A SUB-MODULARITY OF S

Observation 2. *The score function $S : \mathcal{P}(\mathcal{G}) \to \mathbb{R}$ defined by*

$$S(G) = \mathbb{E}_{f_{out}}[\max_{g \in G} \max(0, f_{out}(g))] \tag{7}$$

is submodular.

Proof. Straightforward:

$$\begin{aligned}
S(G \cup \{g\}) &= \mathbb{E}_{f_{\text{out}}}[\max_{g' \in G \cup \{g\}} \max(0, f_{\text{out}}(g'))] \\
&= \mathbb{E}_{\eta}[\max_{g' \in G \cup \{g\}} \max(0, f_{\text{out}}(g', \eta))] \\
&\leq \mathbb{E}_{\eta} \max_{G}[\max(0, f_{\text{out}}(g', \eta)) + \max(0, f_{\text{out}}(g, \eta))] \\
&= \mathbb{E}_{\eta} \max_{G}[\max(0, f_{\text{out}}(g', \eta))] + \mathbb{E}_{\eta}[\max(0, f_{\text{out}}(g, \eta))] \\
&= S(G) + S(\{g\})
\end{aligned}$$

□

Corollary 1. *The greedy algorithm which iteratively selects points maximizing $S(G)$ is a $1 - 1/e$ approximation of the optimal.*

B BIOLOGY BACKGROUND

Here we provide the mathematical formalization of the engaged processes in the CRISPR-based gene knockout experiments from gene embeddings to assay readouts. We take a comprehensive approach for clarity but not all notations below are used in this work.

- **Genes:** Let $\{g_1, g_2, \ldots, g_m\}$ with $g_i \in \mathcal{G}$ be all available genes for intervention.
- **Disease phenotype:** Several phenotype measurements are possible for every disease. Let $d \in D = \{d_1, d_2, \ldots, d_l\}$ be such a measurement from the list of l possible readouts.
- **Intermediate phenotype functions:** Instead of the actual disease phenotype, intermediate readouts are used to measure the effect of a gene intervention on the disease phenotype. These readouts should be correlated with the downstream outcomes, but may present a simplified view of the disease action; for example, they might include the expression of certain proteins in a cancerous cell culture which are known to correlate with tumour growth rate (the disease phenotype). We let $ip \in IP = \{ip_1, ip_2, \ldots, ip_p | ip : D \to \mathbb{R}\}$ be the set of maps from disease phenotype to real numbers that are the intermediate readouts for the effect of each gene intervention.
- **Knock-out function:** $\psi : G^m \to \mathcal{P}(G)$ shows which genes to intervene on. It takes the set of all available genes as input and returns the subset of genes to get knocked out.
- **Disease mechanism function:** $f : G \times \mathcal{P}(G) \to D^l$. This function takes all available genes and also the intervened subset and returns how the effect of the intervention on disease phenotype.
- **Knock-out representation** $\phi_{ko} : \mathcal{P}(G) \to \mathbb{R}^{d_{ko}}$ takes the subset of genes to knock out and returns a real-valued vector as the representation of this intervention.
- **Learnable mechanism:** To make the disease mechanism function amenable to learning algorithms, we use the intervention representation in the input and intermediate phenotype read-out in the output and work with $\{F_j : F_j = ip_j \circ f \circ \phi_{ko}^{-1} \text{for} 1 \leq j \leq p\}$ where $F_j : \mathbb{R}_{ko}^d \to \mathbb{R}$ is the effect of a knock-out represented by the knock-out representation ϕ_{ko} in the input on the j^{th} intermediate phenotype read-out in the output.

