BOtied: Multi-objective Bayesian optimization with tied multivariate ranks

Ji Won Park [*] [1] Nataša Tagasovska [*] [1] Michael Maser [1] Stephen Ra [1] Kyunghyun Cho [1 2 3]

Abstract

Many scientific and industrial applications require the joint optimization of multiple, potentially competing objectives. Multi-objective Bayesian optimization (MOBO) is a sample-efficient framework for identifying Pareto-optimal solutions. At the heart of MOBO is the acquisition function, which determines the next candidate to evaluate by navigating the best compromises among the objectives. Acquisition functions that rely on integrating over the objective space scale poorly to a large number of objectives. In this paper, we show a natural connection between the non-dominated solutions and the highest multivariate rank, which coincides with the extreme level line of the joint cumulative distribution function (CDF). Motivated by this link, we propose the CDF indicator, a Pareto-compliant metric for evaluating the quality of approximate Pareto sets, that can complement the popular hypervolume indicator. We then introduce an acquisition function based on the CDF indicator, called BOtied. BOtied can be implemented efficiently with copulas, a statistical tool for modeling complex, high-dimensional distributions. Our experiments on a variety of synthetic and real-world experiments demonstrate that BOtied outperforms state-of-the-art MOBO algorithms while being computationally efficient for many objectives.

1. Introduction

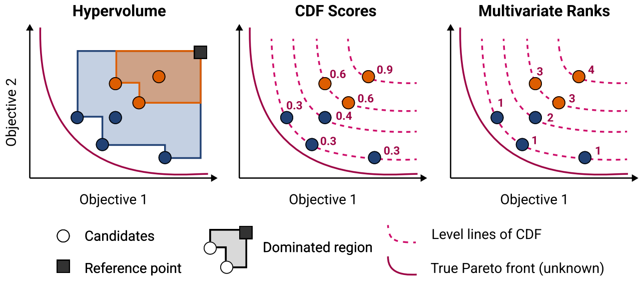


Figure 1. Illustration of the conceptual link between the empirical Pareto front probed by the HV indicator and innermost level line of the CDF probed by the BOtied CDF indicator. The blue set of candidates dominates the orange. The HV indicator is consistent with this ordering; the area of the box dominated by the blue set is greater. The BOtied CDF values and associated multivariate ranks also favor the blue.

Bayesian optimization (BO) has demonstrated promise in a variety of scientific and industrial domains where the goal is to optimize an expensive black-box function using a limited number of potentially noisy function evaluations (Romero et al., 2013; Calandra et al., 2016; Kusne et al., 2020; Shields et al., 2021; Zuo et al., 2021; Bellamy et al., 2022; Park et al., 2022). In BO, we fit a probabilistic surrogate model on the available observations so far. Based on the model, the acquisition function determines the next candidate to evaluate by balancing exploration (evaluating highly uncertain candidates) with exploitation (evaluating designs believed to be optimal). Often, applications call for the joint optimization of $M{\geq}2$ multiple, potentially competing objectives (Marler & Arora, 2004; Jain et al., 2017; Tagasovska et al., 2022). Unlike in single-objective settings, a single optimal solution may not exist and we must identify a set of solutions that represents the best compromises among the multiple objectives. The acquisition function in multi-objective Bayesian optimization (MOBO) navigates these trade-offs as it guides the optimization toward regions of interest.

Many bona fide MO acquisition functions without scalarization involve high-dimensional integrals and scale poorly with increasing M. Moreover, improvement-based acquisition functions including some variants of random scalarization are sensitive to non-informative monotonic transformations of the objectives. This is a pain point for many practical applications. For instance, in biochemistry, the dissociation constant K_D is typically expressed in terms of its log transformation, the $\mathrm{p}K_D \equiv -\log_{10}(K_D)$. It would be desirable to work with acquisition functions that are invariant to the choice of such unit conversions.

[*] Equal contribution [1] Prescient Design, Genentech, South San Francisco, USA [2] Department of Computer Science, New York University, New York City, USA [3] Center for Data Science, New York University, New York City, USA. Correspondence to: Ji Won Park <park.ji_won@gene.com>, Nataša Tagasovska <natasa.tagasovska@roche.com>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

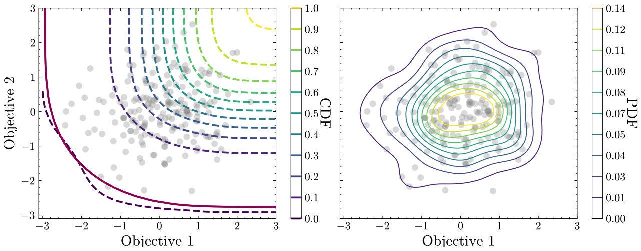


Figure 2. Level lines of the CDF (left) and the PDF (right) from kernel density estimation based on 200 observations (gray dots). The zero level line of the CDF closely traces the true Pareto front (solid red curve).

To address these challenges, we propose BOtied[1], a novel acquisition function based on multivariate ranks. We show that BOtied has the desirable property of being invariant to relative rescaling or monotonic transformations of the objectives. While it maintains the multivariate structure of the objective space, its implementation has highly favorable time complexity and we report wall-clock time competitive with random scalarization.

In Figure 1, we present the intuition behind BOtied. Consider a minimization setup for $M{=}2$ where we seek to identify solutions on the true Pareto front (solid red curves), unknown to us. Suppose we have many candidates, represented as circular posterior blobs in the objective space, where the predictive distributions have been output by our surrogate model. For simplicity, assume the uncertainties (sizes of blobs) are comparable among the candidates. How do we estimate the quality of our solutions (i.e., each candidate set's proximity to the true Pareto front)? One quality indicator is the hypervolume (HV; Zitzler et al., 2003), defined as the size of the polytope bounded from above by a predefined reference point and dominated by the candidate set (shaded areas in the leftmost panel). When one candidate set dominates another, its HV is greater. We can visually confirm that the HV of the blue Pareto approximation is greater than that of the orange.

Next, let us adopt the related, but distinct, perspective of multivariate ranking (see, e.g., Ghosal & Sen, 2022). Define the random objective vector $Y = [f_1(X), \ldots, f_M(X)]$ taking values in $\mathbb{R}^M$, which is the result of applying the objective function $f : \mathbb{R}^d \to \mathbb{R}^M$ on the random design variable X that takes values in $\mathbb{R}^d$. How do we compare two realizations of Y, say $\boldsymbol{y}$ and $\boldsymbol{y}'$?

Ranking vectors is non-trivial, as there is no natural ordering in Euclidean spaces when $M{\geq}2$. We propose to use the joint cumulative distribution function (CDF), defined as the probability of $\boldsymbol{y}$ being weakly dominated: $F_Y(\boldsymbol{y}) = \mathbb{P}(Y_1{\leq}y_1, \ldots, Y_M{\leq}y_M)$, where $\boldsymbol{y} = [y_1, \ldots, y_M] \in \mathbb{R}^M$. The CDF formalizes the rank ordering of vectors as weak dominance in the joint minimization of M objectives (Binois et al., 2015). Specifically, The CDF scores and their α-level lines $L_\alpha^{F_Y} = \{\boldsymbol{y} : F_Y(\boldsymbol{y}) = \alpha\}$ are depicted in the middle panel of Figure 1 for multiple values of $\alpha \in [0, 1]$. All candidates with equal multivariate rank, or ties, lie on the same level line, as shown in the rightmost panel.

