Quality-Weighted Vendi Scores And Their Application To Diverse Experimental Design

Quan Nguyen [1] **Adji Bousso Dieng** [2 3]

Abstract

Experimental design techniques such as active search and Bayesian optimization are widely used in the natural sciences for data collection and discovery. However, existing techniques tend to favor exploitation over exploration of the search space, which causes them to get stuck in local optima. This *collapse* problem prevents experimental design algorithms from yielding diverse high-quality data. In this paper, we extend the Vendi scores—a family of interpretable similarity-based diversity metrics—to account for quality. We then leverage these *quality-weighted Vendi scores* to tackle experimental design problems across various applications, including drug discovery, materials discovery, and reinforcement learning. We found that quality-weighted Vendi scores allow us to construct policies for experimental design that flexibly balance quality and diversity, and ultimately assemble rich and diverse sets of high-performing data points. Our algorithms led to a 70%–170% increase in the number of effective discoveries compared to baselines.

1. Introduction

Many real-world tasks can be framed as expensive discovery problems, where one explores large databases in search of rare, valuable items. For instance, a scientist aiming to find a drug for a disease may need to iterate over millions of molecules to discover those that bind to specific biological targets. These search problems also often involve an expensive labeling process: the scientist will need to perform costly, time-consuming experiments to test for a molecule's binding activity to study its characteristics. This high cost in experimentation rules out exhaustive search and motivates the need for more sophisticated search strategies.

Powerful active search (AS) and Bayesian optimization (BayesOpt) techniques have been developed over the years (Garnett et al., 2012; Jiang et al., 2017; Nguyen et al., 2021; Nguyen & Garnett, 2023; Eriksson et al., 2019) to tackle the problems mentioned above. While these advances have led to a more flexible experimental design framework, there has been a recent surge in interest in modifying existing algorithms to not only perform effective search/optimization but also induce more *diversity* during experimentation. For example, Malkomes et al. (2021) proposed an AS method in which data points sufficiently close to discoveries already made are removed from the pool, effectively encouraging a more diverse search. Maus et al. (2023), on the other hand, formulated a BayesOpt problem where the goal is to maintain and optimize multiple solutions that are constrained to be different from one another. These works leverage local penalization to induce diversity.

Even though local penalization is a natural way to encourage diversity, it requires defining constraints to control the algorithms' behavior. However, setting these constraints can be challenging for complex, high-dimensional spaces. In this work, we present an alternative approach to local penalization to enforce diversity in experimental design. More specifically, we extend the Vendi scores (VS) (Friedman & Dieng, 2023; Pasarkar & Dieng, 2023) to account for the quality of the items in a given input set. We call these new scores *quality-weighted Vendi scores*.

These quality-weighted Vendi scores offer us a mathematically convenient way to evaluate quality and diversity and yield a unified framework for diverse experimental design. We applied the quality-weighted Vendi scores to both AS and BayesOpt across a wide variety of tasks involving drug discovery, materials discovery, and reinforcement learning. For all these tasks, we compared against strong existing baselines for experimental design, as well as analyzed the performance of our algorithms for various quality-diversity trade-offs. In all our experiments, we found that experimental design algorithms leveraging quality-weighted Vendi scores tend to outperform their counterparts in terms of both the diversity and the quality of the data points they yield.

[1]Department of Computer Science & Engineering, Washington University in St. Louis [2]Department of Computer Science, Princeton University [3]Vertaix. Correspondence to: Quan Nguyen <quan@wustl.edu>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

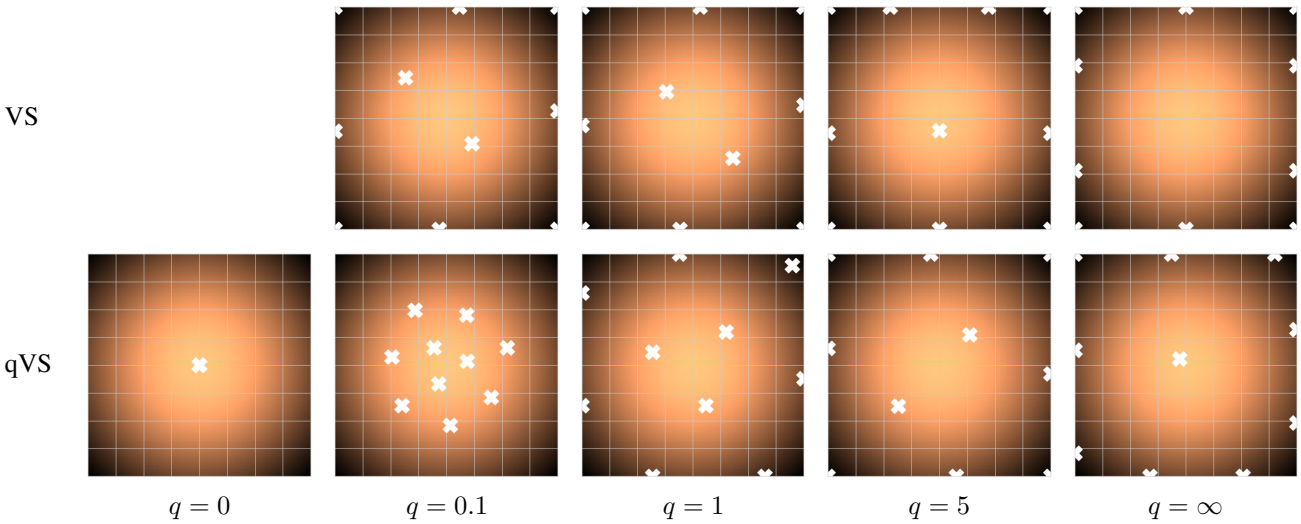


Figure 1. Batches of 10 data points maximizing various VS and qVS functions, obtained with multi-start gradient-based optimization. The scoring function is a Gaussian function centered at the middle point, as illustrated by the heat maps. The quality-weighted Vendi Score balances between the quality of the selected data points and their diversity; this balance is smoothly controlled by the order q.

2. The Quality-Weighted Vendi Score

We first provide background on the Vendi score (VS) as a metric of diversity of a given set and introduce our extension to the VS that incorporates quality scores of individual members. We then examine computationally efficient ways to optimize this quality-weighted VS, which will allow us to tackle a wide range of experimental design tasks in Sect. 3.

2.1. The Vendi Score

Consider a finite set of data points $X = \{x_i\}_{i=1}^n$ in some domain $\mathcal{X}$. Friedman & Dieng (2023) introduced the Vendi Score (VS) to characterize the diversity of a collection of items such as X. VS is defined as the exponential of the Shannon entropy of the normalized eigenvalues of the kernel similarity matrix corresponding to X. Specifically, given a positive semidefinite similarity function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, where $k(x, x) = 1$ for all $x \in \mathcal{X}$, denote by $K \in \mathbb{R}^{n\times n}$ the kernel matrix corresponding to the set $X = \{x_i\}_{i=1}^n$ where each entry $K_{i,j} = k(x_i, x_j)$. Further denote the eigenvalues of K as $\lambda_1, \lambda_2, \ldots, \lambda_n$. The VS is defined as:

$$\mathrm{VS}(X; k) = \exp\left(-\sum_{i=1}^{n} \overline{\lambda}_i \log \overline{\lambda}_i\right), \tag{1}$$

where $\overline{\lambda}_1, \overline{\lambda}_2, \ldots, \overline{\lambda}_n$ are the *normalized* eigenvalues of K such that $\overline{\lambda}_i = \lambda_i / \sum_{i=1}^n \lambda_i$, and $0 \log 0$ is defined to be 0. As K is a positive semidefinite matrix, the eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$ are non-negative and the normalized eigenvalues $\overline{\lambda}_1, \overline{\lambda}_2, \ldots, \overline{\lambda}_n$ sum to 1. The VS is then valid and can be viewed as the Shannon entropy of these normalized eigenvalues.

Friedman & Dieng (2023) explored the features of the VS as a diversity metric, and demonstrated that $\mathrm{VS}(X; k)$ can be interpreted as the effective number of unique samples of X. In the extreme case where all items in X are unique, K is the identity matrix and $\mathrm{VS}(X; K) = n$. At the other end of the spectrum, if all items are identical, K is the all-one matrix and $\mathrm{VS}(X; K) = 1$. Overall, the VS offers us a mathematically principled yet convenient way to quantify the diversity of the items in a set X. Unlike the determinantal point process (DPP) likelihoods (Kulesza & Taskar, 2012), another tool commonly used in diversity-related machine learning tasks, VS does not reduce to 0 when there are duplicates in the input set X. Compared to Hill numbers (Hill, 1973), the VS is not restricted to the assumption that different data points (species) are completely dissimilar to one another and thus allows us to effectively account for similarity between pairs of items.