It is natural to work with real-valued functions with real-value domain which are more friendly to function estimation algorithms. For example, one can use MSE error as a metric to learn the

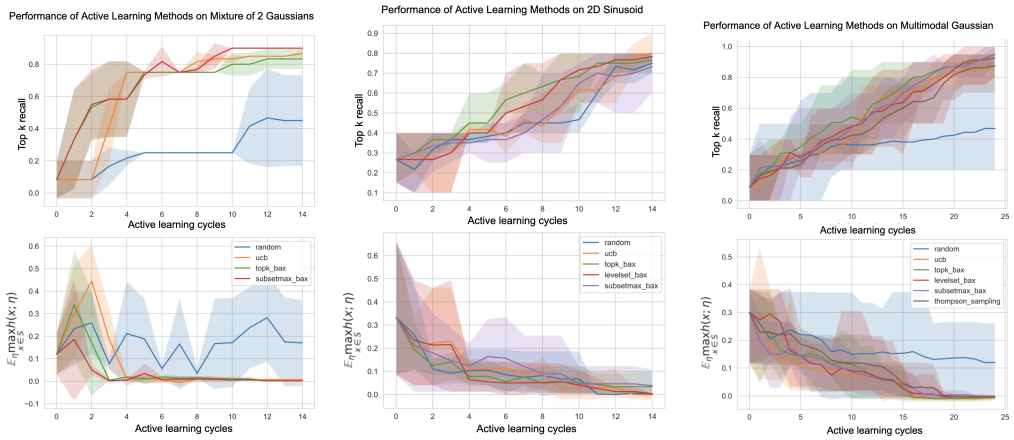

Figure 4: Top-k recall and expected maximal intervention value on: a) a mixture of two RBF kernels; b) a one-dimensional linear combination of sinusoids with multiple local optima; c) a mixture of four RBF kernels of varying scales.

intervention-to-assay mechanism from the available labeled datasets $D = \{(x_i, y_i)\}_{i=1}^n$ using the objective function

$$\hat{F}_j = \frac{1}{n} \sum_{(x_i, y_i)} \arg\min_F \|F(x_i) - y_i)\|. \tag{8}$$

for every j that gives $\{\hat{F}_1, \hat{F}_2, \ldots, \hat{F}_p\}$. Notice that $\hat{y} = \hat{F}_j(x)$ is the best predictor of the intermediate disease phenotype (screen, assay) for the gene intervention $\psi(G^m)$ represented by $x = \phi_{ko}(\psi(G^m))$.

C Clustering of optimal interventions

In the GeneDisco experiments (§ 5.2), we define a diversity metric based on the recall of Top-K clusters. These clusters are obtained for each assay as follows. All experiments we carried out in § 5.2 leverage the Achilles dataset (Dempster et al., 2019) from GeneDisco to represent the different interventions. This dataset characterizes each gene with an 808-dimensional vector. We first select the optimal interventions as the ones in the top percentile of disease phenotype for a given assay. We then project the Achilles representations of each intervention into a lower-dimensional subspace of dimension 20 with PCA. We then fit a Gaussian Mixture Model (GMM) with 20 mixtures to obtain the different clusters, selecting the best result out of 20 random initializations.

D Additional empirical evaluations

For reproducibility, code for all experiments in this work can be found at the following url: `https://github.com/anonymous35780/solaris-2023-iclr`.

D.1 Sample complexity on synthetic datasets

Our objective in this section is to validate a number of properties of the proposed method in interpretable synthetic datasets.

- **Sample complexity:** our method requires fewer samples to reach a global optimum relative to random sampling or naive uncertainty maximization methods.
- **Diversity of candidate set:** unlike standard Bayesian optimization methods, our approach identifies a set of points which approximately maximize the function while also maintaining diversity with respect to a pre-chosen metric, improving the robustness of the candidate set to uncertainty in the mapping between observable and terminal outcomes.

In these experiments and in Figure 2 we consider a number of baselines, including the following.