Multivariate ranking via the CDF can be understood in relation to the associated probability density function (PDF), as the CDF is the integral of the PDF. Figure 2 shows the CDF and the associated PDF side by side for a bi-objective setting ($M{=}2$). The right panel shows the PDF fit on 200 outcome samples (gray dots) via kernel density estimation, where the outcome samples were drawn from an elliptical Gaussian. The left panel shows the level lines of the corresponding CDF. The α-level lines converge to the approximate Pareto front as $\alpha \to 0$. The lowermost level line ($\alpha{\approx}0$) closely traces the convex shape of the true Pareto front shown as the solid red curve.

Contributions Motivated by the interpretation of multivariate ranks as a MO indicator, we make the following contributions: (i) We propose a new Pareto-compliant performance criterion, the CDF indicator (section 3); (ii) We propose a scalable and robust acquisition function based on the CDF and associated multivariate ranks, which we call BOtied (section 4); and (iii) We release the full codebase implementing our evaluations of MOBO acquisitions in a variety of synthetic and real-world data scenarios (section 5).

2. Related work

MO indicators and acquisition functions. A computationally attractive approach to MOBO scalarizes the objectives with random preference weights (Knowles, 2006; Paria et al., 2020) and applies a single-objective acquisition function. The distribution of the weights, however, may be insufficient to encourage exploration when there are many objectives with unknown scales.

Alternatively, we may preserve the MO structure by seeking improvement on a set-based performance metric, such as the HV indicator (Embrechts et al., 2003) or the R2 indicator (Deutz et al., 2019a;b). Improvement-based acquisition functions such as the expected hypervolume improvement (EHVI; Emmerich et al., 2011; Daulton et al., 2020; 2021) are sensitive to the rescaling of the objectives, which may carry drastically different natural units. In particular, computing the HV has time complexity that is super-polynomial in the number of objectives, because it entails computing the volume of an irregular polytope (Yang et al., 2019). Despite the efficiency improvement achieved by box decomposition algorithms (Dächert et al., 2017; Yang et al., 2019), HV computation remains slow when $M{>}4$.

[1] The name stems from non-dominated solutions considered to be ”tied.”

Another class of acquisition strategies is entropy search, which focuses on maximizing the information gain from the next observation (Villemonteix et al., 2009; Hennig & Schuler, 2012; Hoffman & Ghahramani, 2015; Shah & Ghahramani, 2015; Hernández-Lobato et al., 2016b; Belakaria et al., 2019; Tu et al., 2022). Entropy searches are commonly implemented in box decompositions as well, but are costly to evaluate without using more tractable bounds to serve as approximations.

Multivariate ranking. The scale-invariant properties of ranking makes it an attractive tool for optimization. Binois et al. (2015) relates the Pareto front to the extreme level line of the CDF, F_Y. Considering ranking dominance as an alternative to Pareto dominance, Kukkonen & Lampinen (2007) propose computing the ranks of individual objectives separately and combining them post-hoc with a simple aggregation function (min, max, average) to obtain the overall fitness value for a given candidate. Binois et al. (2020) explores the question of how to choose from the set of non-dominated solutions, which grows with M, and makes a game-theoretic argument for how to make the compromise. In particular, they define trade-offs in the copula space, which is the scale-invariant rank transformation of the original objective function. For single-objective BO, Picheny et al. (2019) propose Ordinal BO, which uses a Gaussian process (GP) surrogate that is only sensitive to the rankings of the inputs and objective values. Notably, this method can robustly handle ill-conditioned and multi-modal distributions in the objective function values, for which GP models are known to often fail. Similarly, Eriksson & Poloczek (2021) use the rank transformations to magnify values at the end of the observed ranges. To our knowledge, however, our work is the first to incorporate multivariate rankings enabled by the *joint* CDF into a MOBO algorithm. We explicitly account for the structure of the M-variate objective distribution in identifying the full Pareto front.

A more detailed overview and positioning of BOtied with respect to the MO literature can be found in Appendix A.

3. Background

3.1. Bayesian Optimization

Bayesian optimization (BO) is a popular technique for sample-efficient black-box optimization (see Shahriari et al., 2015; Frazier, 2018, for a review). In a single-objective setting, suppose our objective $f : \mathcal{X} \to \mathbb{R}$ is a black-box function of the design space $\mathcal{X}$ that is expensive to evaluate. Our goal is to efficiently identify a design $\boldsymbol{x}^\star \in \mathcal{X}$ minimizing[2] f. BO leverages two tools, a probabilistic surrogate model and a utility function, to trade off exploration (evaluating highly uncertain designs) and exploitation (evaluating designs believed to minimize f) in a principled manner.

For each iteration t, we have a dataset $\mathcal{D}_t = \{(\boldsymbol{x}^{(1)}, y^{(1)}), \cdots, (\boldsymbol{x}^{(N_t)}, y^{(N_t)})\}$, where for each $n \in [N_t]$, $y^{(n)}$ is a potentially noisy observation of $f(\boldsymbol{x}^{(n)})$. We first infer the posterior distribution $p(\hat{f}|\mathcal{D}_t)$, which serves as a cheap approximation of f. Next, we introduce a utility function $u : \mathcal{X} \times \mathcal{F} \times \mathscr{D}_t :\to \mathbb{R}$. The acquisition function $a(\boldsymbol{x})$ is simply the expected utility of $\boldsymbol{x}$ with respect to our current belief about f:

$$a(\boldsymbol{x}) = \int u(\boldsymbol{x}, \hat{f}, \mathcal{D}_t)\, p(\hat{f}|\mathcal{D}_t)\, d\hat{f}. \tag{1}$$

For example, we obtain the expected improvement (EI) acquisition function if we take $u_{\text{EI}}(\boldsymbol{x}, \hat{f}, \mathcal{D}) = [\hat{f}(\boldsymbol{x}) - \max_{(\boldsymbol{x}', y') \in \mathcal{D}_t} y']_+$, where $[\cdot]_+ = \max(\cdot, 0)$ (Močkus, 1975; Jones et al., 1998). We select a maximizer of a as the new design, evaluate f, and append the observation to the dataset. The surrogate is then refit on the expanded dataset and the procedure repeats.

3.2. Multi-objective optimization

When there are multiple objectives of interest, a single best design may not exist. Suppose there are M objectives, $f : \mathcal{X} \to \mathbb{R}^M$. The goal of MOBO is to identify the set of *Pareto-optimal* solutions such that improving one objective within the set leads to worsening another. We say that $\boldsymbol{x}$ dominates $\boldsymbol{x}'$, or $f(\boldsymbol{x}) \prec f(\boldsymbol{x}')$, if $f_m(\boldsymbol{x}) \le f_m(\boldsymbol{x}')$ for all $m \in [M]$ and $f_m(\boldsymbol{x}) < f_m(\boldsymbol{x}')$ for some m. The set of *non-dominated* solutions $\mathscr{X}^*$ is defined in terms of the Pareto front $\mathcal{P}^*$:

$$\mathscr{X}^\star = \{\boldsymbol{x} : f(\boldsymbol{x}) \in \mathcal{P}^\star\},$$

where $\mathcal{P}^\star = \{f(\boldsymbol{x}) \ : \boldsymbol{x} \in \mathcal{X},\ \nexists\, \boldsymbol{x}' \in \mathcal{X}\ s.t.\ f(\boldsymbol{x}') \prec f(\boldsymbol{x})\}$.

MOBO algorithms typically aim to identify a finite subset of $\mathscr{X}^\star$, which may be infinite, within a given budget of function evaluations.