2.2. Accounting for Quality in the Vendi Score

The VS captures diversity but treats all items in X the same when it comes to their quality. In many situations, however, we may reasonably want our diversity metric to further express preference for items that exhibit desirable characteristics by incorporating "quality scores", upweighting or downweighting the final output based on the quality of individual items. In other words, given a score function $s : \mathcal{X} \to \mathbb{R}$ that quantifies the quality of a given item $x \in \mathcal{X}$, we are interested in an extension to the VS that increases not only with more diverse but also with higher-quality items.

One may be tempted to mimic the DPPs' ability to naturally incorporate "quality scores" into their likelihoods, whereby

the input kernel matrix is modified so that each entry $K_{i,j}$ is multiplied with the two corresponding quality scores $s(x_i)$ and $s(x_j)$. Unfortunately, applying the same modification here for the VS does not lead to desirable results. The mismatch in behavior stems from the inherent mathematical difference between the two operations. The DPP likelihood can be interpreted as the volume spanned by the quality-weighted feature vectors of individual items x_i whose outer products yield the kernel matrix. It therefore increases with higher values of $s(x_i)$. The VS, on the other hand, computes the Shannon entropy of the normalized eigenvalues of the kernel matrix, which does not behave monotonically with respect to the values of the entries in K.

Our chosen solution to extend the VS to account for quality is simple—multiply the VS as defined in Eq. (1) by the average quality score of individual items:

$$\mathrm{qVS}(X; k, s) = \left(\sum_{i=1}^{n} s(x_i)/n\right) \mathrm{VS}(X; k). \tag{2}$$

This quality-weighted VS, qVS for short, possesses multiple desiderata partially inherited from the VS. First, the output is maximized when all items are maximally diverse (the covariances $K_{i,j} = 0$) and achieve the highest quality score. Conversely, the qVS is minimized when all items are identical and yield the lowest quality score. Fixing the quality scores $s(x_i)$, more diverse items (as measured by the VS) yield higher qVS values. Fixing the diversity score measured by the VS, higher-quality items yield higher qVS values. The intuitive interpretation of the VS as the effective number of unique samples carries over as well. The qVS can be interpreted as the effective number of high-quality samples in the input set. Overall, the qVS allows us to express our preference for diverse sets of high-quality data points.

2.3. Controlling the Quality–Diversity Trade-off

The balance between diversity and quality for a given set of items is explicitly achieved by maximizing the qVS in Eq. (2). However, in many scenarios, we may reasonably seek to control this balance to favor more diverse or higher-quality items, depending on our goal. Inspired by the Rényi entropy, Pasarkar & Dieng (2023) proposed a generalization of the VS by introducing an extra hyperparameter $q \geq 0$, defining the VS of order q as:

$$\mathrm{VS}_q(X; k) = \exp\left(\frac{1}{1-q} \log\left(\sum_{i=1}^{n} (\overline{\lambda}_i)^q\right)\right), \tag{3}$$

where $\overline{\lambda}_1, \overline{\lambda}_2, \ldots, \overline{\lambda}_n$ are, again, the normalized eigenvalues of the kernel matrix K corresponding to the input set X. Further, when $q \in \{0, 1, \infty\}$, the VS_q is defined as the limit of Eq. (3) as q approaches the target order. We briefly note that when $q = 1$, we recover the traditional VS that is the Shannon entropy of the normalized eigenvalues of the similarity matrix.

The order q smoothly controls the sensitivity to the non-uniformity of these eigenvalues, and thus the evaluation of the diversity of X. A smaller value of q leads to a VS that is more sensitive to X in that the VS increases faster than that of a larger q when we add items to X. In the extreme case, VS_0 is simply the count function, which increases by 1 every time a new data point is added to X, ignoring the diversity of the set. For a larger q, it takes a completely unique data point (with zero covariances with other items) to lead to an increase of 1 in the VS. We use this more general VS in Eq. (2) to effectively balance quality and diversity.

2.4. Optimizing the Quality-Weighted Vendi Score

With the qVS in hand, we can evaluate the value of a given input set X, assessing its diversity and the quality of its members. A question naturally arises: given the domain $\mathcal{X}$, how can we identify a subset of a particular size that maximizes the qVS? That is, we want to find:

$$X_* = \underset{X \subset \mathcal{X}, |X|=n}{\arg\max} \ \mathrm{qVS}(X; k, s). \tag{4}$$

As the VS in Eq. (1) is differentiable, if the domain $\mathcal{X}$ is a compact space, the qVS can be efficiently maximized using an off-the-shelf gradient-based optimizer with multi-start, assuming that we also have access to the gradients of the kernel k as well as the score function s.

The panels of Fig. 1 show an example in 2 dimensions within $\mathcal{X} = [-1, 1]^2$. Here, we use an isotropic Gaussian kernel: $k(\boldsymbol{x}_1, \boldsymbol{x}_2) = \exp\left(-\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2/2\right)$, and a Gaussian score function centered at the origin: $s(\boldsymbol{x}) = \exp\left(-\|\boldsymbol{x}\|^2/2\right)$, shown as the heat maps. We run a gradient-based optimizer with multi-start to find a batch X_* of size $n = 10$ maximizing the VS (top row) and the qVS (bottom row) of different orders q and show the members of the optimal batches as red x's. We notice interesting distinctions between the panels. The data points at the top maximizing the VS are well spread out across the domain, maximizing pure diversity. Further, as q increases, groups of close-by data points are discouraged, and the points are pushed towards the boundary. The data points maximizing the qVS at the bottom, on the other hand, favor points that yield high values for the score function while still achieving excellent diversity among the items. This figure showcases the qVS' ability to account for both quality and diversity, with the order q serving to flexibly control the balance between the two, with priority for diversity growing as q increases.

If we can only evaluate k or s in a black-box manner, or if the domain $\mathcal{X}$ is a discrete set of items, optimization of the qVS becomes more challenging. In fact, given a dis-

crete search space $\mathcal{X}$, exact maximization of Eq. (4) is a combinatorial problem, as we need to iterate over all possible subsets X of size n in search for the optimal X_*. We instead opt for approximate maximization of the qVS using the sequential greedy heuristic (Nemhauser et al., 1978; Krause & Guestrin, 2007), which has found application in maximizing functions with diminishing returns, including DPPs' likelihoods. More specifically, we sequentially build the approximately optimal batch $\overline{X}_*$ from the empty set, finding the item x that yields the largest increase in the qVS at each iteration:

$$x_* = \underset{x \in \mathcal{X} \setminus \overline{X}_*}{\arg\max}\ \mathrm{qVS}\left(\overline{X}_* \cup \{x\}; k, s\right) - \mathrm{qVS}\left(\overline{X}_*; k, s\right). \tag{5}$$

We repeat this greedy search until the running batch $\overline{X}_*$ reaches the desirable size, at which point we return $\overline{X}_*$ as the set that approximately maximizes the qVS. While we do not prove any theoretical guarantee in terms of optimization performance of the greedy strategy on the qVS, we empirically observe that the procedure works well in our experiments and is efficient enough to run at scale.

3. Designing Diverse Experiments with the Vendi Score

We are tasked with sequentially querying an expensive-to-evaluate oracle to obtain observations of a system of interest. At each iteration of this procedure, we train a probabilistic model on the data collected so far and use the predictions of this model to decide which data point to label next. This process is repeated for a pre-specified number of iterations. The goal is to design effective ways to collect more data to maximize a metric of interest at the end of the procedure.

3.1. Diverse Active Search

Given a large but finite pool of unlabeled data $\mathcal{X}$, we seek to identify data points belonging to a rare, valuable class of interest. We label these valuable data points, referred to as *positives*, with $y = 1$ and use $y = 0$ for the other points, referred to as *negatives*. The label of a data point is not known *a priori*, but can be determined by querying an expensive oracle. Traditional AS targets achieving the highest "hit rate", that is, maximizing the number of positives in the collected data, $\sum_{(x,y)\in\mathcal{D}} y$, where $\mathcal{D}$ is the collection of data we have chosen to label at the end of the search.

Given this formulation, active search (AS) strategies tend to become too exploitative, making many observations within regions in the search space $\mathcal{X}$ known to yield a high hit rate. As Nguyen & Garnett (2023) argued, in settings such as scientific discovery, there are diminishing returns in making additional discoveries in a frequently observed region: "a discovery in a novel region of the design space may offer more marginal insight than the 100th discovery in an already densely labeled region." We thus aim at an alternative AS setting that rewards diverse discoveries. As the VS has been established as a principled diversity metric, we propose to directly use it to measure our search performance when diversity is of interest and modify the AS objective to be maximized to be:

$$\mathrm{VS}_\mathrm{q}\big(\mathcal{D}_+; k\big), \tag{6}$$

where the operator $_+$ gives the subset of positives within a set: $\mathcal{D}_+ \triangleq \{x \mid (x, y) \in \mathcal{D}, y = 1\}$. Our goal in this diversity-aware search is to collect a set of diverse positives. Interestingly, setting $q = 0$ recovers the base version of AS, where our utility function is the count function. We thus view our formulation as a generalization of traditional AS.

