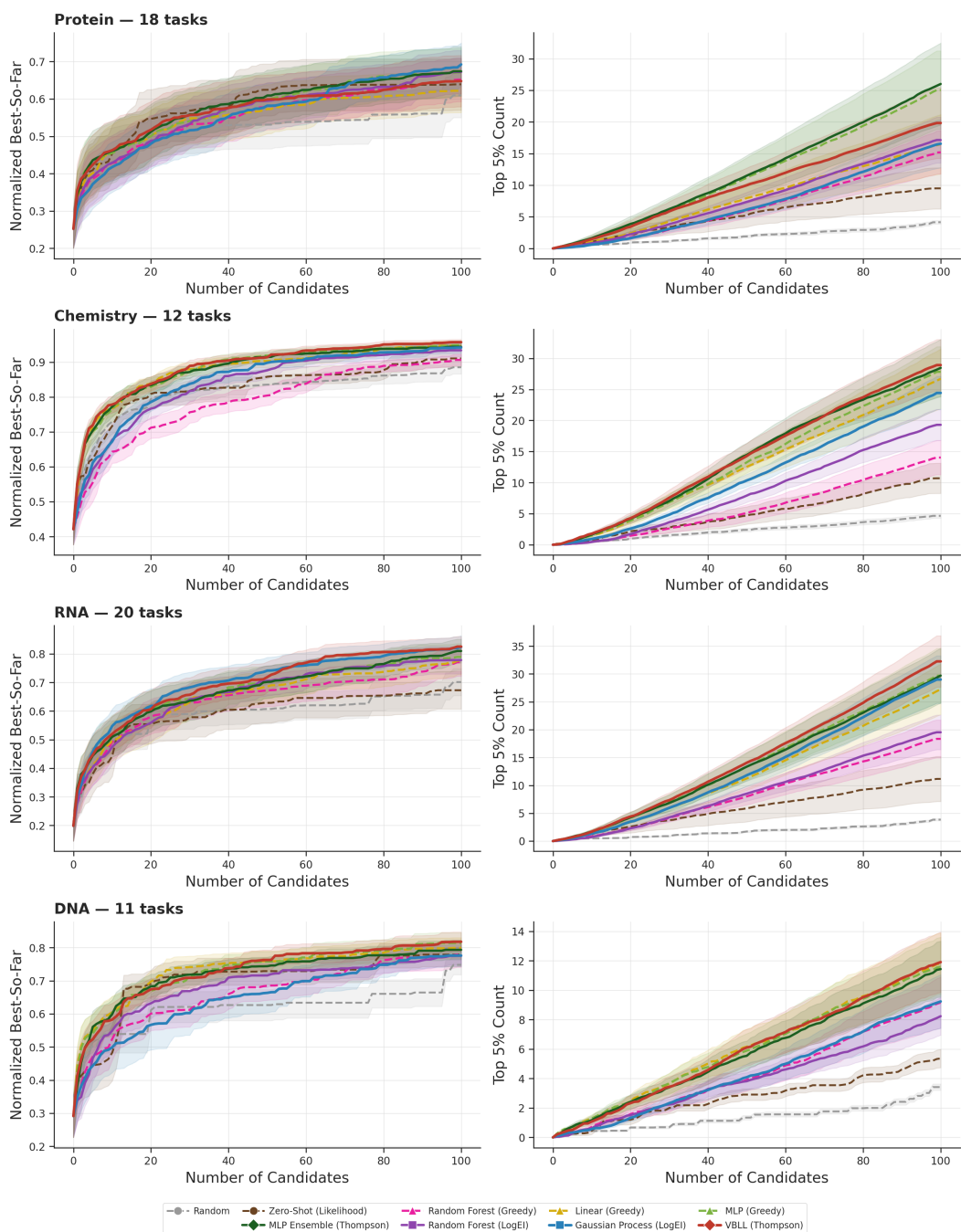


Figure 11: **Per-modality sequential learning curves.** Rows are protein (18 tasks), chemistry (12), RNA (22), and DNA (11); the left column is normalized best-so-far and the right column is top-5% count, both as a function of round t . Bands are ± 1 SEM across tasks and seeds. Best-so-far gaps tighten as the budget grows; top-5% gaps widen.

1091 **G.2 Calibration**

1092 For the three surrogates that expose a closed-form Gaussian predictive distribution (Gaussian
 1093 Process (LogEI), VBLL (Thompson), and MLP Ensemble (Thompson)) we additionally track
 1094 cumulative expected calibration error (ECE) on the heldout pool, recomputed at every BO round.
 1095 ECE is the standard binned reliability gap between predicted and empirical coverage; a perfectly
 1096 calibrated regression posterior would drive it to zero as the loop accumulates labels. The exact GP
 1097 is the calibration reference by construction. VBLL starts uncalibrated but tightens within the first
 1098 ~ 20 rounds and then trails the GP by a roughly constant offset for the remainder of the budget. The
 1099 MLP ensemble stays badly miscalibrated throughout: the disagreement signal across heads does not
 1100 match the true predictive variance and barely improves with more data. The cold-start MLP (Greedy)
 1101 and the random-forest heads are omitted because their uncertainty is either implicit (round-to-round
 1102 reinitialization) or non-Gaussian (tree variance), and ECE under a Gaussian assumption is not the
 1103 right diagnostic for them.

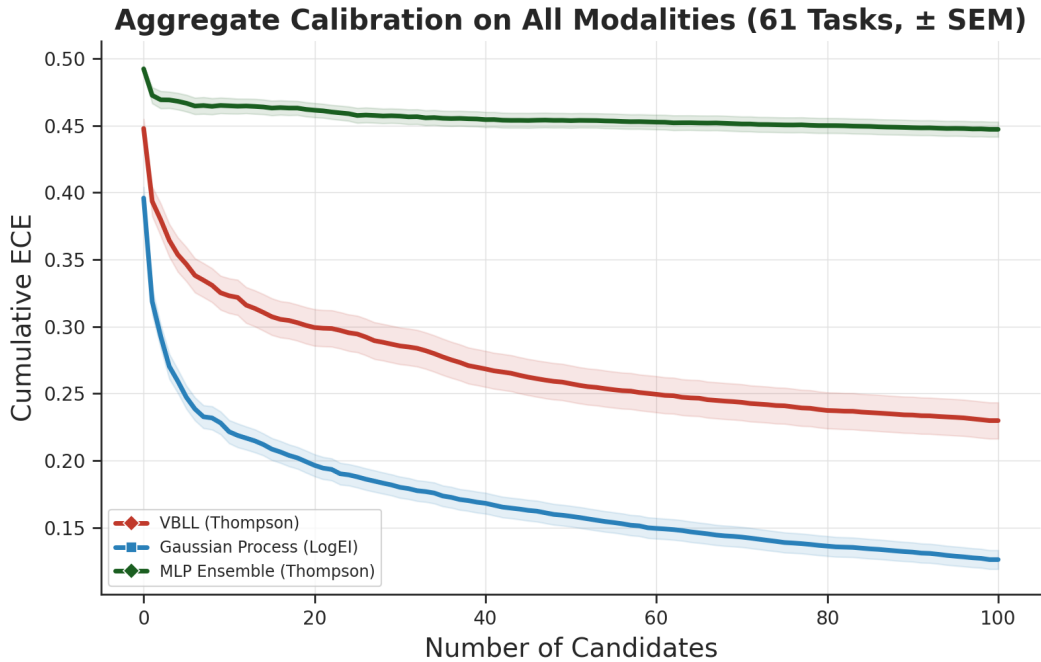


Figure 12: **Aggregate cumulative ECE over all 61 sequential tasks (\pm SEM).** Lower is better. Gaussian Process (LogEI) is the calibration ceiling. VBLL (Thompson) tightens quickly and trails the GP by a roughly constant offset. MLP Ensemble (Thompson) stays nearly flat at high ECE.

1104 The surprising takeaway is that calibration quality and BO performance are essentially decoupled
 1105 in this regime. The MLP ensemble has by far the worst ECE in Figure 12, yet it sits on the Pareto
 1106 frontier in Figure 10 and matches or beats the GP on top-5% count across modalities in Figure 11.
 1107 The GP, conversely, is the best-calibrated method by a wide margin and still loses on the acquisition
 1108 metric. What appears to matter for discrete BO over a fixed candidate pool is whether the posterior
 1109 preserves the *ranking* of high-value candidates and supplies enough disagreement to keep Thompson
 1110 sampling exploratory; faithfully reproducing posterior coverage is neither necessary nor sufficient for
 1111 that. A natural interpretation is that ECE is computed over the entire heldout pool, whereas BO only
 1112 ever queries the upper tail, so a head whose variances are globally miscalibrated can still produce
 1113 a useful ordering of the candidates that actually get acquired. The two axes should therefore be
 1114 read together; the calibration plot is informative about the posterior, but it is not a reliable proxy for
 1115 downstream optimization performance.

1116 **H Batch Bayesian Optimization**

1117 This appendix expands on the batch results in §3.2. We fix the total acquisition budget at 768
 1118 candidates and sweep batch sizes $k \in \{24, 48, 96, 192, 384\}$ over $T \in \{32, 16, 8, 4, 2\}$ rounds. At
 1119 the practical $k=96$ setting (one 96-well plate per round), we report the Pareto frontier of NBSF
 1120 against Top-5% count both pooled and split by modality, then show the round-by-round curves
 1121 underlying the scaling result of Figure 3.