Hypervolume One way to measure the quality of an approximate Pareto front $\mathcal{P}$ is to compute the hypervolume (HV) $\text{HV}(\mathcal{P}|\boldsymbol{r}_{\text{ref}})$ of the polytope bounded from above by $\mathcal{P}$ and from below by $\boldsymbol{r}_{\text{ref}}$, where $\boldsymbol{r}_{\text{ref}} \in \mathbb{R}^M$ is a user-specified *reference point*. More specifically, the HV indicator for a set A is

$$I_{\text{HV}}(A) = \int_{\mathbb{R}^M} \mathbb{I}[A \preceq \boldsymbol{y} \preceq \boldsymbol{r}_{\text{ref}}] d\boldsymbol{y}. \tag{2}$$

We obtain the expected hypervolume improvement (EHVI) acquisition function if we take

$$\begin{aligned} u_{\text{EHVI}}(\boldsymbol{x}, \hat{f}, \mathcal{D}) = \text{HVI}(\mathcal{P}', \mathcal{P}|\boldsymbol{r}_{\text{ref}}) = \\ [I_{\text{HV}}(\mathcal{P}'|\boldsymbol{r}_{\text{ref}}) - I_{\text{HV}}(\mathcal{P}|\boldsymbol{r}_{\text{ref}})]_+, \end{aligned} \tag{3}$$

[2] For simplicity, we define the task as minimization in this paper without loss of generality. For maximization, we can negate f, for instance.

where $\mathcal{P}' = \mathcal{P} \cup \{\hat{f}(\boldsymbol{x})\}$ (Emmerich, 2005; Emmerich et al., 2011).

Noisy observations In the noiseless setting, the observed baseline Pareto front is the true baseline Pareto front, i.e. $\mathcal{P}_t = \{\boldsymbol{y} : \boldsymbol{y} \in \mathcal{Y}_t, \nexists\, \boldsymbol{y}' \in \mathcal{Y}_t \textit{ s.t. } \boldsymbol{y}' \prec \boldsymbol{y}\}$, where $\mathcal{Y}_t := \{\boldsymbol{y}^{(n)}\}_{n=1}^{N_t}$. This does not, however, hold in many practical applications, where measurements carry noise. For instance, given a zero-mean Gaussian measurement process with noise covariance Σ, the feedback for a candidate $\boldsymbol{x}$ is $\boldsymbol{y} \sim \mathcal{N}(f(\boldsymbol{x}), \Sigma)$, not $f(\boldsymbol{x})$ itself. The *noisy* expected hypervolume improvement (NEHVI) acquisition function marginalizes over the surrogate posterior at the previously observed points $\mathcal{X}_t = \{\boldsymbol{x}^{(n)}\}_{n=1}^{N_t}$,

$$u_{\text{NEHVI}}(\boldsymbol{x}, \hat{f}, \mathcal{D}) = \text{HVI}(\hat{\mathcal{P}}'_t, \hat{\mathcal{P}}_t | \boldsymbol{r}_{\text{ref}}), \tag{4}$$

where $\hat{\mathcal{P}}_t = \{\hat{f}(\boldsymbol{x}) : \boldsymbol{x} \in \mathcal{X}_t, \nexists\, \boldsymbol{x}' \in \mathcal{X}_t \textit{ s.t. } \hat{f}(\boldsymbol{x}') \prec \hat{f}(\boldsymbol{x})\}$ and $\hat{\mathcal{P}}' = \hat{\mathcal{P}} \cup \{\hat{f}(\boldsymbol{x})\}$ (Daulton et al., 2021).

4. Multi-objective BO with multivariate ranks

In MOBO, it is common to evaluate the quality of an approximate Pareto set $\mathcal{X}$ by computing its distance from the optimal Pareto set $\mathcal{X}^*$ in the objective space, defined by some distance metric $d : 2^{\mathcal{Y}} \times 2^{\mathcal{Y}} \rightarrow \mathbb{R}$ where $2^{\mathcal{Y}}$ denotes the power set of the objective space $\mathcal{Y}$. HVI (Equation 3) is a popular metric, for instance. One advantage of HV is its sensitivity to any type of improvement; whenever an approximation set A dominates another approximation set B, then the measure yields a strictly better quality value for the former (Zitzler et al., 2003). On the other hand, HV suffers from sensitivity to transformations of the objectives and scales super-polynomially with M, which hinders its practical value. An alternative approach is to use distance-based indicators (Miranda & Von Zuben, 2016; Shilton et al., 2018) that assign scores for the solutions based on a signed distance from each point to the approximate Pareto front, which is again computationally expensive.

In the following, the *(weak) Pareto-dominance* relation is used as a preference relation $\preccurlyeq$ on $\mathcal{Y}$ indicating that a solution $\boldsymbol{y}'$ is at least as good as a solution $\boldsymbol{y}$ (denoted $\boldsymbol{y}' \preccurlyeq \boldsymbol{y}$) iff $f_i(\boldsymbol{y}') \leq f_i(\boldsymbol{y})\ \forall i \in [M]$. This relation can be canonically extended to sets of solutions where a set $A \subseteq X$ weakly dominates a set $B \subseteq X$ (denoted $A \preccurlyeq B$) iff $\forall \boldsymbol{y} \in B\ \exists \boldsymbol{y}' \in A : \boldsymbol{y}' \preccurlyeq \boldsymbol{y}$ (Zitzler et al., 2003). Given the preference relation, we consider the optimization goal of identifying a set of solutions that approximates the set of Pareto-optimal solutions and ideally this set is not strictly dominated by any other approximation set.

Since the generalized weak Pareto dominance relation defines only a partial order on $\mathcal{Y}$, there may be incomparable sets in $\mathcal{Y}$. Incomparability is a key challenge in search and performance assessment for multi-objective optimization and becomes more serious as M increases (Fonseca et al., 2005). One way to circumvent this problem is to define a total order on $\mathcal{Y}$ which guarantees that any two objective vector sets are mutually comparable. To this end, quality indicators have been introduced that assign, in the simplest case, each approximation set a real number — that is, a (unary) indicator function $I : \mathcal{Y} \rightarrow \mathbb{R}$ (Zitzler et al., 2003). One important feature an indicator should have is *Pareto compliance* (Fonseca et al., 2005), which dictates that it must not contradict the order induced by the Pareto dominance relation.

In particular, this means that whenever $A \preccurlyeq B \wedge B \not\preccurlyeq A$, then the indicator value of A must not be worse than the indicator value of B. A stricter version of compliance would be to require that the indicator value of A is strictly better than the indicator value of B (if better means a lower indicator value):

$$A \preccurlyeq B \wedge B \not\preccurlyeq A \Rightarrow I(A) < I(B). \tag{5}$$

4.1. CDF indicator

We propose the CDF indicator, a Pareto-compliant indicator for measuring the quality of Pareto approximations.

Definition 4.1 (Cumulative distribution function)**.** The CDF of a real-valued random variable Y is the function:[3]

$$F_Y(y) = P(Y \leq y) = \int_{-\infty}^{y} f_Y(t)dt, \tag{6}$$

representing the probability that Y takes a value less than or equal to y.

For more than two variables, the joint CDF is given by

$$F_{Y_1,\ldots,Y_M}(\boldsymbol{y}) = P(Y_1 \leq y_1, \ldots, Y_M \leq y_m) \tag{7}$$

$$= \int_{(-\infty,\ldots,-\infty)}^{(y_1,\ldots,y_M)} f_Y(\mathbf{s})d\boldsymbol{s}. \tag{8}$$

Properties of the CDF. Every multivariate CDF is monotonically non-decreasing for each Y_i, right-continuous in each Y_i, and takes values in $[0, 1]$. The monotonically non-decreasing property means that $F_Y(a_1, \ldots, a_M) \geq F_Y(b_1, \ldots, b_M)$ whenever $a_1 \geq b_1, \ldots, a_K \geq b_M$. We leverage these properties to define our CDF indicator.