- **Random:** $\mathbf{x}^* \sim \text{Unif}(\mathcal{X} \setminus \mathcal{D}_t)$.
- **UCB:** naive upper-confidence sampling approach, letting $c \in \mathbb{R}$ be some constant: $\mathbf{x}^* = \arg\max_{\mathbf{x} \in \mathcal{X}} \mu(\mathbf{x}) + c\sqrt{\sigma^2(\mathbf{x})}$.
- **BAX** acquisition (Algorithm 2) for algorithm $\mathcal{A} \in \{\text{Top-k, Levelset, Disco}\}$.
- **Thompson sampling:** acquisition based on maximum of sampled function from a Bayesian posterior. $\mathbf{x}^* = \arg\max_{\mathbf{x} \in \mathcal{X}} \widehat{f}_{\text{ip}}(\mathbf{x}) \quad \widehat{f}_{\text{ip}} \sim P(f_{\text{ip}} | \mathcal{D}_{\text{train}})$.
- **Active sampling:** maximize uncertainty over the input set $\mathbf{x}^* = \arg\max_{\mathbf{x} \in \mathcal{X}} \sigma^2(\mathbf{x})$.

We consider the following synthetic datasets, where for all synthetic experiments we use a batch size equal to one.

Mixture-of-Gaussians: pdf of a mixture of gaussians with means [-0.5, 0.5], variances 0.1 and relative weights [0.3, 0.7]. $x \in [-1, 1]$.

Multimodal mixture: given domain $[-7, 7]$, outputs the (scaled) density of a mixture of Gaussians with means $\{-4, -2, 0, 3\}$, variances $\{0.3, 0.35, 0.3, 0.35\}$, and weights $\{0.6, 0.45, 0.5, 0.4\}$. **2-d sinusoid:** $f(x) = \sin\left[\frac{1}{2}\begin{pmatrix} 0.25 & -\frac{1}{\pi} \\ 0.1 & .02 \end{pmatrix}\mathbf{x}\right], \mathbf{x} \in \mathbb{R}^2, -\pi < \mathbf{x} < \pi$

D.2 GeneDisco detailed results

We provide below detailed results across the five CRISPR assays from the GeneDisco benchmark: the Interferon γ and Interleukin 2 assays based on Schmidt et al. (2021), the Leukemia assay with NK cells from Zhuang et al. (2019), the SARS-CoV-2 assay from Zhu et al. (2021) and the Tau protein assay from Sanchez et al. (2021). All interventions for the five assays were represented based on the Achilles dataset (Dempster et al., 2019). For the active learning baselines already present in GeneDisco we used the same hyperparameters as in Mehrjou et al. (2021). For the additional baselines introduced in this work, we use standard/default hyperparameters everywhere (see our codebase for all details: `https://github.com/anonymous35780/solaris-2023-iclr`) except as specified in Appendix D.3. To prevent model overfitting during the various active learning cycles, we closely followed experimental protocol in Mehrjou et al. (2021) and selected similar model architectures and hyperparameters.

We observe in Table 2 that DiscoBAX outperforms all other 13 baselines we compare against on 3 out of the 5 datasets included in the GeneDisco benchmark, performs on par (significant overlap of confidence intervals) with the best methods on the 4th one (Tau protein) and is only outperformed by "random selection" on the last one (SARS-coV2). As discussed in section 5.2 and as noted in Mehrjou et al. (2021), the fact that random outperforms all other 13 methods on that dataset seems to indicate an issue with the data (eg., the feature space does not correlate with the disease phenotype, high label noise) rather than an algorithmic issue. Critically, none of the other baselines performs consistently high on all 5 assays: for instance, 'random' performs relatively poorly on all other 4 assays and the other methods that are on par with DiscoBAX on the Tau protein assay (eg., BADGE, Coreset) have inconsistent performance on other assays. Aggregated performance across assays is reported in Table 1 and demonstrates the overall higher performance of DiscoBAX over other baselines.

D.3 GeneDisco experiments - hyperparameter selection

For the three BAX alogirthms (Top-K BAX, Levelset BAX and DiscoBAX), we optimize the main hyperparameters of each method (ie., respectively the K parameter the level threshold and the number of Sets in SetSelect). To mitigate the risk of overfitting, we select our hyperparameters based on a single assay (the 'Tau protein' assay), and use the obtained optimal values in experiments for all assays. We perform a grid search for each hyperparameter, repeating each experiment over 5 seeds. We find that on that dataset, optimal values for the hyperparameters are respectively k=5, Levelset=1.0 and S=10.