1122 **H.1 Detailed Results**

1123 Figure 13 pools across all 61 tasks at $k=96$, and Figure 14 disaggregates by modality. The qualitative
 1124 picture closely matches the sequential setting (§G.1): VBLL and MLP_Ensemble occupy the upper-
 1125 right frontier with MLP_Greedy adjacent, the random-forest, linear, and GP variants form the middle
 1126 tier, and Random/Zero-Shot trail. The main batching effect is on coverage: the Top-5% axis
 1127 compresses for every method relative to sequential, since the surrogate cannot incorporate within-
 1128 batch labels before committing all k picks. Figure 15 shows the round-by-round trajectories at each
 1129 k . Smaller batches reach a given NBSF level in fewer cumulative candidates, and the gap on Top-5%
 1130 count remains visible at the end of the budget; the curves converge slowly as k grows, consistent with
 1131 elite-set coverage being feedback-round-limited rather than candidate-budget-limited.

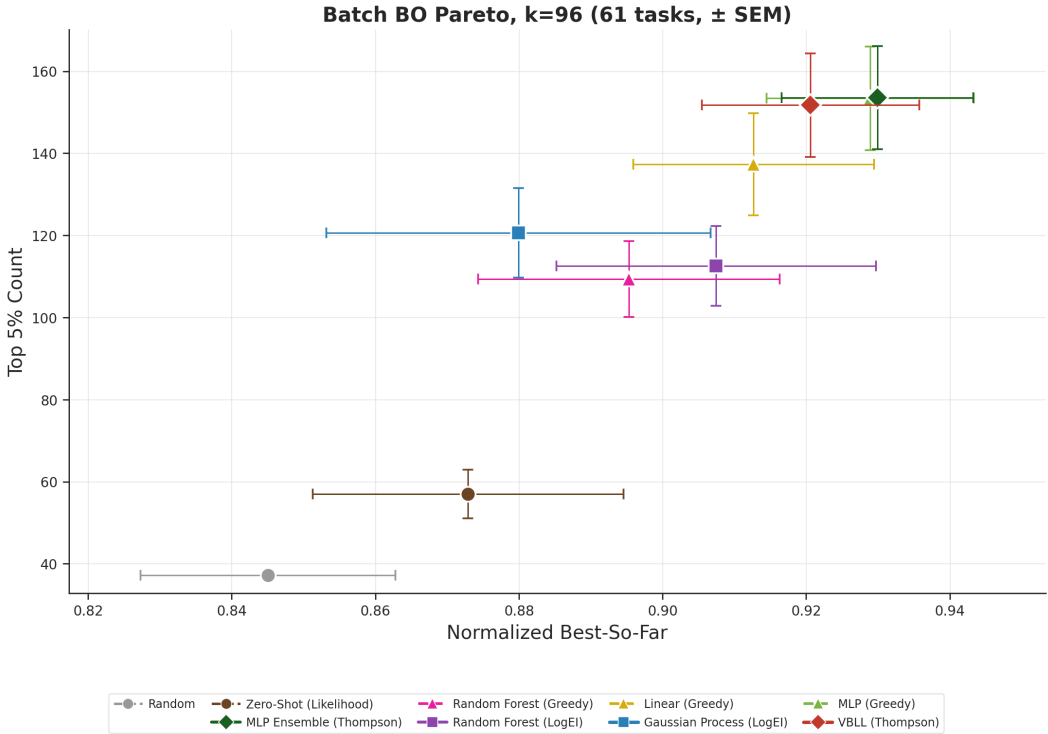


Figure 13: **Pareto frontier of NBSF vs. Top-5% count, pooled across all 61 tasks (batch, $k=96$, $T=8$).** Each marker is one method aggregated over tasks and seeds; error bars are ± 1 SEM. The frontier ordering matches the sequential setting (Figure 10); absolute Top-5% counts are lower because batches commit k picks per posterior update.

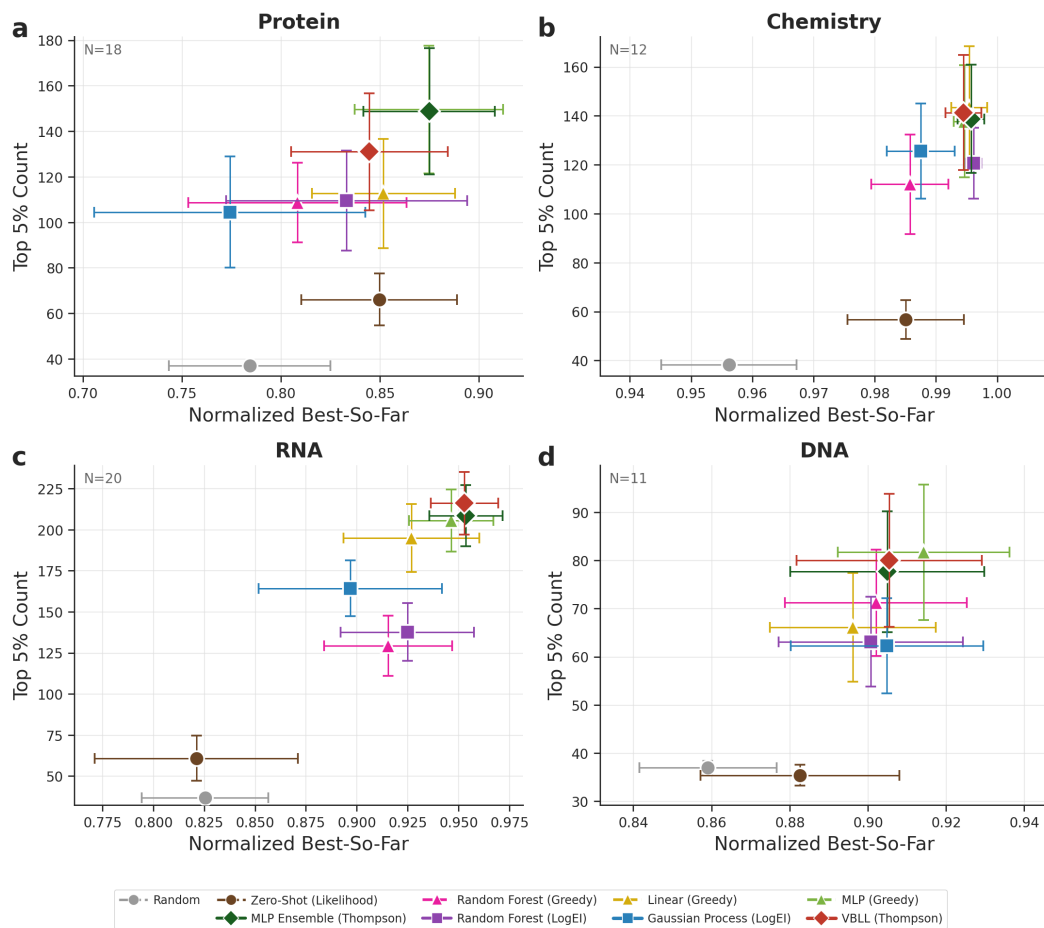


Figure 14: **Per-modality Pareto frontier at $k=96$, $T=8$.** Same axes as Figure 13 split by modality. VBLL leads on chemistry, RNA, and DNA; MLP_Ensemble and MLP_Greedy are most competitive on protein, mirroring the sequential per-modality picture.

Batch BO (k=96) — aggregate by modality (± SEM)

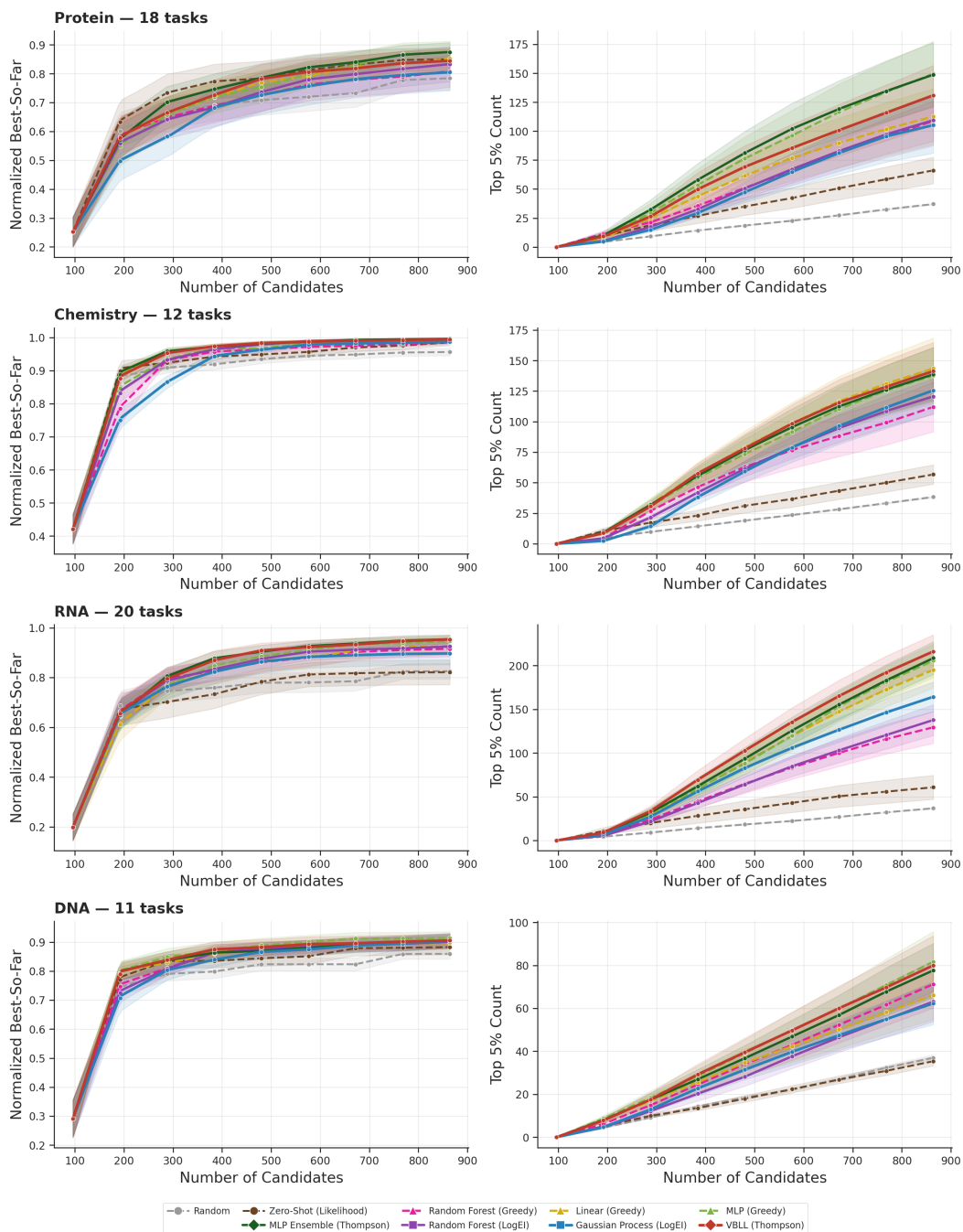


Figure 15: **Round-by-round batch curves at fixed total budget.** NBSF (top) and Top-5% count (bottom) vs. cumulative candidates measured for $k \in \{24, 48, 96, 192, 384\}$. Aggregated over 61 tasks, 9 methods, and 5 seeds. Bands are ± 1 SEM. Smaller k dominates throughout on Top-5% and converges with larger k on NBSF by end-of-budget.