Definition 4.2 (CDF Indicator)**.** The CDF indicator (I_{CDF}) is defined as the minimum multivariate rank:

$$I_{\text{CDF}}(A) := \min_{\boldsymbol{y} \in A} F_Y(\boldsymbol{y}) = \max_{\boldsymbol{y} \in A} [1 - F_Y(\boldsymbol{y})], \tag{9}$$

where A is an approximation set in $\mathcal{Y}$.

[3] In this section, we use the standard notation for densities (f) and distributions (F) defined on the objective space. It will be clear from the context whenever f is again used to refer to the objective function.

Theorem 4.3 (Pareto compliance)**.** *For any pair of approximation sets $A \in \mathcal{Y}$ and $B \in \mathcal{Y}$,*

$$A \preccurlyeq B \wedge B \not\preccurlyeq A \Rightarrow I_{\text{CDF}}(A) \leq I_{\text{CDF}}(B). \tag{10}$$

The proof can be found in Appendix C.

Remark 4.4. Note that I_{F_Y} only depends on the best element in the F_Y rank ordering. One consequence of this is that I_{F_Y} does not discriminate sets with the same best element.

4.1.1. Estimating the CDF with copulas

Estimating F_Y is challenging in high dimensions. Naively estimating the joint multivariate density f_Y and then computing the high-dimensional integral to obtain F_Y would be computationally intensive. To address this, we turn to *copulas* (Nelsen, 2007; Bedford & Cooke, 2002), a statistical tool for flexible density estimation in high dimensions. Vine copulas provide consistent factorization of high-dimensional joints into a product of bivariate densities.

Theorem 4.5. *[Sklar's theorem (Sklar, 1959)] The continuous random vector $Y = (Y_1, \ldots, Y_M)$ has a joint distribution F_Y and marginal distributions $F_1, \ldots, F_M$ iff there exists a unique copula C, which is the joint distribution of $U = (U_1, \ldots, U_M) = F_1(Y_1), \ldots, F_d(Y_M)$.*

A copula is a multivariate distribution function $C : [0,1]^M \to [0,1]$ that joins (couples) uniform marginal distributions $F(y_1, \ldots, y_M) = C\left(F_1(y_1), \ldots, F_M(y_M)\right)$. To be able to estimate a copula, we need to transform the variables of interest to uniform marginals. We do so by the following operation.

Definition 4.6 (Probability integral transform)**.** PIT of a random variable Y with distribution F_Y is the random variable $U = F_Y(Y)$, which is uniformly distributed: $U \sim \text{Unif}([0,1])$.

Theorem 4.5 implies the following corollaries establishing the invariance of the CDF indicator to different scales.

Corollary 4.7 (Scale invariance)**.** *A copula based estimator for the CDF indicator is scale-invariant.*

Corollary 4.8 (Invariance under monotonic transformations)**.** *Let Y_1, Y_2 be continuous random variables with copula C_{Y_1,Y_2}. If $\alpha, \beta : \mathbb{R} \to \mathbb{R}$ are strictly increasing functions, then:*

$$C_{\alpha(Y_1),\beta(Y_2)} = C_{Y_1,Y_2}, \tag{11}$$

where $C_{\alpha(Y_1),\beta(Y_2)}$ is the copula function corresponding to variables $\alpha(Y_1)$ and $\beta(Y_2)$.

Corollary 4.7 follows from the PIT required for copula estimation. The proof for Corollary 4.8, based on Haugh (2016), can be found in subsection C.2 and, without loss of generality, can be extended to $M{>}2$. In Figure 3 we demonstrate the robustness of the copula-based estimator.

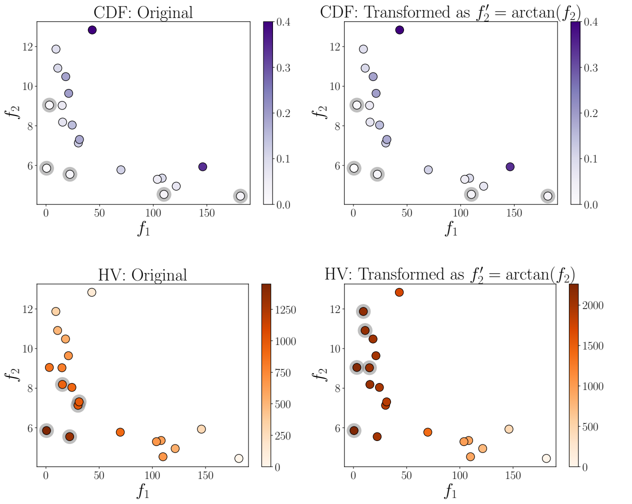


Figure 3. Top: The CDF indicator is invariant to arbitrary monotonic transformations of the objectives (here transforming f_2 via arctan). Bottom: The HV indicator is highly sensitive to them. The color gradient corresponds to indicator value at each solution ($q = 1$). Gray circles are overlaid on the five solutions with the top indicator scores. CDF chooses the same five solutions, but HV prefers ones with high f_1 after f_2 becomes squashed.

The benefits of using copulas to estimate the CDF are threefold: (i) scalability and flexibility with large M, (ii) invariance to relative scales of the different objectives, (iii) invariance to monotonic transformations of the objectives.

From copula density to CDF. It follows from Theorem 4.5 that a joint density of any bivariate random vector (Y_1, Y_2), can be expressed as $f(y_1, y_2) = f_1(y_1) f_2(y_2) c\left(F_1(y_1), F_2(y_2)\right)$ where f_1, f_2 are the marginal densities, F_1, F_2 are the marginal distributions, and c is the copula density. In other words, we can factorize the joint density into a product of the marginals and a copula density. The copula density captures the dependence structure between the two variables after all the complexities in the individual margins are removed. The factorization speeds up the estimation, which breaks down into two simpler steps: estimating the density of the marginal distributions and estimating the copula density. The parameters of the copula and the margins can be estimated with maximum likelihood given a choice of parametric copula families (lower or upper tail dependence, survival copulas, Gaussian, etc.). In addition, recent progress in nonparametric estimation of copulas has enabled the estimation of more complex distributions (Geenens et al., 2017). Once a copula density is fit, the CDF can be obtained analytically in the parametric case or by Monte-Carlo (MC) integration over the density for the nonparametric case. For further details, please refer to Appendix D, Aas et al. (2009), and Joe et al. (2010).

4.1.2. High-dimensional CDF with vine copulas

The above factorization can be generalized to any number of variables. The pair copula constructions called *vines* are hierarchical models, constructed from cascades of bivariate copula blocks, that can accommodate more than two variables (Nagler et al., 2017). Any M-dimensional copula density can be decomposed into a product of $M(M-1)/2$ bivariate (conditional) copula densities (Joe, 1997; Bedford & Cooke, 2002). The factorization is not unique and can be organized in a graphical model, as a sequence of $M-1$ nested trees. We denote a tree as $T_k = (V_k, E_k)$ with V_k and E_k the sets of nodes and edges of tree k for $k = 1, \ldots, M-1$. Each edge e is associated with a bivariate copula. We provide a full example of vine copula decomposition in Appendix D. In practice, in order to construct a vine, one has to choose two components: (1) the structure, or the set of trees $T_k = (V_k, E_k)$ for $k \in [M-1]$ and (2) the pair copulas for $c_{j_e,k_e|D_e}$ where $e \in E_k$ and $k \in [M-1]$. There are efficient algorithms for both steps and we use the implementation by Nagler & Vatter (2018).