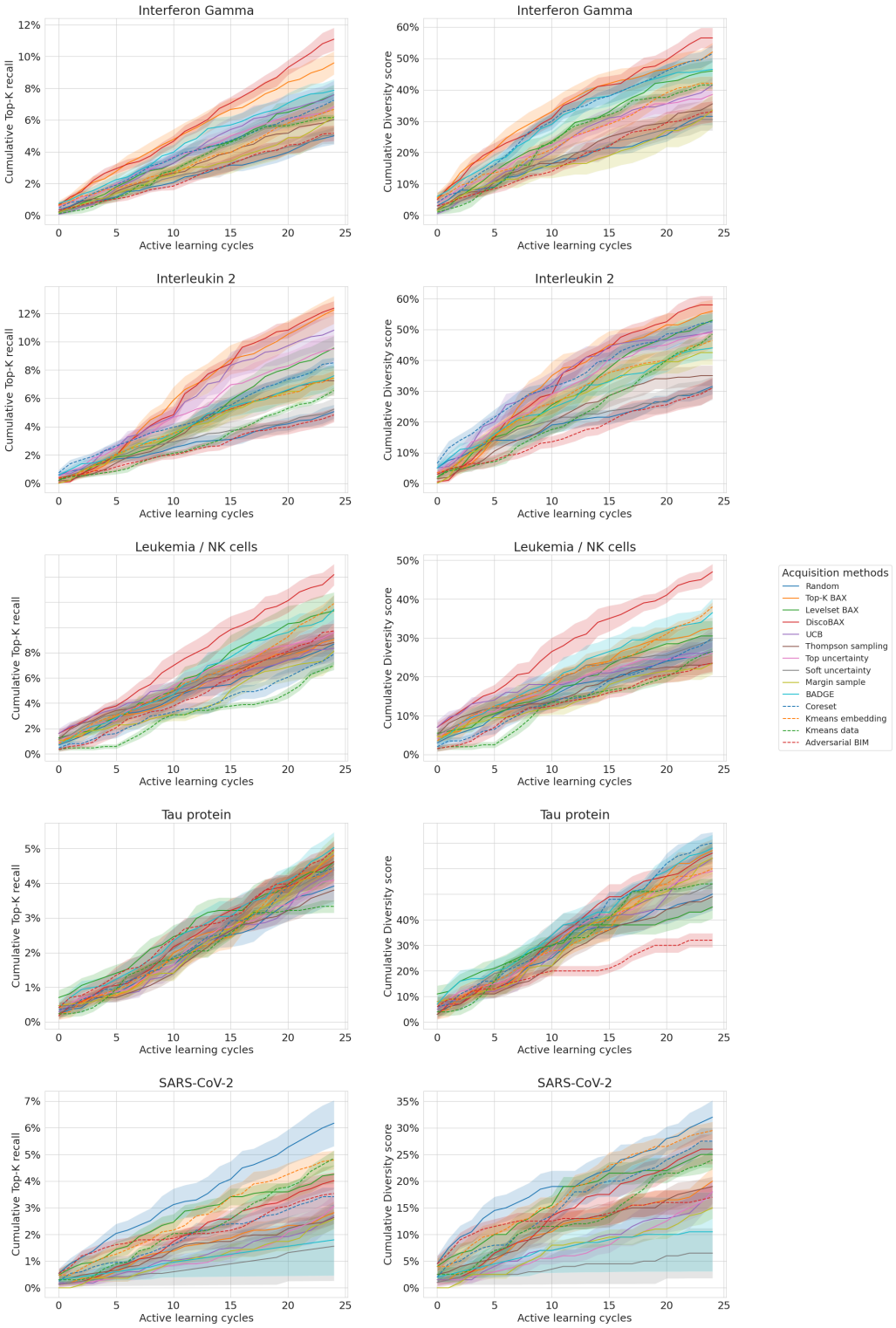

Figure 5: **Top-K recall and Diversity score Vs acquisition cycles for all GeneDisco CRISPR assays**