1132 I Comparison to LLM

1133 This appendix documents reproducibility details for the LLM-ICL baseline used in §3.3. None of the
1134 choices below are load-bearing for the paper’s claim, but reviewers and reimplementors will want
1135 them.

1136 **Prompt structure.** The prompt is fully pool-stable: the candidate pool is rendered once, sorted
1137 by candidate ID, and never re-permuted. This is critical for serving cost. vLLM’s prefix cache
1138 survives round to round, only the appended selection row needs cold prefill at each step, and round 2
1139 wall-clock is roughly an order of magnitude faster than round 1. The prompt contains six sections in
1140 fixed order:

- 1141 • **## Pool:** stable, listing (*id*, sequence) for unobserved candidates and never the fitness label
1142 (the central no-leakage invariant).
- 1143 • **## Initial Screen:** all 96 below-median initial candidates with their measured fitness,
1144 sorted by the task metric.
- 1145 • **## Selections:** append-only table of (*id*, sequence, fitness) tuples for rounds 1 through
1146 $t-1$.
- 1147 • **## Tested IDs:** append-only list, used by the model to filter its own candidate set.
- 1148 • **## Progress:** summary of best-so-far, the top-5 selections, and the number of remaining
1149 candidates.
- 1150 • **## Task:** defines the two BO objectives (NBSF and top-5% count) and the balanced ex-
1151 ploit/explore framing.

1152 **Coverage-aware framing.** The **## Task** block defines the two objectives and instructs the model in
1153 language that does not bias toward either strategy: “*Both exploitation (refining within a known high-*
1154 *fitness region) and exploration (probing a distinct cluster) can be productive [...] balance the two*
1155 *based on what the data suggests.*” The model is then required to emit a brief **## diversity_check ##**
1156 block stating which strategy it has chosen and why, followed by **## selected_id ## N**. The
1157 diversity-check block is a forcing function for the model to commit to a stated strategy each round,
1158 which in turn anchors its selection.

1159 **Contamination controls.** The task-description block never names the source paper, the bench-
1160 mark suite, the original authors, or the assay year, and identifies only the underlying biology in
1161 domain-generic terms: “*GB1 protein B₁-domain*”; “*MPRA for human enhancers in the SK-N-SH*
1162 *neuroblastoma line*”; “*synthetic 5' UTR plus start-codon library driving an mRNA reporter*”; “*small-*
1163 *molecule agonists of PPAR γ , a ligand-activated nuclear receptor*”. Specific receptor or library
1164 identities are retained because they are pre-existing public knowledge that any domain expert would
1165 also have, and excluding them would render the task uninterpretable to the model. We did not
1166 implement an additional purge of the pretraining corpus: the LLM may have seen the underlying
1167 datasets indirectly through derivative literature, which is the residual contamination concern noted in
1168 §3.3.

1169 **Strict parser.** We use a strict parser that strips the `<think>...</think>` block and then the
1170 **## diversity_check ##** block before scanning for **## selected_id ## N**. If the delimiter is
1171 missing, the parser returns one of four failure flags (`no_delimiter`, `ambiguous`, `out_of_pool`,
1172 `tested`) rather than a silent integer match, and the round is logged as `parse_failure` for post-hoc
1173 audit. We do not use a “last integer in the response” fallback (which would silently match an integer
1174 in the diversity-check sentence or a position number in the reasoning). Failed rounds fall back to
1175 a uniform random pick from the untested pool, so parse failure does not bias the LLM toward any
1176 sequence region but does cap the upper bound on its top-5% recall. Per-task parse-success rates
1177 ranged from 84.6% (`gosai_sknsh`) to 92.2% (`ke_mrna`); detailed per-seed counts appear in Table 9.

1178 **Inference.** Inference uses vLLM with `max_model_len=262,144` tokens, `max_tokens=32,768`,
1179 FP8 KV cache, prefix caching enabled, and a single H100 GPU under FP8 weights (35B-parameter
1180 MoE, 3B active per token). Sampling follows the Qwen3.6-recommended thinking-mode settings:
1181 `temperature=1.0`, `top_p=0.95`, `top_k=20`, `presence_penalty=1.5`. A multi-seed wrapper
1182 loads model weights once and warms up DeepGEMM kernels once, then resets the prefix cache and
1183 BO state between seeds (no weight reload, no warmup repeat).

1184 **Task selection rationale.** We selected one task per modality from the 61-task benchmark using three
 1185 criteria: (i) the task is in our benchmark and has a competitive surrogate baseline (so any LLM gap
 1186 reflects a hard regime, not a poorly-tuned baseline); (ii) the pool is small enough to fit in context
 1187 with $\geq 25\text{K}$ reasoning tokens of headroom at round 100 under the empirical Qwen tokenizer; (iii)
 1188 within (ii), we prefer the largest pool that fits, since the comparison is most informative when the
 1189 LLM is given as much context as it can use. Long-protein and long-RNA tasks (over 400 residues
 1190 or nucleotides) overflow the 256K-token window once the pool exceeds a few hundred candidates
 1191 and were dropped from the LLM-eligible cohort entirely. For chemistry (pparg, $N=2,349$) and
 1192 RNA (ke_mrna, $N=5,559$) the entire dataset fits in context. For protein and DNA the full 10K-row
 1193 datasets are too large, so we deterministically subsample (seed 12345, applied identically across
 1194 methods on the LLM-side runs) to $N=1,500$ for gb1 and $N=2,100$ for gosai_sknsh.

1195 **Pool-size confound on top-5% count.** The top-5% threshold is the 95th percentile of the full task
 1196 dataset, computed identically for the LLM and the SciOpt baselines. For the matched-pool tasks
 1197 (pparg, ke_mrna) both methods see the same candidates, so absolute count is directly comparable.
 1198 For gb1 and gosai_sknsh the LLM sees a subsample and SciOpt sees the full dataset, so the LLM
 1199 has fewer top-5% candidates available in its visible pool (Table 8). On these two tasks the absolute-
 1200 count comparison mechanically caps the LLM’s top-5% count by roughly $5\times$, so we additionally
 1201 check the recall rate (hits divided by available top-5% in each method’s pool); the LLM still trails on
 1202 coverage rate everywhere except gb1.

Table 8: Pool sizes and top-5% availability per task per method.

Task	LLM-ICL		SciOpt	
	Pool Size	Top-5% avail	Pool Size	Top-5% avail
gb1	1,500	79	10,000	500
pparg	2,349	131	2,349	131
ke_mrna	5,559	277	5,559	277
gosai_sknsh	2,100	98	10,000	500

1203 **Per-seed results.** Table 9 reports NBSF, top-5% count, and parse-failure count for each (task, seed)
 1204 at round 100. Run-to-run variance is large for LLM-ICL on every task, often comparable to or
 1205 exceeding the mean. On gb1 two of five seeds reach NBSF 0.254 and 0.358 while the other three
 1206 reach 1.000, reflecting the mode-collapse pattern discussed in §3.3.

Table 9: Per-seed final-round (round 100) results for LLM-ICL on the four paper-slate tasks. T5 = top-5% count; PF = parse-failure count out of 100 rounds.

Seed	gb1			pparg			ke_mrna			gosai_sknsh		
	NBSF	T5	PF	NBSF	T5	PF	NBSF	T5	PF	NBSF	T5	PF
1	1.000	28	10	0.830	18	10	0.643	4	6	0.420	5	11
2	1.000	18	9	0.780	9	10	0.752	3	5	1.000	3	18
3	0.254	5	12	0.774	6	24	0.894	41	8	0.420	7	12
4	1.000	25	17	0.934	29	6	0.626	1	15	0.539	8	16
5	0.358	8	11	1.000	37	8	0.685	4	5	0.413	4	20
Mean	0.722	16.8	11.8	0.864	19.8	11.6	0.720	10.6	7.8	0.558	5.4	15.4
S.d.	0.382	10.1	3.1	0.100	13.1	7.4	0.109	17.0	4.1	0.252	2.1	3.8

1207 **Random-baseline note for unmatched-pool columns.** For gb1 and gosai_sknsh, the random
 1208 baseline in Figure 4 is computed on the SciOpt full pool ($N=10,000$), not the LLM’s subsample. A
 1209 random search on the LLM’s pool would have a slightly different baseline NBSF, since both pools
 1210 share the same biology but differ in absolute fitness range. For the matched-pool tasks (pparg,
 1211 ke_mrna) the random baseline is directly comparable to the LLM.