4.2. CDF-based acquisition function: BOtied

Suppose we fit a CDF on $\boldsymbol{y}^{(1)}, \boldsymbol{y}^{(2)}, \ldots, \boldsymbol{y}^{(N_t)}$, the N_t measurements acquired thus far. Denote the resulting CDF as $\hat{F}(\cdot; \mathcal{D}_t)$, where we have made explicit the dependence on the dataset up to time t. The utility function of our BOtied acquisition function is as follows:

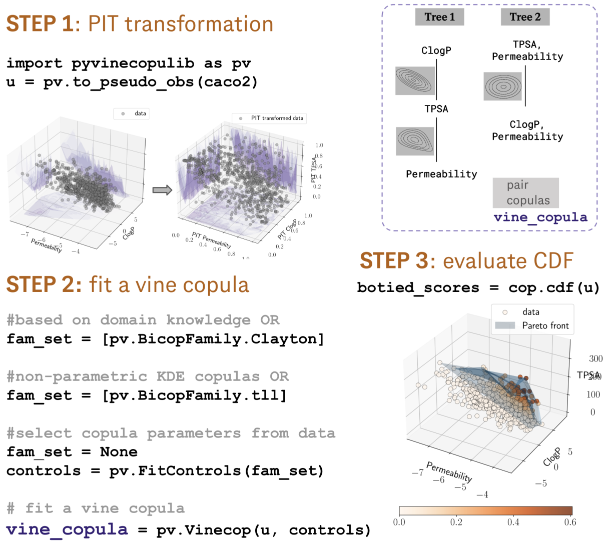
$$u(\mathbf{x}, \hat{f}, \mathcal{D}_t) = 1 - \hat{F}(\hat{f}(\mathbf{x}); \mathcal{D}_t). \tag{12}$$

As with the CDF indicator, our CDF-based acquisition function has an efficient implementation based on vine copulas. For a more precise description of how a CDF-based acquisition function fits within a single round of MOBO, we include Algorithm 1 in Appendix B.

5. Empirical results

Experimental setup. To empirically evaluate the sample efficiency of BOtied, we execute simulated BO rounds on a variety of problems. See Appendix F for more details about our setup. For all the experiments, the surrogate model was an independent GP with a Matern 5/2 ARD kernel. The GP hyperparameters were inferred via maximum a posteriori (MAP) estimation. The code that reproduces all of our experiments and plots is available at `https://github.com/jiwoncpark/botied`.

Metrics. We use the HV indicator presented in section 4, a standard evaluation metric for MOBO, as well as our



STEP 1: PIT transformation

```
import pyvinecopulib as pv
u = pv.to_pseudo_obs(caco2)
```

STEP 2: fit a vine copula

```
#based on domain knowledge OR
fam_set = [pv.BicopFamily.Clayton]

#non-parametric KDE copulas OR
fam_set = [pv.BicopFamily.tll]

#select copula parameters from data
fam_set = None
controls = pv.FitControls(fam_set)

# fit a vine copula
vine_copula = pv.Vinecop(u, controls)
```

STEP 3: evaluate CDF

```
botied_scores = cop.cdf(u)
```

Figure 4. A recipe for estimating the CDF with copulas, in three simple steps and fewer than 5 lines of Python code. Plots are based on the Caco2+ dataset.

CDF indicator I_{CDF} on the noiseless function values. We rely on efficient algorithms for HV computation based on hyper-cell decomposition as described in (Fonseca et al., 2006; Ishibuchi et al., 2011) and implemented in `BoTorch` (Balandat et al., 2020).

BOtied implementation We implement two versions of BOtied that differ in the incorporation of predictive uncertainties in the CDF estimation. In one version (v1), we fit the CDF on all of the MC predictive posterior samples across all the candidates. This can sometimes result in poor CDF fit, particularly when uncertainties are large. The other version (v2) alleviates this issue by fitting the CDF on the posterior means of the candidates. The algorithms for both versions can be found in Algorithm 1, Appendix B. We optimize the BOtied acquisition values using the gradient-free CMA-ES algorithm (Hansen, 2006). The CDF estimation is detailed in Appendix E and BOtied optimization in Appendix F.

Baselines We compare BOtied with the noisy versions of popular acquisition functions. The baseline acquisition strategies are NEHVI (Daulton et al., 2020) described in Equation 4; noisy NParEGO (NParEGO; Knowles, 2006) which uses noisy EI on top of random augmented Chebyshev scalarization; predictive entropy search (PES; Hernández-Lobato et al., 2016a), maximum entropy search (MES; Belakaria et al., 2019), and joint entropy search (JES; Hvarfner et al., 2022) — the differences being the estimation of entropy in the inputs, objectives, or both, respectively; and random (Sobol) selection.

Synthetic datasets. We include synthetic test functions for direct evaluation of f. We focus on ones that support $M \geq 3$: DTLZ2 (d=6, M=4 and d=7, M=6; Deb

Table 1. HV indicators (in the original units) and I_{CDF} across datasets. Higher is better for HV and lower is better for I_{CDF}. The best per column is marked in bold. We report the mean and standard error of each metric across 20 random seeds.

	Penicillin ($d=7, M=3, q=1$)		DTLZ2 ($d=6, M=4, q=1$)		DTLZ2 ($d=7, M=6, q=1$)	
	$I_{\text{CDF}} \downarrow$	HV $\uparrow$	$I_{\text{CDF}} \downarrow$	HV $\uparrow$	$I_{\text{CDF}} \downarrow$	HV $\uparrow$
BOtied v1	0.26 (0.01)	325741 (29515)	0.25 (0.01)	2.26 (0.09)	0.071 (0.005)	0.36 (0.03)
BOtied v2	**0.20 (0.02)**	**342762 (13599)**	0.11 (0.02)	**2.32 (0.06)**	0.064 (0.004)	**0.42 (0.02)**
NParEGO	0.28 (0.01)	303707 (15118)	**0.10 (0.02)**	2.20 (0.11)	0.065 (0.005)	0.38 (0.02)
NEHVI	0.28 (0.01)	314294 (14498)	0.24 (0.01)	1.80 (0.06)	0.074 (0.007)	0.27 (0.01)
PES	0.27 (0.01)	297107 (17383)	0.24 (0.01)	1.85 (0.11)	0.069 (0.004)	0.23 (0.06)
MES	0.24 (0.02)	305874 (14694)	**0.10 (0.02)**	2.12 (0.08)	**0.059 (0.004)**	0.27 (0.06)
JES	0.28 (0.01)	316302 (21193)	0.24 (0.01)	1.97 (0.07)	0.069 (0.006)	0.25 (0.05)
Random	0.24 (0.02)	307896 (22889)	0.11 (0.02)	0.91 (0.08)	0.076 (0.005)	0.10 (0.01)

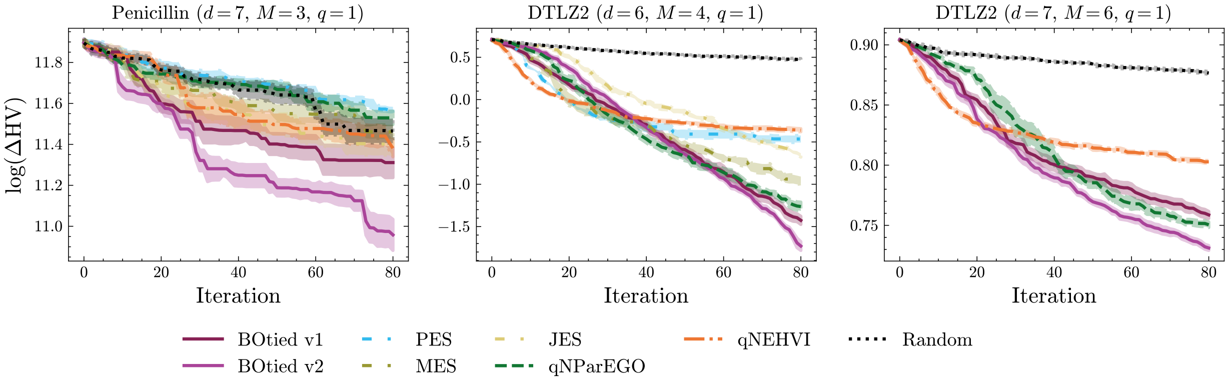


Figure 5. HV vs. iterations for three synthetic test functions. We show the mean and two standard errors over 20 random seeds.