E DEEPER INSIGHT INTO THE ALGORITHM

E.1 INSIGHTS OF EQ. (4):

In this section, we design a simplistic scenario to provide more insight into the proposed objective function eq. (4) and how it serves two purposes, i.e., choosing a set of interventions with high phenotype values and high diversity. For convenience, in fig. 6, we show a simple scenario where 2 out of 3 genes are to be chosen, i.e., $|\mathcal{S}| = 2$ and $|\mathcal{X}| = 3$. There are three ways of choosing a pair of genes out of three options. We aim to show which pair is favoured by eq. (4). Without loss of generality, assume $\mathbf{x}_1$ is chosen. Due to the probabilistic model of f_{out}, all $y_i = f_{\text{out}}(x_i), i = 1, 2, 3$ are random variables whose probability densities (P_i) are plotted next to each gene. It is observed that P_1 and P_3 are concentrated at larger values (higher regions of the vertical axis) compared to P_2 that puts much of its mass at lower values. Hence, in most realization, y_1 takes a large value (star) and $y_1 \approx \max(y_1, \cdot)$ as the second argument is sampled from distributions concentrated at lower values (P_2) or almost equally large values (P_3). The second argument becomes important in the rare events when y_1 takes a small value (cross). In this case, the output of $\max(y_1, \cdot)$ is no longer determined by its first argument and is, with high probability, influenced by the second argument which takes on a large value if realized from P_3 compared with P_2 as the former is concentrated at larger values. Hence, choosing the pair $(\mathbf{x}_1, \mathbf{x}_3)$ produces a larger average than $(\mathbf{x}_1, \mathbf{x}_2)$ and is therefore favourable by eq. (4). Moreover, it is implicitly assumed in the above reasoning that P_1, P_2 and P_3 are not highly correlated. Otherwise, a small value of y_1 led to a small value of y_3 as well. Hence, eq. (4) chooses the genes which produce large values and the mechanisms that are modeled as the random effect are as independent as possible. This choice of genes increases the chance that if one gene fails to proceed to further steps of the drug discovery pipeline for some reason such as safety, tractability, etc, the other chosen genes will preserve high chances of success as they are likely to be involved in mechanisms different from those that cause the failure of the previous gene.

E.2 INSIGHTS OF FIG. 1:

This illustrates the motivation and goal of this research which is finding mathematical formulation and practical implementation of an algorithm that meets the actual needs of initial stages of drug discovery pipeline that neither value-seeking nor diversity-seeking methods can fulfill. The phenotypic effect of genetic perturbation can follow a complex function with many modes. We are mainly interested in genes which cause large changes in the measured phenotype as those are the genes that engage more with the disease and can be a potential target for a drug compound. However, as the figure shows, the value-seeking methods stop after finding one mode of the function (the light gray triangles which are concentrated in one of the modes but do not cover the other modes which have equally large values). This is risky since the genes that are associated with that mode are probably correlated in the sense that if one of them fails in the further steps of the drug discovery pipeline, the other may also fail with high likelihood. On the other end of the spectrum, although a diversity-seeking algorithm proposes uncorrelated genes that are unlikely to fail together (the dark gray circles which cover a large domain but miss the modes), it is highly inefficient and a large number of chosen genes may not be highly involved in the disease mechanism. Hence, the nature of the problem requires a middle-ground method that seeks the modes of the underlying function but covers as many does as possible (the red stars that efficiently cover all modes but not in-between spaces) so that if the genes associated with one mode fails, those associated with the other modes have chance to proceed in the pipeline.