1212 **Reasoning examples.** Figures 16–19 display one structured response per task from rounds
 1213 where LLM-ICL selected a global top-5% candidate. Each example contains the model’s full
 1214 `## reasoning ##`, `## diversity_check ##`, and `## selected_id ##` blocks (the `<think>` chain-
 1215 of-thought is omitted since the model emits an explicit final reasoning section). The reasoning
 1216 is consistently grounded in domain biology (FMN-binding J-helix in GB1; tetralin-carboxylate

1217 scaffolds with halogen substitution for PPAR γ ; conserved core motifs with 5' UTR variation for
 1218 ke_mrna; high-GC neuronal-enhancer architecture for gosai_sknsh), which is consistent with the
 1219 paradigm-appeal claim in Liu et al. 2024 [55]. Despite this reasoning quality, the run-level metrics in
 1220 Table 9 show that grounded chains-of-thought do not translate to surrogate-competitive performance
 1221 under tight measurement budgets.

LLM-ICL on GB1 — One Round of In-Context Learning

Owen3.0-35B-A3B-FP8 · round 38 of 100 · selected ID 737 → fitness 4.916, top-5%

STEP 1 · Optimization setup

You are an expert protein biochemist performing iterative experimental optimization:

Your goal: discover high-performing amino acid sequence variants. We score you on TWO objectives, and both matter:

- **Best-so-far**** – the single highest Fitness you find across all selections.
- **Coverage**** – how many of the top 5% of the full library you discover (top_5_count).

Both exploitation (refining within a known high-fitness region) and exploration (probing a distinct cluster) can be productive. Exploitation is most useful when a high-fitness pattern in your selections hasn't been fully mapped yet; exploration is most useful when your top selections cluster narrowly and other high-fitness regions likely remain undiscovered. Balance the two based on what the data suggests. After each selection you will receive the measured fitness. You have a budget of 100 rounds.

About this experiment:
 GB1 protein B1-domain. Fitness measures binding affinity to IgG-Fc.

About the data format:
 Sequences use amino acid single-letter code. Key properties:
 Hydrophobic (A, V, L, I, W, F, M, P), polar (S, T, Y, N, Q),
 charged (D, E, K, R, H), special (C disulfide, G flexible, P rigid).

Reasoning budget guidance: keep your private reasoning concise and decision-oriented. ****Never re-write a candidate's full sequence in your reasoning – refer to candidates ONLY by their integer ID number.**** The pool is shown above; you can locate any ID instantly. Verbose enumeration in <think> wastes the response budget and risks truncating before the final ****selected id**** block. Aim for s2-3 K tokens of reasoning before committing.

STEP 2 · Iteration state (abridged for display)

```
## Pool (1404 total candidates, sorted by ID)
## Initial Screen (Round 0 – 96 candidates from the low-fitness region, sorted by Fitness)
| ID | Sequence | Fitness |
|----|-----|-----|
| 57 | MOYKLLNGKTLKGETTTEAVDAATAEKVFKQYANDNGRFRKENTYDDATK... | 0.000 |
| 14 | MOYKLLNGKTLKGETTTEAVDAATAEKVFKQYANDNGRFRKENTYDDATK... | 0.000 |
| 18 | MOYKLLNGKTLKGETTTEAVDAATAEKVFKQYANDNGRFRKENTYDDATK... | 0.000 |
| 73 | MOYKLLNGKTLKGETTTEAVDAATAEKVFKQYANDNGRFRKENTYDDATK... | 0.000 |
| 94 | MOYKLLNGKTLKGETTTEAVDAATAEKVFKQYANDNGRFRKENTYDDATK... | 0.000 |
Round 0 distribution: mean=0.000, median=0.000, min=0.000, max=0.000, stdev=0.000.

## Selections (Rounds 1-38)
| Round | ID | Sequence | Fitness |
|----|-----|-----|-----|
| 2 | 200 | MOYKLLNGKTLKGETTTEAVDAATAEKVFKQYANDNGRFRKENTYDDATK... | 0.000 |
| 3 | 196 | MOYKLLNGKTLKGETTTEAVDAATAEKVFKQYANDNGRFRKENTYDDATK... | 0.000 |
| 4 | 1150 | MOYKLLNGKTLKGETTTEAVDAATAEKVFKQYANDNGRFRKENTYDDATK... | 0.000 |
[... +33 more selections ...]

## Tested IDs (37 – exclude from your selection)
96, 100, 106, 111, 130, 141, 143, 200, ... (37 total)

## Progress
Round: 38 / 100
Untested candidates remaining: 1367
Best Fitness so far: 1.000 (ID 833, Round 22)
Selections so far: n=36, mean=0.049, stdev=0.179
Top-5 of your selections (Top-5 table elided for figure; see ## Selections above) –

## Task
Analyze the patterns in your selections (and the top of Round 0). Consider which sequence features correlate with high Fitness.

**select ONE candidate ID from '## Pool' whose ID is NOT in '## Tested IDs'.** That candidate should be:
1. **Likely high-Fitness** – features consistent with your top selections.
2. **A deliberate strategic choice** – either exploitation (reinforcing a productive pattern in your top-5 selections to refine an under-mapped region) or exploration (probing a sequence-distinct region of sequence space).

Choose exploitation when a clear high-fitness pattern is emerging and there is room to refine it. Choose exploration when your top selections cluster narrowly and additional similar picks would yield diminishing returns. Both objectives – best-so-far AND top_5_count – benefit from the right balance.

[response format spec – see Step 3]
```

STEP 3 · Model reasoning + selection

```
## reasoning ##
Analyzing the top performers, I notice a key structural distinction: **ID 833** (fitness 1.0) uniquely has the pattern 'ANDNGRAEEWYDDATKFTFTHTE' – specifically, it carries a rare 'KFTFTH' variant (position 37 shows H instead of the more common E/T/N). Meanwhile, many moderately performing sequences share 'KTFTE' or 'KFTFH' but differ in their central motif.

The critical variable region spans approximately positions 31-34 ('XXE WTY'). My highest performers cluster around motifs like 'RAEE', 'FWLE', 'SDDE', 'VLQE'. These all contain polar or charged side chains (E, D) which likely contribute to IgG-Fc binding surface complementarity.

My current coverage is poor – I've only tested 37 of 1404 candidates despite needing to find top 5% (~70 sequences). With one sequence hitting 1.0 and another at 0.42, there's clearly under-mapped space. Rather than exploit the narrow 'KFTFH' region further, I should probe nearby untested motifs with similar physicochemical properties.

I'll select **ID 737** (ANDNGWDNEWYDDATKFTFPTTE) – this carries 'WDNE' motif (aromatic W + polar D/N), which structurally mirrors features of my top performers. While its 'KFTFP' variant differs slightly from the best performer's 'KFTFH', the motif composition in the critical 31-34 region is promisingly similar to the successful 'RAEE'/'FWLE'/'SDDE' classes.

## diversity check ##
This pick is **exploration** – probing a different motif region (WDNE vs RAEE/FWLE/SDDE) while maintaining favorable physicochemical properties (polar/aromatic character), targeting a largely unexplored cluster in sequence space rather than reinforcing the narrow KFTFH pattern of our single best hit.

## selected id ##
737
```

STEP 4 · Measured fitness returned to the model (next round's prompt)

After Step 3, the B0 loop measures the fitness of the chosen variant and concatenates the measurement onto the running ****Selections**** block at the start of round 39:

Round	ID	Sequence	Fitness
38	737	MOYKLLNGKTLKGETTTEAVDAATAEKVFKQYANDNGRFRKENTYDDATKFTFIT...	0.080

That single row is the entire feedback signal – no surrogate, no acquisition function. The model's only learning channel is re-reading its own past selections and their measured fitness.

Figure 16: Reasoning example: GB1 (Protein). Round 38 of seed 2 on the gb1_1500 pool. LLM-ICL selects ID 737, fitness 4.916 (top-5%). The model identifies physicochemical motif clusters (RAEE / FWLE / SDDE) shared across high-fitness sequences and explicitly chooses to probe a parallel WDNE motif as exploration of an under-mapped region of sequence space.

41

LLM-ICL on PPAR γ — One Round of In-Context Learning

Owen3.6-35B-A3B-FP8 · round 73 of 100 · selected ID 1079 → pIC₅₀ 1.745, global max

STEP 1 · Optimization setup

You are an expert medicinal chemist performing iterative experimental optimization.

Your goal: discover high-performing molecule variants. We score you on two objectives, and both matter:

- **Best-so-far**** — the single highest pIC₅₀ (PPAR γ) you find across all selections.
- **Coverage**** — how many of the top 5% of the full library you discover (top_5_count).

Both exploitation (refining within a known high-fitness region) and exploration (probing a distinct cluster) can be productive. Exploitation is most useful when a high-fitness pattern in your selections hasn't been fully mapped yet; exploration is most useful when your top selections cluster narrowly and other high-fitness regions likely remain undiscovered. Balance the two based on what the data suggests. After each selection you will receive the measured fitness. You have a budget of 160 rounds.

About this experiment:
Optimization of small-molecule agonists of PPAR γ (peroxisome-proliferator activated receptor γ), a ligand-activated nuclear receptor. Fitness measures pIC₅₀ (negative log of half-maximal inhibitory concentration). A carboxylate or acidic head paired with a lipophilic tail is the canonical PPAR γ -binding motif; halogen substitutions and substitution-position patterns modulate affinity.

About the data format:
Molecules are represented as SMILES strings. Aromatic rings use lowercase (c1ccccc1), branches in parentheses, double bonds with =. Functional groups, ring systems, and electronic effects drive properties.

Reasoning budget guidance: keep your private reasoning concise and decision-oriented. ****Never** rewrite a candidate's full sequence in your reasoning — refer to candidates **ONLY** by their integer ID number. ****** The pool is shown above; you can locate any ID instantly. Verbose sequence enumeration in **<think>** wastes the response budget and risks truncating before the final **** selected id **** block. Aim for s2-3 K tokens of reasoning before committing.