& Gupta, 2005) and Penicillin (d=7, M=3; Liang & Lai, 2021), which simulates the penicillin yield, time to production, and undesired byproduct for various parameters of the production process. See Appendix F for more detail.

Real-world datasets. To emulate a multi-objective drug design setting, we postprocess the real-world dataset Caco2 (Wang et al., 2016) from the Therapeutics Data Commons database (Huang et al., 2021; 2022) to create Caco2+. The original Caco2 dataset consists of 906 drug molecules annotated with experimentally measured cell permeability, or the rates of passing through a human colon epithelial cancer cell line. Permeability is a key property in the absorption, distribution, metabolism, and excretion (ADME) profile of drugs. We augment the dataset with additional properties using RDKit (Landrum et al., 2023), including ClogP related to fat solubility and topological polar surface area (TPSA). Subsets of these properties (e.g., permeability and TPSA) are inversely correlated and thus compete with one another during optimization. In late-stage lead-molecule optimization, the trade-offs become more dramatic and as more properties are added (Sun et al., 2022). Demonstrating effective sampling of Pareto-optimal solutions in this setting is thus of great value. We represent each molecule as a concatenation of fingerprint and fragment feature vectors (Thawani et al., 2020).

We also include experiments over three datasets from the DDMOP benchmark (He et al., 2020). Differently from the synthetic test functions which have analytical solutions, each DDMOP dataset represents a complex objective function approximated by expensive numerical simulations. These datasets address cab car optimization, power system chip placement and neural network. Details and table results on each dataset can be found in subsection F.5. See Appendix F for more detail about these datasets.

Vine Copulas for MOBO in practice In Figure 4, we present a simple recipe for estimating CDFs with vine copulas, in three simple steps and fewer than five lines of code. We use the Caco2+ dataset (M=3) as an example. First, the PIT transformation yields the uniform margins. We then choose a copula shape from parametric or non-parametric families, or we leave this undetermined and run model selection based on the Bayesian information criterion (BIC). In the case of Caco2+, we can use the domain knowledge that permeability and TPSA are negatively correlated and specify a Clayton copula. Once a vine copula has been fit, it is fully described by the trees (structure) and bivariate (pair) copula densities associated with each edge. This is all we need to evaluate the CDF on the data points in Step 3. Note that the darker shaded points corresponding to higher CDF scores indeed approach the Pareto front.

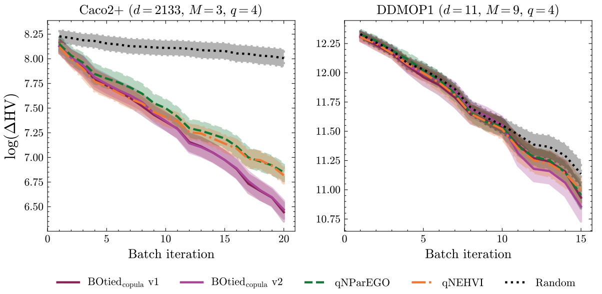


Figure 6. HV vs. iterations for real-world datasets. We show the mean and two standard errors over 20 random seeds.

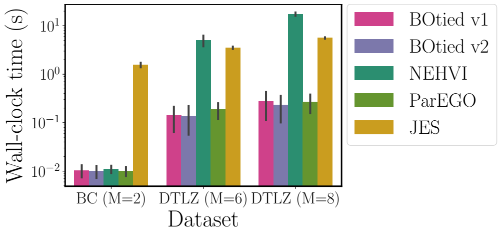


Figure 7. Wall-clock time per single call of acquisition function. Error bars are standard deviations across five repeated calls.

5.1. Results and discussion

We compare the performance of BOtied with baseline acquisition strategies in terms of both the HV and the CDF indicators, on synthetic test functions (Figure 5) as well as on real-world datasets (Figure 6). The metrics for these experiments and additional experiments using various q batch sizes are tabulated in Table 1. Although there is no single best method across all the datasets, the best numbers are consistently achieved by either BOtied v1 or v2 with NParEGO being a close competitor. The NEHVI performance visibly degrades as M increases.

Figure 7 shows that the wall-clock time for NEHVI and JES become very slow for $M \geq 3$. At the same time, BOtied is significantly faster than NEHVI/JES and is as fast as NParEGO, which is based on scalarizing the M objectives.

There are two main benefits to using I_{CDF} rather than HV for evaluation. First, the CDF is bounded between 0 and 1, with scores close to 0 corresponding to the solutions closest to our approximate Pareto front. Unlike HV values, for which the scales do not carry information about the internal ordering, the I_{CDF} values have an interpretable scale. Second, assuming the GP and copula have been properly fit, we can use the magnitude of I_{CDF} to determine the orthogonality, or degree of competition, of the objectives in a given task. In particular, when a candidate strongly dominates a set of points, its I_{CDF} tends below 0.1, while for points that weakly dominate with respect to a small subset of the objectives, the I_{CDF} value is higher.

We stress-test BOtied in a series of ablation studies in Appendix G. In particular, we vary the number of MC posterior predictive samples and find that BOtied v1 is robust to the number of posterior samples, i.e., the multivariate ranks associated with the best-fit copula model do not change significantly with varying numbers of samples. When the posterior shape is complex such that many MC samples are required to fully characterize the posterior, BOtied v2 (in which the copula is fit on the mean of the posterior samples) is more appropriate than v1.

Limitations When fitting the CDF model, there's a tradeoff between flexibility and complexity. Increasing M requires us to adopt more flexible models, which increases the number of modeling choices. In our experiments, we perform model selection based on the Akaike information criterion (AIC) to choose among nonparametric and parametric copula families (Akaike, 1998). Moreover, the current implementation of BOtied is not differentiable, which necessitates the use of gradient-free algorithms such as CMA-ES for optimizing acquisition values where a gradient-based one may be more efficient.

6. Conclusion

We introduce a new perspective on MOBO based on the multivariate CDF. Our proposed MOBO acquisition function, BOtied, is computed by fitting a multivariate CDF on the surrogate predictions and extracting the ranks associated with the CDF scores. It is computationally attractive, as the CDF can be efficiently fit with vine copulas even when M is large. Moreover, it enables model-based estimation of the Pareto front. When domain knowledge about the distribution of the objective values is available, it can be injected into the specification of the CDF model family. We also propose a new Pareto-compliant indicator for measuring the quality of approximate Pareto fronts, the CDF indicator. The CDF indicator, equipped with desirable properties such as invariance to monotonic transformations of the objectives, promises to complement the popular HV indicator.