F ADDITIONAL EMPIRICAL EVALUATIONS ON SYNTHETIC DATASET

Finally, we include an evaluation of the Expected Improvement (EI) acquisition function on the same task as was previously illustrated in Figure 2. Because we use an acquisition batch size of one in these experiments, the parallel acquisition strategy qEI coincides with the incremental expected improvement acquisition function. Concretely, the expected improvement acquisition function per-

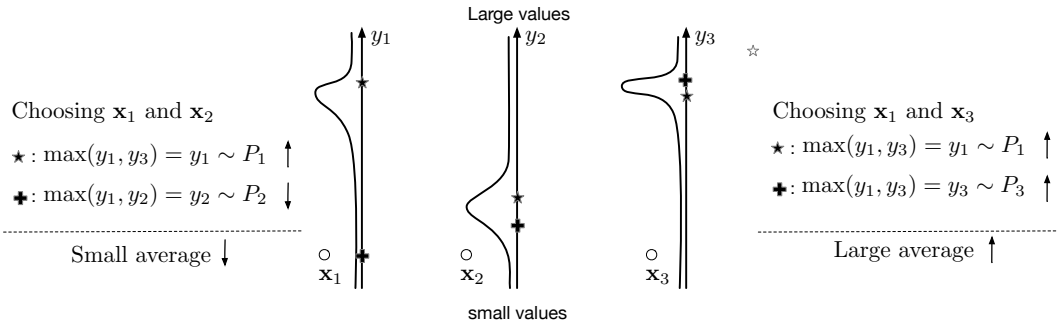

Figure 6: A simple visualization to gain insight into eq. (4).

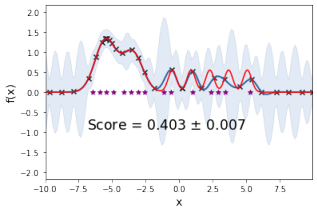

Figure 7: Evaluation of the EI acquisition function on the regression problem discussed previously.

forms the following maximization, given some pool $\mathcal{D}$ of already sampled points:

$$\max_{x \notin \mathcal{D}} \mathbb{E}_{P(f(x)|\mathcal{D})} \max(f(x) - f(x^*), 0) \tag{9}$$

where x^* is the element of $\mathcal{D}$ which maximizes f and $P(\cdot|\mathcal{D})$ denotes the posterior over function values $f(x)$ for some fixed x.

Table 2: **Performance comparison on GeneDisco CRISPR assays** We report the performance of DiscoBAX and all baselines methods on all datasets from the GeneDisco benchmark.