STEP 2 · Iteration state (abridged for display)

```
## Pool (2253 total candidates, sorted by ID)

## Initial Screen (Round 0 — 96 candidates from the low-fitness region, sorted by pIC50 (PPARγ))
ID | SMILES | pIC50 (PPARγ) |
-----|-----|-----|
68 | COC1OC2CC(O)C3C(C2C(=O)NIC(=O)NC1CCCC1)OC1=CC(O)=C1 | 0.334 |
57 | CCCOC1OC2CCC(O)C2CC(C1)CC2C1CCC1CCC1CCC1(CO)C1=CC(O)C2= | 0.328 |
59 | Cn1c(COC2CCCC(C(=O)O)C2)nc2cccc(O)C3CC(C1)CC3 | 0.328 |
44 | CCCOC1CCCC(C(C)C(C(=O)O)CC1)CC(C(=O)O)C1CCC(C(F)(F)F)CC1 | 0.324 |
73 | CCCCCC(C)C1CC2CCCC(NS1=O)C1CCC(C(C)C(C)C)CC2 | 0.319 |
Round 0 distribution: mean=0.239, median=0.252, min=0.092, max=0.334, stdev=0.060.

## Selections (Rounds 1-73)
Round | ID | SMILES | pIC50 (PPARγ) |
-----|-----|-----|
2 | 966 | O=C(O)[C@H](C)C1CCCC1O1CCCC1C2CCCC2(C(F)(F)F)CC1 | 0.118 |
3 | 514 | O=C(O)C1CCCC(O)C1C2CCCC(C(F)(F)F)CC1C2CCCC(C) | 0.297 |
4 | 1789 | ClOC1C2CCCC(C1)CC2)nc1CCOC1CCCC(C)C(=O)N(C)C1 | 0.212 |
[... +68 more selections ...]

## Tested IDs (72 — exclude from your selection)
106, 109, 118, 127, 136, 152, 168, 171, ... (72 total)

## Progress
Round: 73 / 160
Untested candidates remaining: 2181
Best pIC50 (PPARγ) so far: 0.741 (ID 1617, Round 71)
Selections so far: n=71, mean=0.324, stdev=0.162
Top-5 of your selections (Top-5 table elided for figure; see ## Selections above) —

## Task
Analyze the patterns in your selections (and the top of Round 0). Consider which smiles features correlate with high pIC50 (PPARγ).

**Select ONE candidate ID from '## Pool' whose ID is NOT in '## Tested IDs'.** That candidate should be:
1. **Likely high-pIC50 (PPARγ)** — features consistent with your top selections.
2. **A deliberate strategic choice** — either exploitation (reinforcing a productive pattern in your top-5 selections to refine an under-mapped region) or exploration (probing a sequence-distinct region of molecular space).

Choose exploitation when a clear high-fitness pattern is emerging and there is room to refine it. Choose exploration when your top selections cluster narrowly and additional similar picks would yield diminishing returns. Both objectives — best-so-far AND top_5_count — benefit from the right balance.

[response format spec — see Step 3]
```

STEP 3 · Model reasoning + selection

reasoning

My top performers share two key structural motifs:

- **Fused bicyclic scaffold with carboxylic acid**** — IDs 1367 (0.671), 1714 (0.626), 1453 (0.576) all follow the pattern `^CCC(C(=O)O)C1CCC2cc(...)^ccc21`: a tetralin-like fused system with a 3-methylbutanoic acid side chain, connected via an ether linker (OCC) to a heteroaromatic ring (oxazole/thiazole) bearing a distal phenyl substituent.
- **Variation in the distal phenyl group****: unsubstituted phenyl gave 0.671 (ID 1367), unsubstituted ring gave 0.626 (ID 1714), CF₃-phenyl gave 0.576 (ID 1453). The trend suggests that moderate lipophilicity/electronic modulation on the distal group is beneficial but excessive hydrophobicity (CF₃) may penalize.

ID 1079 (`^CCC(C(=O)O)C1CCC2cc(O)CC3ccc(-c4ccc(C1)cc4C)oc3C)ccc21`) is an ****exploitation**** pick using the identical scaffold, but substituting the distal phenyl with a dichloro-phenyl. Halogen substitutions are known to fine-tune PPAR γ affinity through both electronic effects and van der Waals contacts in the ligand-binding pocket. The dichloro variant probes whether increased lipophilicity/bulk beyond phenyl improves affinity without crossing into the penalty region observed with CF₃.

diversity check

Exploitation: This candidate uses the exact same fused-bicyclic-tetralin scaffold as my top-3 performers, varying only the distal phenyl substituent from H/C/H₃ to dichloro. This refines the under-mapped parameter space of halogen substitution within the proven high-fitness pattern.

selected id

1079

STEP 4 · Measured fitness returned to the model (next round's prompt)

After Step 3, the BO loop measures the pIC₅₀ of the chosen variant and concatenates the measurement onto the running **** Selections **** block at the start of round 74:

Round	ID	Sequence	Fitness
73	1079	CCC(C(=O)O)C1CCC2cc(O)CC3ccc(-c4ccc(C1)cc4C)oc3C)ccc21	0.437

That single row is the entire feedback signal — no surrogate, no acquisition function. The model's only learning channel is re-reading its own past selections and their measured fitness.

Figure 17: Reasoning example: PPAR γ Binding (Chemistry). Round 73 of seed 1 on the pparg_full pool. LLM-ICL selects ID 1079, pIC₅₀ 1.745 (global maximum). The model identifies a fused-bicyclic-tetralin scaffold with a carboxylate head shared by its top performers, and exploits the under-mapped halogen-substitution dimension on the distal aromatic group.

42

LLM-ICL on ke_mrna — One Round of In-Context Learning

Owen3.6-35B-A3B-FP8 · round 9 of 100 · selected ID 4244 → DMS score 5.21, top-5%

STEP 1 · Optimization setup

You are an expert RNA biologist performing iterative experimental optimization.

Your goal: discover high-performing RNA sequence variants. We score you on TWO objectives, and both matter:

- **Best-so-far**** – the single highest mRNA Expression you find across all selections.
- **Coverage**** – how many of the top 5% of the full library you discover (top_5_count).

Both exploitation (refining within a known high-fitness region) and exploration (probing a distinct cluster) can be productive. Exploitation is most useful when a high-fitness pattern in your selections hasn't been fully mapped yet; exploration is most useful when your top selections cluster narrowly and other high-fitness regions likely remain undiscovered. Balance the two based on what the data suggests. After each selection you will receive the measured fitness. You have a budget of 100 rounds.

About this experiment:
Optimization of a synthetic 5' UTR + start-codon context library driving an mRNA reporter. Fitness measures relative protein output (translation efficiency × mRNA stability). Kozak-context strength, 5'-end secondary structure, upstream open reading frames (uORFs), and start-codon accessibility modulate expression.

About the data format:
RNA sequences use nucleotide single-letter code (A, C, G, U). Base pairing (A-U, G-C, G-U wobble) drives secondary structure. Sequence context affects folding, stability, and catalytic activity.

Reasoning budget guidance: keep your private reasoning concise and decision-oriented. ****Never re-write a candidate's full sequence in your reasoning – refer to candidates ONLY by their integer ID number.**** The pool is shown above; you can locate any ID instantly. Verbose sequence enumeration in <think> wastes the response budget and risks truncating before the final ****selected id**** block. Aim for <2-3 K tokens of reasoning before committing.

STEP 2 · Iteration state (abridged for display)

```
## Pool (5463 total candidates, sorted by ID)

## Initial Screen (Round 0 – 96 candidates from the low-fitness region, sorted by mRNA Expression)
| ID | Sequence | mRNA Expression |
|----|-----|-----|
| 30 | AGUUGAGAGGGAGGCUCCAGCACAGUGAAUUGGACUAGGGCAGAGCA | 0.065 |
| 60 | AGUUGAGAGGGAGGCUCCAGCACAGUGAAUUGGACAGAGGGCAGAGCA | 0.063 |
| 89 | AGUUGAGAGGGAGGCUCCAGCACAGUGAAUUGGACAGAGGGCAGAGCA | 0.062 |
| 0 | AGUUGAGAGGGAGGCUCCAGCACAGUGAAUUGGACAGAGGGCAGAGCA | 0.062 |
| 43 | AGUUGAGAGGGAGGCUCCAGCACAGUGAAUUGGACAGAGGGCAGAGCA | 0.062 |
Round 0 distribution: mean=0.014, median=0.006, min=0.000, max=0.065, stdev=0.018.

## Selections (Rounds 1-9)
| Round | ID | Sequence | mRNA Expression |
|----|----|-----|-----|
| 2 | 2443 | AGUUGAGAGGGAGGCUCCAGCACAGUGAAUUGGACAGAGGGCAGAGCA | 0.125 |
| 3 | 3967 | AGUUGAGAGGGAGGCUCCAGCACAGUGAAUUGGACAGAGGGCAGAGCA | 0.038 |
| 4 | 4242 | AGUUGAGAGGGAGGCUCCAGCACAGUGAAUUGGACAGAGGGCAGAGCA | 0.245 |
[... +4 more selections ...]

## Tested IDs (0 – exclude from your selection)
2443, 3317, 3967, 4242, 4243, 4245, 4247, 4289

## Progress
Round: 9 / 100
Untested candidates remaining: 5455
Best mRNA Expression so far: 0.305 (ID 4243, Round 6)
Selections so far: n=7, mean=0.229, stdev=0.098
Top-5 of your selections (Top-5 table elided for figure; see ## Selections above) –

## Task
Analyze the patterns in your selections (and the top of Round 0). Consider which sequence features correlate with high mRNA Expression.

**Select ONE candidate ID from '## Pool' whose ID is NOT in '## Tested IDs'.** That candidate should be:
1. **Likely high-mRNA Expression** – features consistent with your top selections.
2. **A deliberate strategic choice** – either exploitation (reinforcing a productive pattern in your top-5 selections to refine an under-mapped region) or exploration (probing a sequence-distinct region of sequence space).

Choose exploitation when a clear high-fitness pattern is emerging and there is room to refine it. Choose exploration when your top selections cluster narrowly and additional similar picks would yield diminishing returns. Both objectives – best-so-far AND top_5_count – benefit from the right balance.

[response format spec – see Step 3]
```