Our method is general and lends itself to a number of immediate extensions. First, whereas we have implemented gradient-free optimization of BOtied in this work, we can take advantage of gradient-based optimization for improved efficiency. In conjunction with a differentiable sorting algorithm (e.g., Cuturi et al., 2019; Blondel et al., 2020), the computation of our acquisition function can be made differentiable for many parametric copula families. Second, we can consider constrained or discrete extensions for broader applicability. Finally, as many applications carry noise in the input as well as the function of interest, accounting for input noise through the established connection between copulas and multivariate value-at-risk (MVaR) estimation will be of great practical interest.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Aas, K., Czado, C., Frigessi, A., and Bakken, H. Pair-copula constructions of multiple dependence. *Insurance: Mathematics and economics*, 44(2):182–198, 2009.

Akaike, H. Information theory and an extension of the maximum likelihood principle. In *Selected papers of hirotugu akaike*, pp. 199–213. Springer, 1998.

Balandat, M., Karrer, B., Jiang, D. R., Daulton, S., Letham, B., Wilson, A. G., and Bakshy, E. BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization. In *NeurIPS*, 2020.

Bautista, D. C. *A sequential design for approximating the pareto front using the expected pareto improvement function*. The Ohio State University, 2009.

Bedford, T. and Cooke, R. M. Vines – A New Graphical Model for Dependent Random Variables. *The Annals of Statistics*, 30(4):1031–1068, 2002.

Belakaria, S., Deshwal, A., and Doppa, J. R. Max-value entropy search for multi-objective bayesian optimization. *NeurIPS*, 32, 2019.

Bellamy, H., Rehim, A. A., Orhobor, O. I., and King, R. Batched bayesian optimization for drug design in noisy environments. *Journal of Chemical Information and Modeling*, 62(17):3970–3981, 2022.

Binois, M., Rullière, D., and Roustant, O. On the estimation of pareto fronts from the point of view of copula theory. *Information Sciences*, 324:270–285, 2015.

Binois, M., Picheny, V., Taillandier, P., and Habbal, A. The kalai-smorodinsky solution for many-objective bayesian optimization. *The Journal of Machine Learning Research*, 21(1):5978–6019, 2020.

Blondel, M., Teboul, O., Berthet, Q., and Djolonga, J. Fast differentiable sorting and ranking. In *International Conference on Machine Learning*, pp. 950–959. PMLR, 2020.

Calandra, R., Seyfarth, A., Peters, J., and Deisenroth, M. P. Bayesian optimization for learning gaits under uncertainty: An experimental comparison on a dynamic bipedal walker. *Annals of Mathematics and Artificial Intelligence*, 76:5–23, 2016.

Cuturi, M., Teboul, O., and Vert, J.-P. Differentiable ranking and sorting using optimal transport. *Advances in neural information processing systems*, 32, 2019.

D. Segall, M. Multi-parameter optimization: identifying high quality compounds with a balance of properties. *Current pharmaceutical design*, 18(9):1292, 2012.

Dächert, K., Klamroth, K., Lacour, R., and Vanderpooten, D. Efficient computation of the search region in multi-objective optimization. *European Journal of Operational Research*, 260(3):841–855, 2017.

Daulton, S., Balandat, M., and Bakshy, E. Differentiable expected hypervolume improvement for parallel multi-objective bayesian optimization. *NeurIPS*, 33:9851–9864, 2020.

Daulton, S., Balandat, M., and Bakshy, E. Parallel bayesian optimization of multiple noisy objectives with expected hypervolume improvement. *NeurIPS*, 34:2187–2200, 2021.

Daulton, S., Wan, X., Eriksson, D., Balandat, M., Osborne, M. A., and Bakshy, E. Bayesian optimization over discrete and mixed spaces via probabilistic reparameterization. In *ICML 2022 Workshop on Adaptive Experimental Design and Active Learning in the Real World*, 2022.

Deb, K. and Gupta, H. Searching for robust pareto-optimal solutions in multi-objective optimization. In *Evolutionary Multi-Criterion Optimization: Third International Conference, EMO 2005, Guanajuato, Mexico, March 9-11, 2005. Proceedings 3*, pp. 150–164. Springer, 2005.

Deb, K., Gupta, S., Daum, D., Branke, J., Mall, A. K., and Padmanabhan, D. Reliability-based optimization using evolutionary algorithms. *IEEE transactions on evolutionary computation*, 13(5):1054–1074, 2009.

Deutz, A., Emmerich, M., and Yang, K. The expected r2-indicator improvement for multi-objective bayesian optimization. In *Evolutionary Multi-Criterion Optimization: 10th International Conference, EMO 2019, East Lansing, MI, USA, March 10-13, 2019, Proceedings 10*, pp. 359–370. Springer, 2019a.

Deutz, A., Yang, K., and Emmerich, M. The r2 indicator: A study of its expected improvement in case of two objectives. In *AIP Conference Proceedings*, volume 2070, pp. 020054. AIP Publishing LLC, 2019b.

Embrechts, P., Höing, A., and Juri, A. Using copulae to bound the value-at-risk for functions of dependent risks. *Finance and Stochastics*, 7(2):145–167, 2003.

Emmerich, M. *Single-and multi-objective evolutionary design optimization assisted by gaussian random field metamodels*. PhD thesis, Dortmund, Univ., Diss., 2005, 2005.

Emmerich, M. T., Deutz, A. H., and Klinkenberg, J. W. Hypervolume-based expected improvement: Monotonicity properties and exact computation. In *2011 IEEE Congress of Evolutionary Computation (CEC)*, pp. 2147–2154. IEEE, 2011.

Eriksson, D. and Poloczek, M. Scalable constrained bayesian optimization. In *International Conference on Artificial Intelligence and Statistics*, pp. 730–738. PMLR, 2021.

Fonseca, C. M., Knowles, J. D., Thiele, L., Zitzler, E., et al. A tutorial on the performance assessment of stochastic multiobjective optimizers. In *Third international conference on evolutionary multi-criterion optimization (EMO 2005)*, volume 216, pp. 240, 2005.

Fonseca, C. M., Paquete, L., and López-Ibánez, M. An improved dimension-sweep algorithm for the hypervolume indicator. In *2006 IEEE international conference on evolutionary computation*, pp. 1157–1163. IEEE, 2006.

Frazier, P. I. A tutorial on bayesian optimization. *arXiv preprint arXiv:1807.02811*, 2018.

Gardner, J. R., Kusner, M. J., Xu, Z. E., Weinberger, K. Q., and Cunningham, J. P. Bayesian optimization with inequality constraints. In *ICML*, volume 2014, pp. 937–945, 2014.

Geenens, G., Charpentier, A., and Paindaveine, D. Probit transformation for nonparametric kernel estimation of the copula density. *Bernoulli*, 23(3):1848–1873, 2017.

Ghosal, P. and Sen, B. Multivariate ranks and quantiles using optimal transport: Consistency, rates and nonparametric testing. *The Annals of Statistics*, 50(2):1012–1037, 2022.

Griffiths, R.-R., Klarner, L., Moss, H. B., Ravuri, A., Truong, S., Rankovic, B., Du, Y., Jamasb, A., Schwartz, J., Tripp, A., Kell, G., Bourached, A., Chan, A., Moss, J., Guo, C., Lee, A. A., Schwaller, P., and Tang, J. Gauche: A library for gaussian processes in chemistry, 2022.

Hansen, N. The cma evolution strategy: a comparing review. *Towards a new evolutionary computation: Advances in the estimation of distribution algorithms*, pp. 75–102, 2006.

Haugh, M. An introduction to copulas. quantitative risk management, 2016.