How should we design our queries to the oracle, sequentially selecting among the unlabeled data, so as to maximize the objective defined above? Assuming access to a probabilistic predictive model that outputs, $\Pr(y = 1 \mid x, \mathcal{D})$, the probability that an unlabeled item x has a positive label given the data observed so far, we can derive the one-step Bayesian optimal decision, the data point x_* that maximizes the expected increase of the objective in Eq. (6):

$$x_* = \underset{x \in \mathcal{X} \setminus \mathcal{D}}{\arg\max}\quad \mathbb{E}\left[\mathrm{VS}_\mathrm{q}\big((\mathcal{D} \cup \{x\})_+; k\big)\right] - \mathrm{VS}_\mathrm{q}\big(\mathcal{D}_+; k\big), \tag{7}$$

where the expectation is taken with respect to the label of each unlabeled candidate $x \in \mathcal{X} \setminus \mathcal{D}$.

In a purely sequential regime where queries are made one after another, Bayesian decision theory guides us to select x_* to greedily maximize our VS objective. In batch settings where multiple queries are made simultaneously to maximize experimental throughput—which are common in the real world—one may be tempted to simply extend the search criterion in Sect. 3.1 from a single candidate $x \in \mathcal{X} \setminus \mathcal{D}$ to a batch of queries $X \subset \mathcal{X} \setminus \mathcal{D}$, seeking to maximize the expected increase of the VS of the positives in $\mathcal{D} \cup X$. This is a daunting task. First, for any candidate batch X of size b, we need to iterate over 2^b possible label combinations of this batch to compute the expected VS after making these queries. Second, similar to the task of finding a batch to optimize the VS or the qVS, finding a batch to optimize the expected VS means searching over a combinatorial space, requiring exponential computational effort.

The difficulties above motivate us to find a computationally tractable alternative criterion to select the next queries at each iteration of a batch AS problem. An ideal batch of queries should balance between high probability for the candidates to have positive labels (after all, positives are the target of our search) and diversity among both those queries and the already observed positives. As its goal is precisely offering an evaluation function that balances between some metric of quality and diversity, we turn to the

qVS for this task. Specifically, we set the scoring function $s(x)$ to be exactly the probability of being a positive $\pi(x) = \Pr(y = 1 \mid x, \mathcal{D})$, and use the qVS of the positives within $\mathcal{D} \cup X$ as our diverse search criterion. In other words, we seek to find:

$$X_* = \arg\max_{X \subset \mathcal{X} \setminus \mathcal{D}} \text{qVS}\big(\mathcal{D}_+ \cup X; k, \pi\big) = \arg\max_{X \subset \mathcal{X} \setminus \mathcal{D}} \left(\sum_{x \in X_+(\mathcal{D}) \cup X} \pi(x)/n \right) \text{VS}\big(\mathcal{D}_+ \cup X; k\big), \tag{8}$$

where $n = |\mathcal{D}_+ \cup X|$. Our last step is to identify the batch X that maximizes this qVS metric, and we appeal to the methods described in Sect. 2.4 for this task. We show the pseudocode for our algorithm in Alg. 1.

3.2. Diverse Bayesian Optimization

Bayesian optimization (BayesOpt) (Garnett, 2022) is a framework for optimizing black-box functions. Given a domain $\mathcal{X}$, which can be either discrete or continuous, BayesOpt sets out to find the global optimum of an objective function f of interest:

$$x_* = \arg\max_{x \in \mathcal{D}} f(x).$$

Unlike active search (AS) where we work with binary labels, BayesOpt deals with real-valued labels y that are the outputs of the objective function f. Further, while AS focuses on finding many valuable data points within a search space, BayesOpt targets the singular, most valuable data point, tackling a different yet also relevant class of discovery problems commonly encountered in experimental design.

Similar to our discussion on AS, the pure-optimization formulation of BayesOpt often leads to overly exploitative strategies. Even if a BayesOpt algorithm can effectively identify the global optimum of the objective function, Maus et al. (2023) argued this "all-or-nothing" goal of finding a single best solution to a problem is undesirable in many scenarios. For example, when trying to discover metal-organic frameworks (MOFs) with high capacity for storage of toxic gasses, scientists may apply BayesOpt to identify a hypothetical MOF that, when simulated by a computer program, possesses a high storage capacity. However, this high-performing MOF may turn out to be infeasible to synthesize in practice due to having unrealistic physical attributes. This leads to wasted resources and efforts. Having a diverse set of MOF candidates would give the scientist higher chances at finding a synthesizable MOF that meats the target criteria.

Maus et al. (2023) proposed a modified formulation that aims to find many diverse solutions. Their framework involves maintaining a set of possible solutions that are of high quality *and* diverse, where diversity is enforced by constraining the solutions to be at least a pre-specified distance away from one another. Formally, denote by δ a distance function for an objective function f of interest, they seek a sequence of M solutions $\{x_1^*, x_2^*, \ldots, x_M^*\}$ such that:

$$\begin{aligned} x_1^* &= \arg\max_{x \in \mathcal{X}} f(x), \\ x_i^* &= \arg\max_{x \in \mathcal{X}} f(x) \text{ subject to } \delta(x_i^*, x_j^*) \geq \tau, \forall j < i, \end{aligned} \tag{9}$$

where τ is a user-specified distance threshold that controls the diversity of the resulting solutions. Maus et al. (2023) dubbed this formulation *rank-ordered* BayesOpt, as the solutions $x_1^*, x_2^*, \ldots, x_M^*$ are ranked in that each subsequent solution is constrained to be far away from those that precede it. The authors further proposed extending the trust region-based BayesOpt algorithm TuRBO (Eriksson et al., 2019) to this setting. TuRBO tackles high-dimensional problems via local optimization and consists of a set of local optimizers. Each optimizer maintains a trust region around a promising region, and the size of the region expands or shrinks based on optimization performance. (Local optimization is often accomplished with Thompson sampling (Russo et al., 2018) within each trust region.) This strategy is particularly amenable to the rank-ordered formulation above, as one could center a trust region around each member of the solution set x_i^*, and iteratively refine that member via local optimization while obeying the diversity constraints. The resulting algorithm, called ROBOT, was shown to be able to identify diverse and high-quality solutions in several tasks.

As mentioned, the goal of rank-ordered BayesOpt is to collect diverse data points that yield high objective values. We propose to also use our qVS for this task and seek:

$$X_* = \arg\max_{X \subset \mathcal{X}} \text{qVS}_\text{q}(X; k_\delta, f) = \arg\max_{X \subset \mathcal{X}} \Big(\sum_{x \in X} f(x)/M \Big) \text{VS}(X; k_\delta), \tag{10}$$

where $M = |X|$ is the number of solutions we wish to return to the user, and k_δ is the similarity function derived from the distance function δ (by, for example, inversing or subtracting from a maximum distance). This formulation removes the hyperparameter τ in Eq. (9) that constraints the solutions to be at least some distance away. We view this as a desirable feature in many instances, for example if the task of setting τ is not straightforward, which is often the case in high dimensions where reasoning about distances becomes challenging. By relying on the qVS to automatically balance diversity and quality among our solutions, we avoid this hyperparameter that might be difficult to tune. Furthermore, our algorithm can also take advantage of the trust region-based strategy of TuRBO. Specifically, at each iteration of the BayesOpt loop, we use Thompson sampling to generate

samples $\bar{f}$ of the Gaussian process fitted on the objective function f within a local region of the domain, and use these samples $\bar{f}$ in lieu of the actual $f(x)$ values to maximize the criterion in Eq. (10). The selected data points that maximize the criterion are chosen as our queries at the current iteration. We give the pseudocode for our algorithm in Alg. 2.

Lastly, we acknowledge two potential concerns. First, if the geometry of the domain of the problem in question is well understood, and the user is confident the constraints in the formulation of ROBOT in Eq. (9) are desirable, then that method should indeed be preferred to ours, as ROBOT specifically adheres to the constraints provided to it. Second, while the qVS and VS metrics do have a hyperparameter of their own—the order q—which controls the sensitivity to the diversity of the items, we argue that the basic form with $q = 1$ serves as a good starting point, and observe good performance of $q = 1$ in our experiments. Empirically, a user may tune q during a validation step prior to running actual experiments by observing the induced behavior on a toy example and adjusting q to match their preference.

4. Related Works

The quality-weighted diversity metric described in this paper, qVS, directly extends the Vendi score (Friedman & Dieng, 2023), which has found use in a wide range of applications (Pasarkar et al., 2023; Berns et al., 2023; Wu et al., 2023; Liu et al., 2024). The qVS is an alternative to the commonly used likelihoods of determinantal point processes (DPP), which can also account for the quality of a set of items but do not have a natural interpretation suitable for *evaluating* diversity and quality.