Dataset	Method	Top-K recall	Diversity score	Overall score
Interferon γ	Adversarial BIM	33% (3.6%)	5.2% (0.5%)	13.1% (1.4%)
	BADGE	46.5% (3.9%)	7.9% (0.7%)	19.1% (1.7%)
	Coreset	51.5% (3%)	7.2% (0.6%)	19.3% (1.3%)
	DiscoBAX (ours)	**56.5% (3.4%)**	**11.1%** (0.8%)	**25% (1.6%)**
	Kmeans Data	41.5% (1.3%)	6.1% (0.2%)	16% (0.5%)
	Kmeans Embedding	42% (2.4%)	6.7% (0.6%)	16.7% (1.2%)
	Levelset BAX	46% (3.6%)	7.6% (0.7%)	18.7% (1.5%)
	Margin sample	33.5% (5.8%)	6.2% (1.1%)	14.4% (2.6%)
	qEI	45% (3.6%)	7.9% (0.4%)	18.9% (1.1%)
	qPOI	44% (3.1%)	8.1% (0.4%)	18.9% (1.1%)
	qUCB	43.7% (3.7%)	7.8% (0.4%)	18.5% (1.2%)
	Random	31.5% (2.9%)	5% (0.6%)	12.5% (1.3%)
	Soft Uncertainty	30.5% (3.7%)	5.1% (0.6%)	12.4% (1.5%)
	Thompson Sampling	35.5% (2.6%)	6% (0.7%)	14.6% (1.4%)
	Top-K BAX	52% (3.1%)	9.6% (0.8%)	22.3% (1.5%)
	Top Uncertainty	38.5% (3%)	6.8% (0.7%)	16.2% (1.4%)
	UCB	41.5% (2.6%)	7.6% (0.9%)	17.7% (1.6%)
Interleukin 2	Adversarial BIM	31% (3.6%)	4.8% (0.5%)	12.2% (1.4%)
	BADGE	44% (3.6%)	7.6% (1%)	18.3% (1.9%)
	Coreset	52.5% (2.9%)	8.5% (0.4%)	21.1% (1.1%)
	DiscoBAX (ours)	**58% (3.1%)**	**12.4% (0.5%)**	**26.8% (1.3%)**
	Kmeans Data	48.5% (1.7%)	6.6% (0.3%)	17.8% (0.7%)
	Kmeans Embedding	46.5% (2.8%)	7.5% (0.5%)	18.6% (1.2%)
	Levelset BAX	53% (3%)	9.5% (0.9%)	22.5% (1.6%)
	Margin sample	42.5% (4.2%)	7.4% (0.9%)	17.8% (2%)
	qEI	52.5% (2.9%)	11.4% (0.9%)	24.5% (1.6%)
	qPOI	54% (2.8%)	11.9% (0.9%)	25.3% (1.6%)
	qUCB	52.5% (4.7%)	11.3% (1%)	24.4% (2.2%)
	Random	31.5% (2.7%)	5.1% (0.5%)	12.6% (1.2%)
	Soft Uncertainty	31% (4%)	5.2% (0.8%)	12.7% (1.8%)
	Thompson Sampling	35% (3.5%)	7.2% (1.1%)	15.9% (2%)
	Top-K BAX	56% (3.9%)	12.2% (1%)	26.2% (2%)
	Top Uncertainty	49% (2.8%)	9.5% (1.1%)	21.6% (1.7%)
	UCB	49.5% (2.8%)	10.8% (1.1%)	23.1% (1.8%)
SARS-CoV-2	Adversarial BIM	17% (2.4%)	3.5% (0.8%)	7.7% (1.4%)
	BADGE	10.5% (7.8%)	1.8% (1.4%)	4.3% (3.3%)
	Coreset	27.5% (2.6%)	3.4% (0.4%)	9.7% (1%)
	DiscoBAX (ours)	26% (3%)	4% (0.3%)	10.2% (1%)
	Kmeans Data	24% (1.9%)	4.9% (0.3%)	10.8% (0.8%)
	Kmeans Embedding	29.5% (1.7%)	4.8% (0.4%)	11.9% (0.8%)
	Levelset BAX	25% (2.6%)	4.3% (0.4%)	10.3% (1%)
	Margin sample	15% (2.1%)	2.6% (0.4%)	6.2% (0.9%)
	qEI	23.5% (3.3%)	4.1% (0.8%)	9.8% (1.6%)
	qPOI	21.7% (3%)	3.5% (0.4%)	8.7% (1%)
	qUCB	21.5% (2.9%)	4.4% (0.6%)	9.7% (1.3%)
	Random	**32% (3.3%**	**6.2% (0.9%)**	**14% (1.7%)**
	Soft Uncertainty	6.5% (4.9%)	1.6% (1.4%)	3.2% (2.6%)
	Thompson Sampling	19% (1.9%)	2.6% (0.3%)	7.1% (0.7%)
	Top-K BAX	20% (2.8%)	2.8% (0.4%)	7.5% (1.1%)
	Top Uncertainty	18% (2.3%)	3.1% (0.5%)	7.4% (1%)
	UCB	18% (2.4%)	2.7% (0.5%)	7% (1.1%)