STEP 3 · Model reasoning + selection

```
## reasoning ##
Analysis of the top 5 performers (IDs 4242-4289) reveals a highly consistent core motif: 'CUCCAGCACAGUGAAUUGGACAGAGGGCAGAGCAA'. All five share this exact backbone. They differ primarily in their 5' termini, with successful variants starting with 'AGUU' followed by diverse intermediate sequences (e.g., 'UGAGAGCGG', 'CUUCUCGGG', 'UGAACACGG'). The consensus indicates that the core sequence is essential for high expression, while the 5' region can tolerate significant variation—likely affecting ribosome accessibility, secondary structure, or initiation efficiency.

Given only 7 total selections and a clear signal emerging around this conserved core, exploitation is the right move. By selecting another candidate that preserves the winning core but varies the 5' region, I can map the neighborhood of this high-fitness peak more thoroughly. Candidate 4244 matches the core perfectly ('CUCCAGCACAGUGAAUUGGACAGAGGGCAGAGCAA') and has a GC-rich 5' start ('AGUUGAGAGGG') distinct from the 5' variants already tested in rounds 2-8. This targeted exploitation should either identify a better local optimum or confirm the boundaries of the current peak.

## diversity check ##
Exploitation — selecting a candidate with the same proven core motif but a different 5' sequence to refine the mapped high-fitness region before broader exploration.

## selected_id ##
4244
```

STEP 4 · Measured fitness returned to the model (next round's prompt)

After Step 3, the B0 loop measures the dms score of the chosen variant and concatenates the measurement onto the running ****Selections**** block at the start of round 10:

Round	ID	Sequence	Fitness
9	4244	AGUUGAGAGGGAGGCUCCAGCACAGUGAAUUGGACAGAGGGCAGAGCAA	0.298

That single row is the entire feedback signal – no surrogate, no acquisition function. The model's only learning channel is re-reading its own past selections and their measured fitness.

Figure 18: Reasoning example: mRNA Splicing (RNA). Round 9 of seed 3 on the ke_mrna_full pool. LLM-ICL selects ID 4244, DMS score 5.21 (top-5%). The model identifies a conserved 36-nt core motif (CUCCAGCACAGUGAAUUGGACAGAGGGCAGAGCAA) preserved across the top performers and exploits within the 5'-end variation neighborhood of the existing high-fitness peak.

43

LLM-ICL on gosai_sknsH — One Round of In-Context Learning

Owen3.6-35B-A3B-FB8 · round 67 of 100 · selected ID 1106 → STARR-seq 2.93, top-5 %

STEP 1 · Optimization setup

You are an expert regulatory genomicist performing iterative experimental optimization.

Your goal: discover high-performing DNA sequence variants. We score you on two objectives, and both matter:

- **Best-so-far**** – the single highest Enhancer Activity (SK-N-SH) you find across all selections.
- **Coverage**** – how many of the top 5% of the full library you discover (top_5_count).

Both exploitation (refining within a known high-fitness region) and exploration (probing a distinct cluster) can be productive. Exploitation is most useful when a high-fitness pattern in your selections hasn't been fully mapped yet; exploration is most useful when your top selections cluster narrowly and other high-fitness regions likely remain undiscovered. Balance the two based on what the data suggests. After each selection you will receive the measured fitness. You have a budget of 160 rounds.

About this experiment:
 MPPRA for human enhancers in SK-N-SH neuroblastoma cells. Fitness measures STARR-seq signal. NEUROD, ASCL1, and POU3F2 motifs are characteristic of neuronal enhancers.

About the data format:
 DNA sequences use nucleotide single-letter code (A, T, C, G). Base pairing (A-T, G-C) drives structure. Motifs like TATA boxes, GC content, and transcription factor binding sites affect function.

Reasoning budget guidance: keep your private reasoning concise and decision-oriented. ****Never re-write a candidate's full sequence in your reasoning – refer to candidates ONLY by their integer ID number.**** The pool is shown above; you can locate any ID instantly. Verbose sequence enumeration in `<think>` wastes the response budget and risks truncating before the final `## selected id ##` block. Aim for s2-3 K tokens of reasoning before committing.

STEP 2 · Iteration state (abridged for display)

```
## Pool (2004 total candidates, sorted by ID)

## Initial Screen (Round 0 – 96 candidates from the low-fitness region, sorted by Enhancer Activity (SK-N-SH))
| ID | Sequence | Enhancer Activity (SK-N-SH) |
|----|-----|-----|
| 93 | CTCCTGCACAGAGGACACTTGTATCCTTATGGATTGATGAGT... | 0.059 |
| 85 | GGGTGTGGTGGAGGCTGGTATGGTGGCTGGGGTGGGGTGGAGAC... | 0.047 |
| 24 | AAATTAGCGGGGCTGGTGGGGGGCCCTGATGCCAGCTACTGGAG... | 0.046 |
| 86 | TTTTITTTTTTAAATAGCCAAGTAAGCCAGGATGTTGGCTATGCC... | 0.046 |
| 84 | CTAAKCCGTACCCCATTTGGTGGCTGGGGGGGGCTGCTTTATT... | 0.046 |
Round 0 distribution: mean=0.026, median=0.026, min=0.000, max=0.050, stdev=0.013.

## Selections (Rounds 1-67)
| Round | ID | Sequence | Enhancer Activity (SK-N-SH) |
|----|-----|-----|-----|
| 2 | 122 | AAATTAAATGGATCTATTGTGAAACTTGGGCTGATGGGGCAGAC... | 0.011 |
| 3 | 103 | GCTTATTAGGTGAAGAAACTATCTCCAGGTGGACTTGGGATGGGA... | 0.004 |
| 4 | 111 | AGGATTCATACAGATTATTAGACCCAGGGCTTATATCTATGATCCA... | 0.002 |
[... +62 more selections ...]

## Tested IDs (66 – exclude from your selection)
101, 103, 111, 112, 113, 122, 151, 242, ... (66 total)

## Progress
Round: 67 / 160
Untested candidates remaining: 1938
Best Enhancer Activity (SK-N-SH) so far: 0.420 (ID 1207, Round 25)
Selections so far: n=65, mean=0.180, stdev=0.092
Top-5 of your selections (Top-5 table elided for figure; see ## Selections above) -

## Task
Analyze the patterns in your selections (and the top of Round 0). Consider which sequence features correlate with high Enhancer Activity (SK-N-SH).

**Select ONE candidate ID from '## Pool' whose ID is NOT in '## Tested IDs'.** That candidate should be:
1. **Likely high-Enhancer Activity (SK-N-SH)** – features consistent with your top selections.
2. **A deliberate strategic choice** – either exploitation (reinforcing a productive pattern in your top-5 selections to refine an under-mapped region) or exploration (probing a sequence-distinct region of sequence space).

Choose exploitation when a clear high-fitness pattern is emerging and there is room to refine it. Choose exploration when your top selections cluster narrowly and additional similar picks would yield diminishing returns. Both objectives – best-so-far AND top_5_count – benefit from the right balance.

[response format spec – see Step 3]
```

STEP 3 · Model reasoning + selection

```
## reasoning ##
Top performers (IDs 1207, 1935, 825, 2010, 1398) consistently feature high GC content, dense purine/pyrimidine tracts, and shared regulatory cores (e.g., 'GGGAGGCAGAGGTTGCAGTGAG', 'CACACAC'). These motifs strongly align with neuronal enhancer architecture expected in SK-N-SH cells. With 33 rounds remaining, exploitation is optimal: refining within this high-fitness sequence neighborhood can reveal additive context effects and potentially push activity past 0.420. ID 1106 preserves the exact high-value core modules seen in my top-5 while presenting novel upstream/downstream flanks, making it a precise target for performance calibration.

## diversity_check ##
Exploitation. This pick directly targets the high-GC/multimeric enhancer pattern dominating the top-5, aiming to refine motif spacing and flanking context rather than branching into a divergent sequence regime.

## selected_id ##
1106
```

STEP 4 · Measured fitness returned to the model (next round's prompt)

After Step 3, the B0 loop measures the starr-seq of the chosen variant and concatenates the measurement onto the running `## Selections ##` block at the start of round 68:

Round	ID	Sequence	Fitness	
67	1106	GCTAGGCGTGGTGGCTGCTGCTGTAATCCAGCTACTCGGGAGGCTGAGGTAGG...	0.182	- TOP-5%

That single row is the entire feedback signal – no surrogate, no acquisition function. The model's only learning channel is re-reading its own past selections and their measured fitness.

Figure 19: Reasoning example: Neuroblastoma CRE (DNA). Round 67 of seed 1 on the gosai_sknsH subsample. LLM-ICL selects ID 1106, STARR-seq activity 2.93 (top-5%). The model identifies a high-GC architecture with shared regulatory cores (GGGAGGCAGAGGTTGCAGTGAG, CACACAC) characteristic of neuronal enhancers in the SK-N-SH line, and exploits the motif-spacing neighborhood of its current top performers.