We use the qVS to tackle two specific experimental design problems, active search (AS) and Bayesian optimization (BayesOpt), which commonly model discovery tasks in science and engineering. While Garnett et al. (2012) originally formulated AS as maximizing the raw number of discovered targets, multiple subsequent works have extended the AS framework to settings where diversity is of concern. Vanchinathan et al. (2015) considered an Upper Confidence Bound-style algorithm (Auer, 2002) with an extra priority for diversity. Malkomes et al. (2021) proposed maximizing a coverage objective, defined as the sum of the volumes of hyperspheres drawn around the targets discovered. This coverage metric encourages the queries to be far away from one another. Nguyen & Garnett (2023) considered a multiclass setting and opted for a metric that rewards diversity in the labels.

In BayesOpt, diversity has been artificially induced to aid optimization, commonly via a DPP (Wang et al., 2018; Nava et al., 2022). Maus et al. (2023), on the other hand, directly targeted discovering diverse, high-quality solutions. As discussed in Sect. 3, they extended the state-of-the-art TuRBO algorithm by constraining the current solutions to be some distance away from one another. The resulting BayesOpt method ROBOT, as well as methods for AS mentioned earlier, crucially depend on accurately specifying the constraints to achieve the desired diversity. However, these distance constraints can be challenging to specify and enforce accurately, especially in high dimensions or with structured data. The qVS, in addition to being interpretable, offers a more flexible approach to diverse experimental design: instead of employing hard constraints on how far apart the solutions should be, we rely on the qVS to balance quality and diversity. The qVS, as we will show in Sect. 3, can be flexibly applied to many settings, including settings with non-continuous data for which TuRBO and ROBOT aren't applicable.

5. Experiments

We now present results from our numerical experiments, comparing active search (AS) and Bayesian optimization (BayesOpt) performance of our methods against a wide range of baselines. In each experimental setting, we average results across 10 repeats with different initial data, chosen uniformly at random from the search space (these sets of 10 different initial data sets are shared across the methods). We briefly highlight the data sets used in our experiments below and include further experimental details in Appx. C.

5.1. Diverse Active Search

We first discuss AS, where our goal is to collect a diverse set of positive points in a binary setting. We study the performance of our method, which we call qVS-AS, under different orders $q \in \{0, 0.1, 0.5, 1, 2, \infty\}$, both in the search behavior as defined in Eq. (8) and in the evaluation metric. Again note that $q = 0$ gives us the traditional AS setting (Garnett et al., 2012), which counts the raw number of positives discovered and does not account for diversity. We also consider relevant active learning/search algorithms discussed in Sect. 4: Expected Coverage Improvement (ECI) (Malkomes et al., 2021) and SELECT (Vanchinathan et al., 2015). These methods come with their own hyperparameters to tune, and we only report the results obtained from the highest-performing hyperparameters.

We consider the molecular discovery problem studied by Mukadum et al. (2021), where our target is photoswitches (molecules that change their properties upon irradiation) in chemical databases that exhibit both desirable light absorbance and long half-lives. Roughly 36% of the molecules in the search space are targets. Finally, we have a materials discovery application where we search for alloys that can form valuable bulk metallic glasses with higher toughness and better wear resistance than crystalline alloys. This data

Table 1. Average Vendi Scores under different orders q across 10 repeated experiments of the molecular discovery problem with the photoswitch data; the best performance in each column is highlighted in **bold** (including ties). A star (*) superscript indicates that the reported result is chosen from the best hyperparameter in that setting.

method		Vendi Score						max. pairwise dist.	kernel matrix det.
		$q=0$	$q=0.1$	$q=0.5$	$q=1$	$q=2$	$q=\infty$		
random search		72.20 (1.96)	60.66 (1.78)	44.30 (1.34)	30.59 (0.98)	18.42 (0.70)	7.96 (0.43)	0.94 (0.00)	0.32 (0.01)
ECI*		67.10 (1.93)	56.65 (1.24)	42.07 (0.70)	29.70 (0.35)	18.52 (0.34)	8.64 (0.26)	**0.95 (0.00)**	0.37 (0.02)
SELECT *		125.90 (5.70)	97.33 (4.65)	58.07 (1.21)	30.09 (1.47)	12.53 (1.50)	4.40 (0.43)	0.95 (0.00)	0.06 (0.01)
qVS-AS	$q=0$	**167.50 (11.03)**	105.70 (6.32)	52.78 (2.39)	20.83 (0.45)	7.30 (0.18)	3.28 (0.03)	0.93 (0.00)	0.02 (0.02)
	$q=0.1$	139.60 (4.76)	**120.26 (3.80)**	**67.29 (1.55)**	31.71 (0.82)	12.68 (0.51)	5.19 (0.32)	0.95 (0.00)	0.03 (0.01)
	$q=0.5$	72.00 (1.22)	67.91 (1.11)	53.53 (0.83)	**39.85 (0.75)**	24.55 (0.69)	9.01 (0.28)	0.95 (0.00)	0.39 (0.01)
	$q=1$	42.50 (0.49)	41.39 (0.51)	38.11 (0.46)	34.73 (0.41)	**29.72 (0.36)**	14.46 (0.39)	0.95 (0.00)	0.74 (0.00)
	$q=2$	33.70 (0.74)	33.39 (0.73)	32.25 (0.71)	30.98 (0.68)	28.88 (0.65)	**17.37 (0.40)**	0.94 (0.00)	0.85 (0.00)
	$q=\infty$	21.90 (0.68)	21.84 (0.68)	21.59 (0.64)	21.29 (0.60)	20.75 (0.53)	15.93 (0.32)	0.94 (0.00)	**0.93 (0.00)**

Table 2. Average Vendi Scores under different orders q across 10 repeated experiments of the materials discovery problem with the bulk metal glass data; the best performance in each column is highlighted in **bold** (including ties). A star (*) superscript indicates that the reported result is chosen from the best hyperparameter in that setting.

method		Vendi Score						max. pairwise dist.	kernel matrix det.
		$q=0$	$q=0.1$	$q=0.5$	$q=1$	$q=2$	$q=\infty$		
random search		7.20 (0.86)	7.18 (0.85)	7.09 (0.82)	6.99 (0.79)	6.82 (0.74)	5.42 (0.41)	13.16 (0.62)	0.88 (0.08)
ECI*		25.00 (0.00)	24.89 (0.00)	24.46 (0.00)	23.92 (0.00)	22.85 (0.00)	12.85 (0.00)	15.35 (0.52)	0.99 (0.00)
SELECT *		143.90 (2.85)	130.63 (4.47)	99.59 (5.66)	69.57 (6.12)	36.19 (5.13)	9.67 (1.46)	12.42 (0.60)	0.00 (0.00)
qVS-AS	$q=0$	**186.00 (3.04)**	146.35 (4.20)	69.49 (5.14)	26.42 (3.86)	8.66 (1.45)	3.31 (0.36)	6.45 (0.54)	0.00 (0.00)
	$q=0.1$	174.30 (3.59)	**162.82 (4.05)**	120.07 (5.23)	75.92 (5.11)	31.67 (3.07)	8.13 (0.75)	10.49 (0.36)	0.00 (0.00)
	$q=0.5$	160.40 (3.07)	156.68 (3.08)	141.15 (3.13)	120.74 (3.20)	83.15 (3.14)	20.94 (1.19)	15.08 (0.39)	0.00 (0.00)
	$q=1$	153.70 (1.24)	151.93 (1.23)	**144.48 (1.19)**	**134.45 (1.17)**	113.21 (1.24)	35.32 (1.28)	18.58 (0.70)	0.31 (0.05)
	$q=2$	137.60 (1.72)	136.82 (1.70)	133.59 (1.65)	129.32 (1.57)	**120.19 (1.38)**	50.55 (0.65)	22.34 (0.35)	0.96 (0.00)
	$q=\infty$	65.40 (4.40)	65.34 (4.39)	65.11 (4.33)	64.82 (4.26)	64.25 (4.13)	**52.71 (2.50)**	**25.58 (0.17)**	**1.00 (0.00)**

set comprises 106 810 alloys from the materials literature (Kawazoe et al., 1997; Ward et al., 2016), approximately 4% of which exhibit glass-forming ability.

To first ensure the quality of our predictive model, we perform the following benchmarking experiments. In each experiment, we train the model on 100 random points from a data set. We then pick out the test points that yield the highest posterior probabilities $\Pr(y = 1 \mid x, \mathcal{D})$ and record the proportion of this set are positives. For each data set, we repeat this experiment 10 times and record the average precision-at-k, which ranges consistently from 80% to 100%, indicating that our model produces high-quality predictions and recovers pure sets of rare positives.

We use the VS of the collected positives in Eq. (6) as our performance metric and show in the first portion of Tabs. 1 and 2 the VS (of different orders q) achieved by each method, averaged across the 10 repeats. We see that our method qVS-AS performs well across the problems, achieving the highest VS in most cases; when it is not the best, it is typically a close second behind a method with tuned hyperparameters. Inspecting the performance of different realizations of qVS-AS under varying values of the order q, we observe a reasonable trend: qVS-AS with the order q matching that of the evaluation metric tends to perform the best; further, there is a smooth change in performance as we move across the different values of q, showcasing the ability of this hyperparameter to smoothly control our algorithm's behavior.