Dataset	Method	Top-K recall	Diversity score	Overall score
Leukemia/NK	Adversarial BIM	23.5% (2.2%)	4.9% (0.3%)	10.7% (0.8%)
	BADGE	36.5% (3.9%)	5.7% (0.6%)	14.4% (1.5%)
	Coreset	30% (3.2%)	3.9% (0.4%)	10.9% (1.2%)
	DiscoBAX (ours)	**47% (2.1%)**	**7.1% (0.4%)**	**18.2% (1%)**
	Kmeans Data	26.5% (1.1%)	3.5% (0.2%)	9.6% (0.4%)
	Kmeans Embedding	38% (1.3%)	5.9% (0.4%)	15% (0.7%)
	Levelset BAX	30.5% (4.1%)	5.7% (0.8%)	13.1% (1.8%)
	Margin sample	23.5% (3.1%)	4.1% (0.6%)	9.8% (1.4%)
	qEI	26.5% (3.2%)	4.3% (0.6%)	10.7% (1.4%)
	qPOI	31% (1.5%)	4.8% (0.6%)	12.2% (0.9%)
	qUCB	33% (2.9%)	5.4% (0.7%)	13.4% (1.4%)
	Random	26.5% (3.5%)	4.3% (0.6%)	10.7% (1.5%)
	Soft Uncertainty	29.5% (2.3%)	4.6% (0.4%)	11.6% (0.9%)
	Thompson Sampling	23.5% (2.6%)	4.4% (0.4%)	10.2% (1.1%)
	Top-K BAX	32.5% (2.9%)	4.5% (0.4%)	12.1% (1.1%)
	Top Uncertainty	26% (3.1%)	4.8% (0.6%)	11.2% (1.3%)
	UCB	26.5% (3%)	4.2% (0.6%)	10.5% (1.3%)
Tau protein	Adversarial BIM	16% (1.5%)	**5% (0.3%)**	8.9% (0.6%)
	BADGE	34% (2.8%)	**5% (0.5%)**	**13.1% (1.1%)**
	Coreset	**35% (2.2%)**	4.4% (0.3%)	12.5% (0.9%)
	DiscoBAX (ours)	33% (2.1%)	4.6% (0.4%)	12.3% (0.9%)
	Kmeans Data	27% (1.1%)	3.3% (0.2%)	9.5% (0.5%)
	Kmeans Embedding	30% (2.6%)	4.4% (0.4%)	11.5% (1%)
	Levelset BAX	22.5% (2.4%)	4.6% (0.5%)	10.2% (1.1%)
	Margin sample	32% (3.3%)	4.9% (0.5%)	12.5% (1.2%)
	qEI	32.1% (2.8%)	4.3% (0.6%)	11.7% (1.3%)
	qPOI	31% (2.5%)	4.5% (0.5%)	11.8% (1.1%)
	qUCB	31.2% (2.6%)	4.4% (0.4%)	11.7% (1%)
	Random	25% (3.3%)	3.9% (0.5%)	9.9% (1.3%)
	Soft Uncertainty	27% (2.4%)	4.6% (0.4%)	11.1% (0.9%)
	Thompson Sampling	24.5% (2.4%)	3.8% (0.3%)	9.7% (0.9%)
	Top-K BAX	33.5% (2.4%)	4.8% (0.4%)	12.7% (1%)
	Top Uncertainty	29.5% (1.2%)	4.1% (0.2%)	11% (0.5%)
	UCB	32% (2.7%)	4.4% (0.3%)	11.8% (1%)

Table 3: **GeneDisco experiment - Hyperparameter selection**

Method	Hyperparameter value	Top-K recall	Diversity score	Overall score
Top-K BAX	2	32% (3.6%)	4% (0.5%)	11.3% (1.3%)
	3	32% (3.3%)	4.3% (0.5%)	11.8% (1.1%)
	5	33% (2.4%)	4.4% (0.4%)	**12.1% (0.9%)**
	10	30% (3.2%)	4.2% (0.4%)	11.2% (1.3%)
Levelset BAX	0.8	19% (2.1%)	3.4% (0.3%)	8% (0.8%)
	1	30% (4.3%)	5.4% (0.7%)	**12.7% (1.8%)**
	1.2	21% (1.3%)	4.1% (0.6%)	9.3% (0.9%)
	1.5	29% (0.7%)	5.4% (0.5%)	12.5% (0.6%)
DiscoBAX	2	36% (5.3%)	4.8% (0.8%)	13.1% (2%)
	3	32% (2.6%)	4.1% (0.5%)	11.4% (1.1%)
	5	37.5% (2.7%)	5.4% (0.4%)	14.2% (1%)
	10	38% (1.8%)	5.5% (0.3%)	**14.5% (0.8%)**