44

1222 J Comparison to SGPO

- 1223 • **Datasets:** All three datasets are combinatorially-complete 4-site libraries, with GB1 at
1224 positions 39/40/41/54, TrpB4 at 34/35/78/79, and PhoQ at 50/51/54/55.
- 1225 • **Initialization and Batch Size:** All methods start from the same 100-sequence cold-start on
1226 each seed. The init set is sampled uniformly from the bottom-0.5 quantile of the labeled
1227 pool’s real fitness. The total labeling budget is $t = 1000$ sequences per seed (100 init + 9
1228 acquisition rounds $\times 100$ per round), mimicking the adaptive setting in SGPO.
- 1229 • **Seeds and Confidence Intervals:** Each method performs 5 independent BO runs (seeds
1230 0–4). The confidence bound is ± 1 sample standard deviation across the 5 seeds, capturing
1231 variability across seeds.
- 1232 • **SciOpt Configuration:** We benchmark against ESM-C backbone with MLP and VLLM.
1233 We also run the random baseline for comparison.
- 1234 • **SGPO configuration.** We made no SGPO source code modifications. We use the strongest
1235 reported SGPO-style configuration: a masked diffusion language model prior $p_\theta(x)$ steered
1236 by decoupled annealing posterior sampling (DAPS), with Thompson-style selection over an
1237 ensemble of value functions. At each round, value functions are trained on all accumulated
1238 measured candidates D_{t-1} . Runs use original defaults throughout ($\alpha = 128$, $\text{num_steps} =$
1239 50 , $\text{mh_steps} = 1000$, $\text{max_dist} = 1$, $5 \times$ iteration cap).
- 1240 • **SGPO Mutation Position:** For each dataset, we set DAPS-MH’s residues field to the four
1241 combinatorially-varied positions. The residues field is upstream-supported and is DAPS’
1242 canonical scope-restriction mechanism, so this is a configuration choice.
- 1243 • **SGPO Mode Collapse: Per-seed variability in unique-candidate count:** SciOpt methods
1244 successfully acquire 100 fresh pool sequences per round, so every seed reaches identical
1245 cumulative unique counts. SGPO does not due to DAPS-MH duplicate-collapse. At the
1246 final round, the cumulative counts of unique candidates across the five seeds span 333–412
1247 on GB1 (mean 371, std 38), 771–914 on TrpB (mean 865, std 57), and 782–952 on PhoQ
1248 (mean 863, std 76). To produce a single SGPO curve per task, we plot the cross-seed
1249 mean cumulative unique candidates and place an X-marker at the end of the trajectory. The
1250 cross-seed performance CI is plotted according to the above.

1251 K Contamination audit

1252 The benchmark labels are wet-lab measurements that no pretraining corpus contains directly, so a
1253 literal (x, y) leak is implausible. The realistic risks are (a) *sequence-level* memorization, where the
1254 FM has seen the candidate x and assigns it an unusually informative likelihood, and (b) *narrative*
1255 memorization, where a chat-tuned LLM has read the assay paper and carries a prior over high-fitness
1256 motifs or scaffolds. Channel (a) is what Zero_Shot would exploit; channel (b) is what LLM-ICL
1257 would exploit.

1258 **FM-side: candidate pools vs. pretraining corpora.** Each primary FM has been pretrained on a
1259 corpus that overlaps the candidate space at the wild-type level but not at the variant level. ESM-
1260 C 600M [24] sees UniRef parents but not the combinatorial DMS variants; RNA-FM [15] sees
1261 RNACentral but not the synthetic UTR / ribozyme libraries; NTv3-650M [7] sees multi-species
1262 genomes but not the 200 bp MPRA libraries; ChemFM-1B [13] is trained on PubChem and so is the
1263 highest-overlap modality, with plausible exact-molecule overlap on the GOLLuM-derived chemistry
1264 tasks.

1265 **Implications for Zero_Shot.** Zero_Shot ranks candidates by the FM’s pseudo-likelihood without
1266 ever seeing labels (§B.2), so it is the surrogate most exposed to channel (a). If pretraining contam-
1267 ination were producing a useful fitness prior, Zero_Shot would be expected to dominate Random,
1268 especially on chemistry. Empirically it sits on or below the random-search frontier on every modality
1269 (Figure 2), consistent with the FM prior tracking sequence naturalness rather than the assay-specific
1270 target.

1271 **LLM-side: Qwen3.6 web-text overlap.** Qwen3.6-35B-A3B-FP8 [69] has a knowledge cutoff
1272 postdating every benchmark publication on our slate, so the model has plausibly read each source
1273 paper and derivative literature. We mitigate the most direct path via the prompt controls in App. I:

1274 the task block names only the underlying biology in domain-generic terms (never the source paper,
1275 benchmark, authors, or assay year), and the prompt never includes unobserved fitness labels. We
1276 did not perform a corpus purge, so this is a soft control. The empirical pattern is again inconsistent
1277 with a useful narrative leak: LLM-ICL trails learned surrogates on NBSF on every task, the gb1
1278 mode-collapse (two of five seeds end at NBSF 0.254 and 0.358) and high gosai_sknsh parse-failure
1279 rate (84.6% parse-success) match the prior-collapse failure mode documented for instruction-tuned
1280 LLMs under tight budgets [80, 101, 90], and the reasoning traces in App. I are domain-grounded but
1281 generic enough to come from background knowledge alone.

1282 **Direction of any residual leak.** Residual contamination would inflate the leaked baseline, not the
1283 SciOpt result: a more contaminated FM helps Zero_Shot, which already trails the learned surrogates,
1284 and a more contaminated LLM helps LLM-ICL, which already trails them too. The main-paper gaps
1285 are therefore conservative with respect to leakage. Two honest caveats: (i) PubChem overlap may
1286 give every method on chemistry a uniform lift, so absolute NBSF on pparg should be read as “with
1287 PubChem-overlap allowed,” and (ii) the LLM’s gb1 mode-collapse may be partly memorization-
1288 driven, which would make the favorable gb1 LLM number a less informative comparison than the
1289 other three tasks.

1290 **NeurIPS Paper Checklist**

1291 The checklist is designed to encourage best practices for responsible machine learning research,
1292 addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove
1293 the checklist: **The papers not including the checklist will be desk rejected.** The checklist should
1294 follow the references and follow the (optional) supplemental material. The checklist does NOT count
1295 towards the page limit.

1296 Please read the checklist guidelines carefully for information on how to answer these questions. For
1297 each question in the checklist:

- 1298 • You should answer [Yes], [No], or [N/A].
- 1299 • [N/A] means either that the question is Not Applicable for that particular paper or the
1300 relevant information is Not Available.
- 1301 • Please provide a short (1–2 sentence) justification right after your answer (even for [N/A]).

1302 **The checklist answers are an integral part of your paper submission.** They are visible to the
1303 reviewers, area chairs, senior area chairs, and ethics reviewers. You will also be asked to include it
1304 (after eventual revisions) with the final version of your paper, and its final version will be published
1305 with the paper.

1306 The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation.
1307 While [Yes] is generally preferable to [No], it is perfectly acceptable to answer [No] provided a
1308 proper justification is given (e.g., error bars are not reported because it would be too computationally
1309 expensive” or “we were unable to find the license for the dataset we used”). In general, answering
1310 [No] or [N/A] is not grounds for rejection. While the questions are phrased in a binary way, we
1311 acknowledge that the true answer is often more nuanced, so please just use your best judgment and
1312 write a justification to elaborate. All supporting evidence can appear either in the main paper or the
1313 supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification
1314 please point to the section(s) where related material for the question can be found.

1315 **IMPORTANT, please:**

- 1316 • **Delete this instruction block, but keep the section heading “NeurIPS Paper Checklist”,**
- 1317 • **Keep the checklist subsection headings, questions/answers and guidelines below.**
- 1318 • **Do not modify the questions and only use the provided macros for your answers.**

1319 **1. Claims**

1320 Question: Do the main claims made in the abstract and introduction accurately reflect the
1321 paper’s contributions and scope?

1322 Answer: [Yes]

1323 Justification: We highlight the three main contributions of the paper in the introduction
1324 where we state we provide a practical foundation-model driven BO framework, a systematic
1325 empirical study of this framework across 61 different experimental regression tasks, and a
1326 head to head comparison with leading AI-for-science paradigms.

1327 Guidelines:

- 1328 • The answer [N/A] means that the abstract and introduction do not include the claims
1329 made in the paper.
- 1330 • The abstract and/or introduction should clearly state the claims made, including the
1331 contributions made in the paper and important assumptions and limitations. A [No] or
1332 [N/A] answer to this question will not be perceived well by the reviewers.
- 1333 • The claims made should match theoretical and experimental results, and reflect how
1334 much the results can be expected to generalize to other settings.
- 1335 • It is fine to include aspirational goals as motivation as long as it is clear that these goals
1336 are not attained by the paper.

1337 **2. Limitations**

1338 Question: Does the paper discuss the limitations of the work performed by the authors?

1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390

Answer: [Yes]

Justification: We have a long section in the discussion regarding limitations of the paper. We mention things ranging from some of the choices made in the evaluation to limitations regarding the choice of LLM we compared against and lack of wet-lab experiments.

Guidelines:

- The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate “Limitations” section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren’t acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

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

Answer: [N/A]

Justification:

Guidelines:

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

4. Experimental result reproducibility

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

Answer: [Yes]

1391 Justification: We provide comprehensive details regarding our framework and choices along
1392 with hyperparameters throughout the paper. We will also release the code upon publication.