To further study the diversity of the data collected by each method, we consider two metrics that quantify the spread of the discovered targets: the maximum distance between any pair of positives discovered and the determinant of the kernel matrix of the collected positives (i.e., the squared volume spanned by the feature vectors of the selected data points with positive labels). The last two columns in Tabs. 1 and 2 show these results, where our method can again be observed to achieve consistently good performance.

While our main target application is scientific discovery problems such as these two tasks above, Appx. C includes a product recommendation problem where the goal is to assemble diverse recommendations to a user. Our method performs well in that setting as well, illustrating its empirical effectiveness across different applications. Appx. C further includes a visualization of our method's search, which appropriately balances between exploitation of a known target region and exploration the space to find more diverse targets.

5.2. Diverse Bayesian Optimization

We now present results from BayesOpt tasks, as formulated in Sect. 3. To study the performance of our method, qVS-

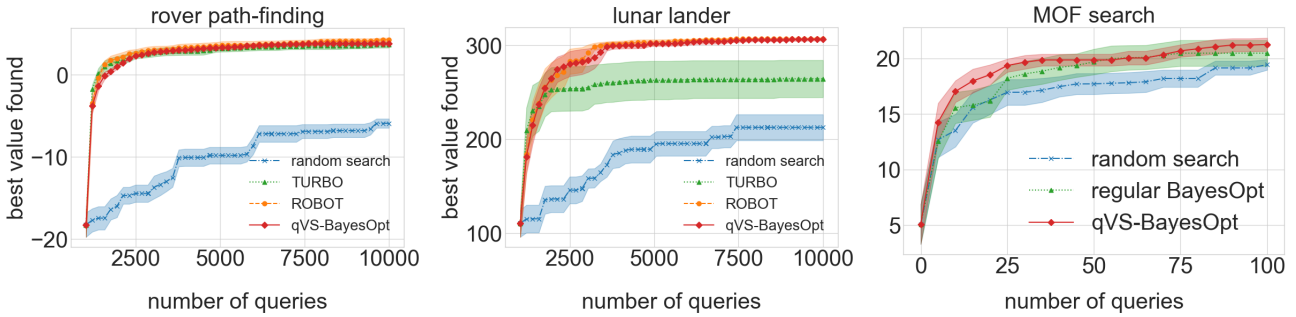
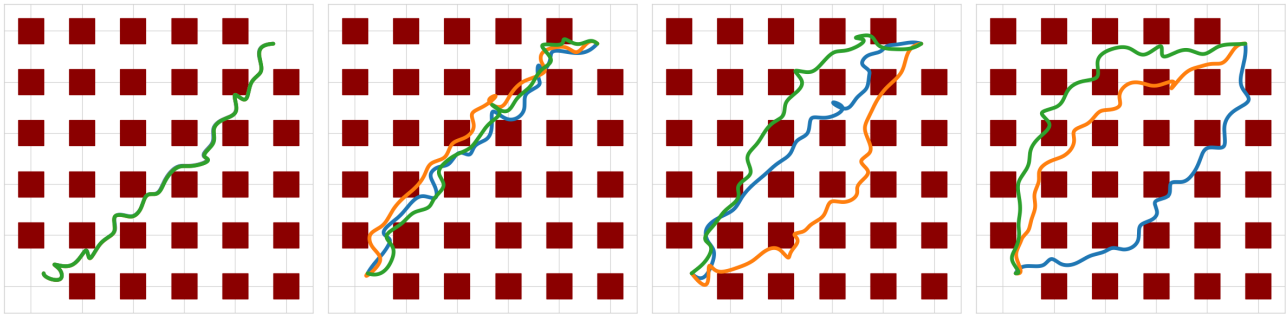


Figure 2. Average optimization performance and standard errors across 10 repeated experiments. Our method (shown in **red**) performs competitively across the different settings.



(a) TuRBO (paths overlapping) (b) qVS-BayesOpt with $q = 0.5$ (c) qVS-BayesOpt with $q = 1$ (d) qVS-BayesOpt with $q = 1.5$

Figure 3. Trajectories identified by various search methods in the rover path finding problem. Our method finds a diverse set of paths, whose diversity can be controlled using the order q of the qVS.

BayesOpt, we include as baselines (1) TuRBO (Eriksson et al., 2019), the diversity-blind algorithm upon which qVS-BayesOpt is based, (2) ROBOT (Maus et al., 2023), which also tackles diverse BayesOpt, and (3) a random search algorithm that uniformly samples its queries from the search space at random. ROBOT has a hyperparameter τ that controls the quality–diversity trade-off in its search, which we set to the values used in the original investigation by Maus et al. (2023). We test these methods on three optimization tasks: (1) the rover path-finding task involves optimizing the path of a mars rover while avoiding obstacles; (2) the lunar lander task from reinforcement learning where we aim to optimize the control policy for an autonomous vehicle to safely land on a given terrain; (3) the metal-organic framework (MOF) storage capacity optimization task, where we aim to identify the MOFs that have the highest storage capacity.

While the first two tasks are formulated as continuous optimization problems (60- and 12-dimensional, respectively), the third involves a discrete search space of structured data (a database of 1000 MOFs). Typically, to deal with structured data such as molecules in BayesOpt, one may train a deep learning model such as a variational autoencoder (VAE) (Kingma & Welling, 2013) to obtain a continuous embedding of the candidates one searches over (see Gómez-Bombarelli et al. (2018) for an example in drug discovery). From there, one can apply BayesOpt algorithms such as TuRBO to that continuous embedding. However, unlike drug-like molecules which have enjoyed enduring interest from the machine learning community, MOFs are relatively unexplored materials to which, to our knowledge, there does not exist any consistently suitable VAE that can be applied. We instead only work with a MOF-specific kernel function that operates on any given pair among the 1000 candidate MOFs. Without a continuous embedding, the trust region-based algorithm TuRBO, and thus its extension to diverse BayesOpt, ROBOT, cannot be applied to this MOF search task. Instead, we employ a simple Upper Confidence Bound (UCB) algorithm (Auer, 2002) as our baseline of traditional BayesOpt. To realize our algorithm with the qVS, we directly use the UCB score as our metric of quality in Eq. (10) which we use as our criterion for finding the next queries (instead of the sample $\bar{f}$ from Thompson sampling in TuRBO). Details of this algorithm is given in Appx. A.

Results from these experiments are reported in Fig. 2, where we show the highest objective value achieved across the 10 repeats. We see that our method qVS-BayesOpt ($q = 1$) performs well across the experiments and remains competitive against the state-of-the-art ROBOT in the two continuous optimization problems. Surprisingly, in the lunar lander and MOF search tasks, encouraging more diversity in our search

not only does not result in any slowdown in optimization progress compared to traditional BayesOpt, but actually leads to improved performance. We hypothesize this is because the search spaces in these two problems consist of many local optima in which exploitative BayesOpt algorithms could become trapped. To highlight our method's ability to identify diverse solutions, we first show in Fig. 3 the rover paths optimized by TuRBO, which targets pure optimization, in a representative run. (Here, the number of solutions to be returned to the user $M = 3$.) We see that these paths are effectively identical to and overlap one another. On the other hand, the other panels show the set of 3 solutions optimized by qVS-BayesOpt from the run using the same initial data as TuRBO above, which exhibit varying degrees of diversity corresponding to $q \in \{0.5, 1, 1.5\}$. We also include in Appx. C a comparison between the different algorithms in our MOF experiments, further illustrating our method's ability to make diverse discoveries.

6. Conclusion

We extended the Vendi scores to account for quality. We used these new quality-weighted Vendi scores, or qVS, to propose a unified framework for experimental design tasks to make diverse discoveries in discrete and continuous spaces. To optimize qVS, we proposed the sequential greedy strategy, widely used to optimize functions with diminishing returns. Our extensive experiments on scientific discovery problems show that the algorithms resulting from our framework can collect diverse, high-quality data, effectively balancing exploitation and exploration.

Impact Statement

Our work targets experimental design techniques that are often leveraged for scientific discovery. As argued in our paper, many of these techniques may grow exploitative and become stuck at local optima throughout the search space. We see our efforts to encourage more diverse sampling as a step towards addressing biased, potentially unrepresentative data assembled by these automatic data collection procedures. It is possible to use these techniques for purposes with negative consequences (e.g., optimizing adversarial attacks on deployed machine learning models (Suya et al., 2020; Wan et al., 2021)); however, we judge that the potential positive impacts of our methods for diverse experimental design outweigh the negative impacts.

Acknowledgements

Adji Bousso Dieng is supported by the NSF, Office of Advanced Cyberinfrastructure (OAC): #2118201 and by a Schmidt Sciences' AI2050 Early Career Fellowship. This paper is dedicated to Kwame Nkrumah.

References

Auer, P. Using Confidence Bounds for Exploitation-Exploration Trade-offs. *Journal of Machine Learning Research*, 3(Nov):397–422, 2002. (Cited on pgs. 6, 8, and 13.)

Berns, S., Colton, S., and Guckelsberger, C. Towards Mode Balancing of Generative Models via Diversity Weights. *arXiv preprint*, 2023. arXiv:2304.11961 [cs.LG]. (Cited on pg. 6.)

Cox, T. F. and Cox, M. A. *Multidimensional Scaling*. Chapman and Hall, 2001. (Cited on pg. 14.)

Eriksson, D., Pearce, M., Gardner, J., Turner, R. D., and Poloczek, M. Scalable Global Optimization via Local Bayesian Optimization. In *Advances in Neural Information Processing Systems*, volume 32, 2019. (Cited on pgs. 1, 5, 8, and 13.)

Friedman, D. and Dieng, A. B. The Vendi Score: A Diversity Evaluation Metric for Machine Learning. *Transactions on Machine Learning Research*, 2023. (Cited on pgs. 1, 2, and 6.)

Garnett, R. *Bayesian Optimization*. Cambridge University Press, 2022. (Cited on pg. 5.)

Garnett, R., Krishnamurthy, Y., Xiong, X., Schneider, J., and Mann, R. Bayesian Optimal Active Search and Surveying. In *Proceedings of the 29th International Conference on Machine Learning*, 2012. (Cited on pgs. 1, 6, and 12.)

Gómez-Bombarelli, R., Wei, J. N., Duvenaud, D., Hernández-Lobato, J. M., Sánchez-Lengeling, B., Sheberla, D., Aguilera-Iparraguirre, J., Hirzel, T. D., Adams, R. P., and Aspuru-Guzik, A. Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules. *ACS Central Science*, 4(2):268–276, 2018. (Cited on pg. 8.)

Hill, M. O. Diversity and Evenness: A Unifying Notation and Its Consequences. *Ecology*, 54(2):427–432, 1973. (Cited on pg. 2.)

Jiang, S., Malkomes, G., Converse, G., Shofner, A., Moseley, B., and Garnett, R. Efficient Nonmyopic Active Search. In *Proceedings of the 34th International Conference on Machine Learning*, pp. 1714–1723, 2017. (Cited on pg. 1.)

Kawazoe, Y., Yu, J.-Z., Tsai, A.-P., and Masumoto, T. (eds.). *Nonequilibrium Phase Diagrams of Ternary Amorphous Alloys*, volume 37A of *Condensed Matters*. Springer-Verlag, 1997. (Cited on pg. 7.)

Kingma, D. P. and Welling, M. Auto-Encoding Variational Bayes. *arXiv preprint*, 2013. arXiv:1312.6114 [stat.ML]. (Cited on pg. 8.)

Krause, A. and Guestrin, C. Near-optimal Observation Selection using Submodular Functions. In *Proceedings of the 21st AAAI Conference on Artificial Intelligence*, pp. 1650–1654, 2007. (Cited on pg. 4.)

Kulesza, A. and Taskar, B. Determinantal Point Processes for Machine Learning. *Foundations and Trends® in Machine Learning*, 5(2–3):123–286, 2012. (Cited on pg. 2.)

Liu, T.-W., Nguyen, Q., Dieng, A. B., and Gomez-Gualdron, D. Diversity-driven, efficient exploration of a mof design space to optimize mof properties: application to nh3 adsorption. *ChemRxiv preprint*, 2024. (Cited on pg. 6.)

Malkomes, G., Cheng, B., Lee, E. H., and Mccourt, M. Beyond the Pareto Efficient Frontier: Constraint Active Search for Multiobjective Experimental Design. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 7423–7434, 2021. (Cited on pgs. 1, 6, and 13.)

Maus, N., Wu, K., Eriksson, D., and Gardner, J. Discovering Many Diverse Solutions with Bayesian Optimization. In *Proceedings of the 26th International Conference on Artificial Intelligence and Statistics*, 2023. (Cited on pgs. 1, 5, 6, 8, 12, and 13.)

McInnes, L., Healy, J., and Melville, J. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv preprint*, 2018. arXiv:1802.03426 [stat.ML]. (Cited on pg. 13.)

Mukadum, F., Nguyen, Q., Adrion, D. M., Appleby, G., Chen, R., Dang, H., Chang, R., Garnett, R., and Lopez, S. A. Efficient Discovery of Visible Light-Activated Azoarene Photoswitches with Long Half-Lives Using Active Search. *Journal of Chemical Information and Modeling*, 61(11):5524–5534, 2021. (Cited on pgs. 6 and 13.)

Nava, E., Mutny, M., and Krause, A. Diversified Sampling for Batched Bayesian Optimization with Determinantal Point Processes. In *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics*, 2022. (Cited on pg. 6.)

Nemhauser, G. L., Wolsey, L. A., and Fisher, M. L. An Analysis of Approximations for Maximizing Submodular Set Functions—I. *Mathematical Programming*, 14(1): 265–294, 1978. (Cited on pg. 4.)

Nguyen, Q. and Garnett, R. Nonmyopic Multiclass Active Search with Diminishing Returns for Diverse Discovery. In *Proceedings of the 26th International Conference on Artificial Intelligence and Statistics*, 2023. (Cited on pgs. 1, 4, 6, and 14.)

Nguyen, Q., Modiri, A., and Garnett, R. Nonmyopic Multifidelity Acitve Search. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 8109–8118, 2021. (Cited on pg. 1.)

Nielsen, F. On a Variational Definition for the Jensen-Shannon Symmetrization of Distances Based on the Information Radius. *Entropy*, 23(4):464, 2021. (Cited on pg. 13.)

Pasarkar, A. and Dieng, A. B. Cousins Of The Vendi Score: A Family Of Similarity-Based Diversity Metrics For Science And Machine Learning. *arXiv preprint*, 2023. arXiv:2310.12952 [cs.LG]. (Cited on pgs. 1 and 3.)

Pasarkar, A. P., Bencomo, G. M., Olsson, S., and Dieng, A. B. Vendi Sampling For Molecular Simulations: Diversity As A Force For Faster Convergence And Better Exploration. *The Journal of Chemical Physics*, 159(14), 2023. (Cited on pg. 6.)

Rogers, D. and Hahn, M. Extended-Connectivity Fingerprints. *Journal of Chemical Information and Modeling*, 50(5):742–754, 2010. (Cited on pg. 13.)

Russo, D. J., Van Roy, B., Kazerouni, A., Osband, I., and Wen, Z. A Tutorial on Thompson Sampling. *Foundations and Trends® in Machine Learning*, 11(1):1–96, 2018. (Cited on pg. 5.)

Suya, F., Chi, J., Evans, D., and Tian, Y. Hybrid Batch Attacks: Finding Black-box Adversarial Examples with Limited Queries. In *29th USENIX Security Symposium (USENIX Security 20)*, pp. 1327–1344, 2020. (Cited on pg. 9.)

Vanchinathan, H. P., Marfurt, A., Robelin, C.-A., Kossmann, D., and Krause, A. Discovering Valuable Items from Massive Data. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1195–1204, 2015. (Cited on pgs. 6 and 13.)

Wan, X., Kenlay, H., Ru, R., Blaas, A., Osborne, M. A., and Dong, X. Adversarial Attacks on Graph Classifiers via Bayesian Optimisation. In *Advances in Neural Information Processing Systems*, volume 34, pp. 6983–6996, 2021. (Cited on pg. 9.)

Wang, Z., Garrett, C. R., Kaelbling, L. P., and Lozano-Pérez, T. Active model learning and diverse action sampling for task and motion planning. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4107–4114, 2018. (Cited on pg. 6.)

Ward, L., Agrawal, A., Choudhary, A., and Wolverton, C. A general-purpose machine learning framework for predicting properties of inorganic materials. *npj Computational Materials*, 2(1):1–7, 2016. (Cited on pgs. 7 and 13.)

Willett, P., Barnard, J. M., and Downs, G. M. Chemical Similarity Searching. *Journal of Chemical Information and Computer Sciences*, 38(6):983–996, 1998. (Cited on pg. 13.)

Wu, S., Lu, K., Xu, B., Lin, J., Su, Q., and Zhou, C. Self-Evolved Diverse Data Sampling for Efficient Instruction Tuning. *arXiv preprint*, 2023. arXiv:2311.08182 [cs.CL]. (Cited on pg. 6.)

Xiao, H., Rasul, K., and Vollgraf, R. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint*, 2017. arXiv:1708.07747 [cs.LG]. (Cited on pgs. 13 and 14.)