1393 Guidelines:

- 1394 • The answer [N/A] means that the paper does not include experiments.
- 1395 • If the paper includes experiments, a [No] answer to this question will not be perceived
1396 well by the reviewers: Making the paper reproducible is important, regardless of
1397 whether the code and data are provided or not.
- 1398 • If the contribution is a dataset and/or model, the authors should describe the steps taken
1399 to make their results reproducible or verifiable.
- 1400 • Depending on the contribution, reproducibility can be accomplished in various ways.
1401 For example, if the contribution is a novel architecture, describing the architecture fully
1402 might suffice, or if the contribution is a specific model and empirical evaluation, it may
1403 be necessary to either make it possible for others to replicate the model with the same
1404 dataset, or provide access to the model. In general, releasing code and data is often
1405 one good way to accomplish this, but reproducibility can also be provided via detailed
1406 instructions for how to replicate the results, access to a hosted model (e.g., in the case
1407 of a large language model), releasing of a model checkpoint, or other means that are
1408 appropriate to the research performed.
- 1409 • While NeurIPS does not require releasing code, the conference does require all submis-
1410 sions to provide some reasonable avenue for reproducibility, which may depend on the
1411 nature of the contribution. For example
 - 1412 (a) If the contribution is primarily a new algorithm, the paper should make it clear how
1413 to reproduce that algorithm.
 - 1414 (b) If the contribution is primarily a new model architecture, the paper should describe
1415 the architecture clearly and fully.
 - 1416 (c) If the contribution is a new model (e.g., a large language model), then there should
1417 either be a way to access this model for reproducing the results or a way to reproduce
1418 the model (e.g., with an open-source dataset or instructions for how to construct
1419 the dataset).
 - 1420 (d) We recognize that reproducibility may be tricky in some cases, in which case
1421 authors are welcome to describe the particular way they provide for reproducibility.
1422 In the case of closed-source models, it may be that access to the model is limited in
1423 some way (e.g., to registered users), but it should be possible for other researchers
1424 to have some path to reproducing or verifying the results.

1425 5. Open access to data and code

1426 Question: Does the paper provide open access to the data and code, with sufficient instruc-
1427 tions to faithfully reproduce the main experimental results, as described in supplemental
1428 material?

1429 Answer: [No]

1430 Justification: We do not upload the code and datasets as supplementary details right now as
1431 it is expensive. We plan on fully open-sourcing everything upon publication.

1432 Guidelines:

- 1433 • The answer [N/A] means that paper does not include experiments requiring code.
- 1434 • Please see the NeurIPS code and data submission guidelines ([https://neurips.cc/
1435 public/guides/CodeSubmissionPolicy](https://neurips.cc/public/guides/CodeSubmissionPolicy)) for more details.
- 1436 • While we encourage the release of code and data, we understand that this might not
1437 be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not
1438 including code, unless this is central to the contribution (e.g., for a new open-source
1439 benchmark).
- 1440 • The instructions should contain the exact command and environment needed to run to
1441 reproduce the results. See the NeurIPS code and data submission guidelines ([https://neurips.cc/
1442 public/guides/CodeSubmissionPolicy](https://neurips.cc/public/guides/CodeSubmissionPolicy)) for more details.
- 1443 • The authors should provide instructions on data access and preparation, including how
1444 to access the raw data, preprocessed data, intermediate data, and generated data, etc.

- 1445
- 1446
- 1447
- 1448
- 1449
- 1450
- 1451
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
 - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
 - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

1452 6. Experimental setting/details

1453 Question: Does the paper specify all the training and test details (e.g., data splits, hyperpa-
1454 rameters, how they were chosen, type of optimizer) necessary to understand the results?

1455 Answer: [Yes]

1456 Justification: We provide comprehensive training and test details needed to understand the
1457 results.

1458 Guidelines:

- 1459
- 1460
- 1461
- 1462
- 1463
- The answer [N/A] means that the paper does not include experiments.
 - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
 - The full details can be provided either with the code, in appendix, or as supplemental material.

1464 7. Experiment statistical significance

1465 Question: Does the paper report error bars suitably and correctly defined or other appropriate
1466 information about the statistical significance of the experiments?

1467 Answer: [Yes]

1468 Justification: We provide error bars for every figure in our experiments.

1469 Guidelines:

- 1470
- 1471
- 1472
- 1473
- 1474
- 1475
- 1476
- 1477
- 1478
- 1479
- 1480
- 1481
- 1482
- 1483
- 1484
- 1485
- 1486
- 1487
- 1488
- 1489
- The answer [N/A] means that the paper does not include experiments.
 - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
 - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
 - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
 - The assumptions made should be given (e.g., Normally distributed errors).
 - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
 - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
 - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
 - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

1490 8. Experiments compute resources

1491 Question: For each experiment, does the paper provide sufficient information on the com-
1492 puter resources (type of compute workers, memory, time of execution) needed to reproduce
1493 the experiments?

1494 Answer: [Yes]

1495 Justification: We give extensive detail on computational sources used in the appendix. We
1496 use CPUs or an H100 GPU with 80GB VRAM for all of these experiments.

1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549

Guidelines:

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

9. Code of ethics

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

Answer: [Yes]

Justification: We follow all guidelines

Guidelines:

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

10. Broader impacts

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

Answer: [Yes]

Justification: We clearly mention how this can be used for scientific discovery campaigns which would have positive societal impacts for society. These could potentially also have negative societal impacts although that is true with any scientific development, so we do not explicitly state this.

Guidelines:

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

11. Safeguards

1550 Question: Does the paper describe safeguards that have been put in place for responsible
1551 release of data or models that have a high risk for misuse (e.g., pre-trained language models,
1552 image generators, or scraped datasets)?

1553 Answer: [N/A]

1554 Justification:

1555 Guidelines:

- 1556 • The answer [N/A] means that the paper poses no such risks.
- 1557 • Released models that have a high risk for misuse or dual-use should be released with
1558 necessary safeguards to allow for controlled use of the model, for example by requiring
1559 that users adhere to usage guidelines or restrictions to access the model or implementing
1560 safety filters.
- 1561 • Datasets that have been scraped from the Internet could pose safety risks. The authors
1562 should describe how they avoided releasing unsafe images.
- 1563 • We recognize that providing effective safeguards is challenging, and many papers do
1564 not require this, but we encourage authors to take this into account and make a best
1565 faith effort.

1566 12. Licenses for existing assets

1567 Question: Are the creators or original owners of assets (e.g., code, data, models), used in
1568 the paper, properly credited and are the license and terms of use explicitly mentioned and
1569 properly respected?

1570 Answer: [Yes]

1571 Justification: We cite all the datasets we used and all of them are licensed appropriately for
1572 academic use.

1573 Guidelines:

- 1574 • The answer [N/A] means that the paper does not use existing assets.
- 1575 • The authors should cite the original paper that produced the code package or dataset.
- 1576 • The authors should state which version of the asset is used and, if possible, include a
1577 URL.
- 1578 • The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- 1579 • For scraped data from a particular source (e.g., website), the copyright and terms of
1580 service of that source should be provided.
- 1581 • If assets are released, the license, copyright information, and terms of use in the
1582 package should be provided. For popular datasets, paperswithcode.com/datasets
1583 has curated licenses for some datasets. Their licensing guide can help determine the
1584 license of a dataset.
- 1585 • For existing datasets that are re-packaged, both the original license and the license of
1586 the derived asset (if it has changed) should be provided.
- 1587 • If this information is not available online, the authors are encouraged to reach out to
1588 the asset's creators.

1589 13. New assets

1590 Question: Are new assets introduced in the paper well documented and is the documentation
1591 provided alongside the assets?

1592 Answer: [N/A]

1593 Justification:

1594 Guidelines:

- 1595 • The answer [N/A] means that the paper does not release new assets.
- 1596 • Researchers should communicate the details of the dataset/code/model as part of their
1597 submissions via structured templates. This includes details about training, license,
1598 limitations, etc.
- 1599 • The paper should discuss whether and how consent was obtained from people whose
1600 asset is used.

- 1601 • At submission time, remember to anonymize your assets (if applicable). You can either
1602 create an anonymized URL or include an anonymized zip file.

1603 **14. Crowdsourcing and research with human subjects**

1604 Question: For crowdsourcing experiments and research with human subjects, does the paper
1605 include the full text of instructions given to participants and screenshots, if applicable, as
1606 well as details about compensation (if any)?

1607 Answer: [N/A]

1608 Justification:

1609 Guidelines:

- 1610 • The answer [N/A] means that the paper does not involve crowdsourcing nor research
1611 with human subjects.
- 1612 • Including this information in the supplemental material is fine, but if the main contribu-
1613 tion of the paper involves human subjects, then as much detail as possible should be
1614 included in the main paper.
- 1615 • According to the NeurIPS Code of Ethics, workers involved in data collection, curation,
1616 or other labor should be paid at least the minimum wage in the country of the data
1617 collector.

1618 **15. Institutional review board (IRB) approvals or equivalent for research with human**
1619 **subjects**

1620 Question: Does the paper describe potential risks incurred by study participants, whether
1621 such risks were disclosed to the subjects, and whether Institutional Review Board (IRB)
1622 approvals (or an equivalent approval/review based on the requirements of your country or
1623 institution) were obtained?

1624 Answer: [N/A]

1625 Justification:

1626 Guidelines:

- 1627 • The answer [N/A] means that the paper does not involve crowdsourcing nor research
1628 with human subjects.
- 1629 • Depending on the country in which research is conducted, IRB approval (or equivalent)
1630 may be required for any human subjects research. If you obtained IRB approval, you
1631 should clearly state this in the paper.
- 1632 • We recognize that the procedures for this may vary significantly between institutions
1633 and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the
1634 guidelines for their institution.
- 1635 • For initial submissions, do not include any information that would break anonymity (if
1636 applicable), such as the institution conducting the review.

1637 **16. Declaration of LLM usage**

1638 Question: Does the paper describe the usage of LLMs if it is an important, original, or
1639 non-standard component of the core methods in this research? Note that if the LLM is used
1640 only for writing, editing, or formatting purposes and does *not* impact the core methodology,
1641 scientific rigor, or originality of the research, declaration is not required.

1642 Answer: [Yes]

1643 Justification: We baseline our methodology against LLM In-Context Learning for scientific
1644 applications. We use an open-weight model and give thorough detail on how we prompt it.

1645 Guidelines:

- 1646 • The answer [N/A] means that the core method development in this research does not
1647 involve LLMs as any important, original, or non-standard components.
- 1648 • Please refer to our LLM policy in the NeurIPS handbook for what should or should not
1649 be described.