Algorithm 1 qVS-AS for diverse active search

inputs observations $\mathcal{D}$, query batch size n
returns query batch X of size n maximizing Eq. (8)
$X \leftarrow \emptyset$ ▷ sequentially built from the empty set
for $i \leftarrow 1, \ldots, n$ **do**
  for $x \in \mathcal{X} \setminus (\mathcal{D} \cup X)$ **do**
    $\alpha(x) = \mathrm{qVS}\Big(X_+(\mathcal{D}) \cup X \cup \{x\}; k, p\Big)$ ▷ each candidate is scored by the qVS if added to X
  end for
  $X \leftarrow X \cup \{\arg\max_{x \in \mathcal{X} \setminus (\mathcal{D} \cup X)} \alpha(x)\}$ ▷ add the candidate yielding the largest qVS to X
end for

Algorithm 2 qVS-BayesOpt with TuRBO for diverse Bayesian optimization

inputs observations $\mathcal{D}$, number of trust regions M, query batch size n
returns query batch X of size n maximizing Eq. (10)

for $m \leftarrow 1, \ldots, M$ **do** ▷ generate high-quality candidates in each trust region with TuRBO
  $\overline{X}_m = \mathrm{TuRBO}_m(\mathcal{D})$
end for
$\overline{X} \leftarrow \cup_{m=1}^{M} \overline{X}_m$ ▷ merge all high-quality candidates

$X \leftarrow \emptyset$ ▷ sequentially built from the empty set
for $i \leftarrow 1, \ldots, n$ **do**
  for $x \in \overline{X} \setminus X$ **do**
    $\alpha(x) = \mathrm{qVS}\Big(X \cup \{x\}; k, \bar{f}\Big)$ ▷ each candidate is scored by the qVS if added to X
  end for
  $X \leftarrow X \cup \{\arg\max_{x \in \overline{X} \setminus X} \alpha(x)\}$ ▷ add the candidate yielding the largest qVS to X
end for

A. Details on Search & Optimization Algorithms

We now describe our qVS-AS and qVS-BayesOpt algorithms. Our algorithms, as well as the baselines, use the same predictive models throughout the experiments. In active search (AS), we reuse the k-nearest neighbors classification model by Garnett et al. (2012), which computes the probability that a given unlabeled data point has a positive label as the proportion of positive members among its labeled nearest neighbors. Our Bayesian optimization (BayesOpt) experiments extend the software published by Maus et al. (2023), which uses a Gaussian process as the predictive model, as standard in BayesOpt tasks.

Alg. 1 shows our qVS-AS algorithm for AS, which sequentially builds the batch of queries at a given search iteration by greedily adding in candidates that lead to the largest increase in the qVS. Alg. 2 similarly shows our qVS-BayesOpt

Algorithm 3 qVS-BayesOpt for structured data within a discrete search space

inputs observations $\mathcal{D}$, query batch size n
returns query batch X of size n maximizing Eq. (10) in the discrete case
$X \leftarrow \emptyset$ ▷ sequentially built from the empty set
for $i \leftarrow 1, \ldots, n$ **do**
  for $x \in \mathcal{X} \setminus (\mathcal{D} \cup X)$ **do**
    $\alpha(x) = \mathrm{qVS}\Big(\mathcal{D} \cup X \cup \{x\}; k, \mathrm{UCB}\Big)$ ▷ each candidate is scored by the qVS if added to X
  end for
  $X \leftarrow X \cup \{\arg\max_{x \in \mathcal{X} \setminus (\mathcal{D} \cup X)} \alpha(x)\}$ ▷ add the candidate yielding the largest qVS to X
end for

algorithm for continuous BayesOpt, which builds on top of TuRBO (Eriksson et al., 2019) with Thompson sampling to generate an initial set of high-quality candidates and uses the qVS to select the final batch of queries. Finally, Alg. 3 shows qVS-BayesOpt for our metal–organic framework search experiments, where the qVS is computed with the commonly used UCB (Auer, 2002) score (with the exploration–exploitation trade-off hyperparameter $\beta = 2$) instead of Thompson samples to deal with a discrete search space of structured data.

We also detail how we set the hyperparameters of the active learning baselines used in our experiments.

- The policy by Vanchinathan et al. (2015) has two hyperparameters: λ encourages diversity (measured by the logdet of the Gram matrix of the collected data), and β_t encourages UCB-style exploration of the space. We run all variants of this policy with $\lambda \in \{0.25, 0.5, 0.75\}$ and $\beta_t \in \{0.1, 0.3, 1, 3, 10\}$ and report the best performance.
- The policy by Malkomes et al. (2021) has a hyperparameter r, which sets the radius of the spheres that compute their coverage measure. As we have access to similarity scores (between 0 and 1) among points in each of our data sets, the spheres in this method cover points that are sufficiently similar (similarity greater than threshold $1 - r$) to a collected target. We run all variants of this policy with $r \in \{0.25, 0.5, 0.75\}$ and report the best performance.

B. Details on Data Sets

We describe the data sets used in our numerical experiments in this section.

- The FashionMNIST data set (Xiao et al., 2017) is used in its entirety with 70,000 data points. We then compute a 2-dimensional embedding using UMAP (McInnes et al., 2018). We use this embedding as the features of the data points when training the predictive k-NN (with $k = 5$), and the radial basis function (RBF) kernel with a length scale $\ell = 1$ for calculating the Vendi Scores.
- For the photoswitch experiments, we obtain the data from Mukadum et al. (2021), which contains the Morgan fingerprint (Rogers & Hahn, 2010) for each molecule in the database, from which we compute the Tanimoto similarity coefficient (Willett et al., 1998) between each pair of molecules. These coefficients are used by the k-NN predictive model ($k = 10$) and the VS calculations. The distance metric used to calculate maximum pairwise distances in our experiments is defined as the corresponding coefficient subtracted from 1.
- Following Ward et al. (2016), we represent each data point in the bulk metal glass data set with various physical attributes that were found to be informative in predicting glass-forming ability. Each feature is subsequently scaled to range between 0 and 1. We set $k = 100$ for the k-NN model and use the RBF kernel with $\ell = 0.1$ for the VS kernel.
- The metal–organic framework (MOF) data is from an ongoing collaboration with a group of materials scientists. The MOF-specific kernel used in that collaboration is re-implemented in our experiments, for both the Gaussian process and the VS. This kernel is a weighted sum of an RBF kernel operating on numerical features of the MOFs such as the dimensions of the MOF pore (size, volume, etc.), two Tanimoto kernels defined similarly as that used in the photoswitch experiments operating on the MOFs' node and linker molecules respectively, and a Jensen–Shannon kernel (Nielsen, 2021) operating on the distributions of the pores. The weights of these individual kernels are optimized to maximize the marginal likelihood of the training data when training the Gaussian process. For calculating the Vendi Scores, each weight is set to be 0.25 so that our diversity metric stays consistent through the search. (The data will be made public in the near future.)

Further, we follow Maus et al. (2023) in implementing the rover path-finding and lunar lander problems.

C. More Experimental Results

We first discuss the design of our numerical experiments in Sect. 5. In active search experiments, each experimental run consists in total of 200 queries, divided into 40 batches of 5 queries. The first 2 Bayesian optimization problems involve running a BayesOpt algorithm for 10,000 queries in batches of 32, while the first 1,024 are sampled uniformly at random. Due to the limited size of the MOF data (1000 data points), each run in that setting consists of 100 queries, starting with 2 initial random observations.

Table 3. Average Vendi Scores under different orders q across 10 repeated experiments of the product recommendation problem with the FashionMNIST data; the best performance in each column is highlighted in **bold** (including ties). A star (*) superscript indicates that the reported result is chosen from the best hyperparameter in that setting.

method		Vendi Score										max. pairwise dist.		kernel matrix det.	
		$q=0$		$q=0.1$		$q=0.5$		$q=1$		$q=2$		$q=\infty$					
random search		19.70	(1.64)	19.08	(1.45)	18.31	(1.30)	17.81	(1.20)	16.94	(1.02)	11.62	(1.20)	9.11	(1.07)	**0.64**	**(0.10)**
ECI*		32.80	(1.39)	32.79	(1.39)	32.76	(1.38)	32.73	(1.37)	32.67	(1.35)	**29.23**	**(1.34)**	7.77	(1.03)	0.48	(0.08)
SELECT *		135.40	(6.56)	65.89	(3.67)	45.65	(3.39)	31.88	(3.08)	20.17	(2.43)	7.59	(0.81)	1.79	(0.19)	0.00	(0.00)
qVS-AS	$q=0$	**155.00**	**(9.92)**	**69.90**	**(4.12)**	40.81	(2.09)	21.92	(0.79)	10.58	(0.36)	4.39	(0.12)	1.45	(0.19)	0.00	(0.00)
	$q=0.1$	76.40	(4.65)	59.81	(3.54)	43.59	(2.69)	34.47	(2.25)	24.68	(1.87)	8.94	(0.64)	5.39	(0.60)	0.00	(0.00)
	$q=0.5$	65.40	(3.47)	60.67	(3.02)	**54.88**	**(2.74)**	**49.92**	**(2.49)**	**43.80**	**(2.18)**	22.34	(1.23)	7.90	(0.81)	0.00	(0.00)
	$q=1$	61.40	(3.77)	57.77	(3.36)	52.94	(3.11)	48.75	(2.86)	43.52	(2.53)	24.49	(1.25)	7.93	(0.86)	0.00	(0.00)
	$q=2$	46.70	(4.23)	44.09	(4.04)	41.13	(3.69)	38.47	(3.39)	35.03	(3.06)	21.75	(1.98)	8.32	(0.84)	0.04	(0.02)
	$q=\infty$	25.50	(2.27)	25.29	(2.20)	24.51	(1.95)	23.71	(1.71)	22.59	(1.40)	19.50	(0.98)	**9.75**	**(0.77)**	0.50	(0.01)

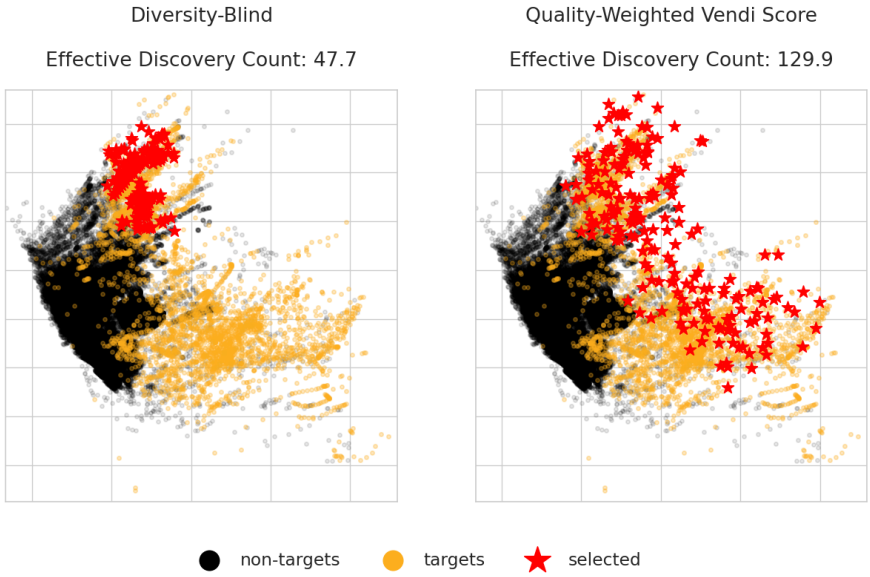


Figure 4. Data points collected by diversity-blind search and our diversity-aware policy in the materials discovery problem with bulk metal glasses. Our method appropriately balances between exploring the search space and focusing on regions containing positive data, and discovers more effective positives as a result.

Following Nguyen & Garnett (2023), we use the FashionMNIST data set (Xiao et al., 2017) of 70,000 images of articles of clothing to simulate a product recommendation problem. Here, we assume that a user is looking for, unbeknownst to the recommendation engine, t-shirts and tops (members of class 0, one-tenth of the data set) while shopping online, and the goal is to assemble a diverse set of products belonging to this unknown class. The results are shown in Tab. 3, where we once again observe the relatively consistent performance of our method qVS-AS.

To visually illustrate our method's ability to assemble diverse data, we show in Fig. 4 the locations of the queries made by diversity-blind AS and our method, within the two-dimensional embedding of the bulk metal glass data set computed by performing PCA on the features. We see that the overly exploitative diversity-blind search simply focuses on a small portion of the search space, while our method qVS-AS (with $q = 1$) is able to thorough explore the different regions of positives. We also show the VS of the collected positives of the two policies (interpreted as the effective discovery count), where our policy clearly outperforms diversity-blind search.

We further seek to visualize the search behaviors of our method by first using the multidimensional scaling technique (Cox & Cox, 2001) to compute a 2-dimensional embedding from the kernel matrix of the candidates within the database. This embedding is shown in the first panel of Fig. 5, where the scatter points' colors and opacity levels are set based on the corresponding MOFs' storage capacity levels (the objective value to be maximized). We then mark the MOFs that are

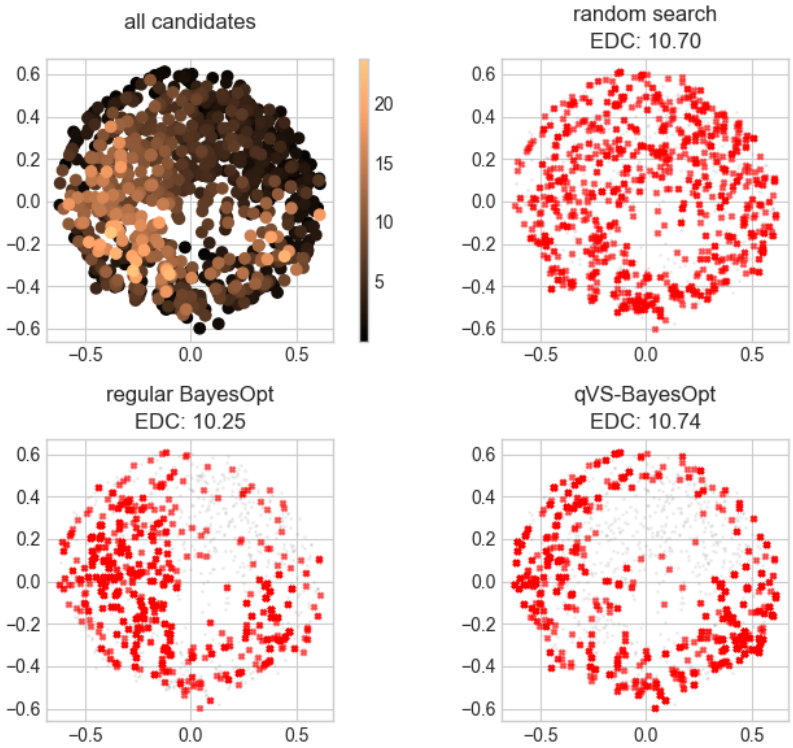


Figure 5. Data points collected by each search strategy in the MOF search problem across the repeats, illustrated as **red** x's. (All data points in the search space are visualized in the first panel.) Our method focuses on specific, high-performing regions compared to the random search, while exploring the space more evenly compared to regular Bayesian optimization.

selected by each method in the remaining panels and observe a number of distinct trends: compared to random search, regular BayesOpt focuses on the lower-left region where the storage capacity is high; qVS-BayesOpt, on the other hand, further inspects the lower-right portion, which also contains high-capacity MOFs but in fewer numbers. It is exactly this diverse sampling strategy that we hoped to achieve with the qVS. We further compute a metric similar to the effective discovery count (EDC) whereby a MOF is classified as "good" if it yields a storage capacity (the objective value to be maximized) of at least 15, and the VS of these good MOFs collected by each policy is reported. We see that our qVS method outperforms both baselines; interestingly, regular BayesOpt yields a lower EDC than even random search—another indication of the failure mode of its overly exploitative strategy. To study the effect of the order q on the algorithm's performance, we include in Tab. 4 the best objective value found under $q \in \{0, 0.1, 0.5, 1, 2, \infty\}$. We see that $q = 0.1$ yields the best performance, while some values of q lead to even worse performance than regular BayesOpt (when $q = 0$). This behavior indicates the importance of setting q appropriately; an interesting future direction could be to dynamically set q based on search progress.

To conclude our analysis, we include another relevant metric in the MOF search application, which is the percentage energy penalty incurred when the stored toxins are eventually released for disposal, which ranges from 0 to 1 and a lower value indicates higher energy efficiency. Here, Fig. 6 shows the average penalty across the optimization runs discussed above by the three algorithms, where we once again observe that (1) regular BayesOpt could fail to compete against even random search and (2) qVS-BayesOpt outperforms the two baselines.

D. Software

Code can be found at `https://github.com/vertaix/Quality-Weighted-Vendi-Score`.

	best storage value
$q = 0$	20.47 (1.16)
$q = 0.1$	22.07 (0.81)
$q = 0.5$	20.07 (0.61)
$q = 1$	21.24 (0.59)
$q = 2$	20.23 (0.56)
$q = \infty$	20.23 (0.56)

Table 4. Average storage capacity values (higher is better) and standard errors of the best MOFs found by our algorithm under different orders q. Here, $q = 0$ corresponds to regular, diversity-blind Bayesian optimization.

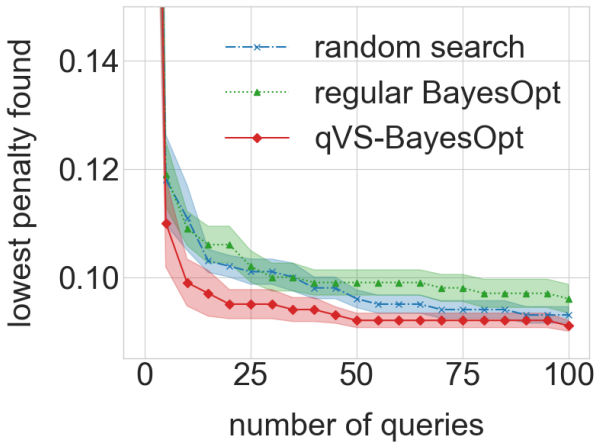


Figure 6. Energy penalty to release stored toxins (lower is better) of the best MOFs found by different algorithms. Our qVS-BayesOpt ($q = 1$) outperforms regular BayesOpt and random search.