He, C., Tian, Y., Wang, H., and Jin, Y. A repository of real-world datasets for data-driven evolutionary multiobjective optimization. *Complex & Intelligent Systems*, 6(1):189–197, 2020. doi: 10.1007/s40747-019-00126-2.

Hennig, P. and Schuler, C. J. Entropy search for information-efficient global optimization. *Journal of Machine Learning Research*, 13(6), 2012.

Hernández-Lobato, D., Hernandez-Lobato, J., Shah, A., and Adams, R. Predictive entropy search for multi-objective bayesian optimization. In *ICML*, pp. 1492–1501. PMLR, 2016a.

Hernández-Lobato, J. M., Gelbart, M. A., Adams, R. P., Hoffman, M. W., and Ghahramani, Z. A general framework for constrained bayesian optimization using information-based search. 2016b.

Hoffman, M. W. and Ghahramani, Z. Output-space predictive entropy search for flexible global optimization. In *NeurIPS workshop on Bayesian Optimization*, pp. 1–5, 2015.

Huang, K., Fu, T., Gao, W., Zhao, Y., Roohani, Y., Leskovec, J., Coley, C. W., Xiao, C., Sun, J., and Zitnik, M. Therapeutics data commons: Machine learning datasets and tasks for drug discovery and development. *NeurIPS Datasets and Benchmarks*, 2021.

Huang, K., Fu, T., Gao, W., Zhao, Y., Roohani, Y., Leskovec, J., Coley, C. W., Xiao, C., Sun, J., and Zitnik, M. Artificial intelligence foundation for therapeutic science. *Nature Chemical Biology*, 2022.

Hvarfner, C., Hutter, F., and Nardi, L. Joint entropy search for maximally-informed bayesian optimization. *Advances in Neural Information Processing Systems*, 35:11494–11506, 2022.

Ishibuchi, H., Akedo, N., and Nojima, Y. A many-objective test problem for visually examining diversity maintenance behavior in a decision space. In *Proceedings of the 13th annual conference on Genetic and evolutionary computation*, pp. 649–656, 2011.

Jain, T., Sun, T., Durand, S., Hall, A., Houston, N. R., Nett, J. H., Sharkey, B., Bobrowicz, B., Caffry, I., Yu, Y., et al. Biophysical properties of the clinical-stage antibody landscape. *Proceedings of the National Academy of Sciences*, 114(5):944–949, 2017.

Jin, Y. and Sendhoff, B. Pareto-based multiobjective machine learning: An overview and case studies. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38(3):397–415, 2008.

Joe, H. *Multivariate Models and Dependence Concepts*. Chapman & Hall/CRC, 1997.

Joe, H., Li, H., and Nikoloulopoulos, A. K. Tail dependence functions and vine copulas. *Journal of Multivariate Analysis*, 101(1):252–270, January 2010.

Jones, D. R., Schonlau, M., and Welch, W. J. Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 13(4):455–492, 1998.

Kavasseri, R. and Srinivasan, S. K. Joint placement of phasor and power flow measurements for observability of power systems. *IEEE Transactions on Power Systems*, 26(4):1929–1936, 2011.

Knowles, J. Parego: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation*, 10(1):50–66, 2006.

Kukkonen, S. and Lampinen, J. Ranking-dominance and many-objective optimization. In *2007 IEEE Congress on Evolutionary Computation*, pp. 3983–3990. IEEE, 2007.

Kusne, A. G., Yu, H., Wu, C., Zhang, H., Hattrick-Simpers, J., DeCost, B., Sarker, S., Oses, C., Toher, C., Curtarolo, S., et al. On-the-fly closed-loop materials discovery via bayesian active learning. *Nature communications*, 11(1): 5966, 2020.

Landrum, G., Tosco, P., Kelley, B., Ric, sriniker, Cosgrove, D., gedeck, Vianello, R., NadineSchneider, Kawashima, E., N, D., Jones, G., Dalke, A., Cole, B., Swain, M., Turk, S., AlexanderSavelyev, Vaucher, A., Wójcikowski, M., Take, I., Probst, D., Ujihara, K., Scalfani, V. F., guillaume godin, Pahl, A., Berenger, F., JLVarjo, Walker, R., jasondbiggs, and strets123. rdkit/rdkit: 2023_03_1 (q1 2023) release. April 2023. doi: 10.5281/zenodo.7880616. URL https://doi.org/10.5281/zenodo.7880616.

Liang, Q. and Lai, L. Scalable bayesian optimization accelerates process optimization of penicillin production. In *NeurIPS 2021 AI for Science Workshop*, 2021.

Marler, R. T. and Arora, J. S. Survey of multi-objective optimization methods for engineering. *Structural and multidisciplinary optimization*, 26:369–395, 2004.

Miranda, C. S. and Von Zuben, F. J. Necessary and sufficient conditions for surrogate functions of pareto frontiers and their synthesis using gaussian processes. *IEEE Transactions on Evolutionary Computation*, 21(1):1–13, 2016.

Močkus, J. On bayesian methods for seeking the extremum. In *Optimization techniques IFIP technical conference*, pp. 400–404. Springer, 1975.

Nagler, T. and Czado, C. Evading the curse of dimensionality in nonparametric density estimation with simplified vine copulas. *Journal of Multivariate Analysis*, 151:69–89, 2016.

Nagler, T. and Vatter, T. *kde1d: Univariate Kernel Density Estimation*, 2018. R package version 0.2.1.

Nagler, T., Schellhase, C., and Czado, C. Nonparametric estimation of simplified vine copula models: comparison of methods. *Dependence Modeling*, 5(1):99–120, 2017.

Nelsen, R. B. *An introduction to copulas*. Springer Science & Business Media, 2007.

Paria, B., Kandasamy, K., and Póczos, B. A flexible framework for multi-objective bayesian optimization using random scalarizations. In *UAI*, pp. 766–776. PMLR, 2020.

Park, J. W., Stanton, S., Saremi, S., Watkins, A., Dwyer, H., Gligorijevic, V., Bonneau, R., Ra, S., and Cho, K. Propertydag: Multi-objective bayesian optimization of partially ordered, mixed-variable properties for biological sequence design. *NeurIPS AI for Science workshop*, 2022.

Picheny, V., Vakili, S., and Artemev, A. Ordinal bayesian optimisation. *arXiv preprint arXiv:1912.02493*, 2019.

Romero, P. A., Krause, A., and Arnold, F. H. Navigating the protein fitness landscape with gaussian processes. *Proceedings of the National Academy of Sciences*, 110(3): E193–E201, 2013.

Shah, A. and Ghahramani, Z. Parallel predictive entropy search for batch global optimization of expensive objective functions. *NeurIPS*, 28, 2015.

Shahriari, B., Swersky, K., Wang, Z., Adams, R. P., and De Freitas, N. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104 (1):148–175, 2015.

Shields, B. J., Stevens, J., Li, J., Parasram, M., Damani, F., Alvarado, J. I. M., Janey, J. M., Adams, R. P., and Doyle, A. G. Bayesian reaction optimization as a tool for chemical synthesis. *Nature*, 590(7844):89–96, 2021.

Shilton, A., Rana, S., Gupta, S., and Venkatesh, S. Multi-target optimisation via bayesian optimisation and linear programming. In *UAI*, pp. 145–155, 2018.

Sklar, A. Fonctions de Répartition à n Dimensions et Leurs Marges. *Publications de L'Institut de Statistique de L'Université de Paris*, 8:229–231, 1959.

Sun, D., Gao, W., Hu, H., and Zhou, S. Why 90% of clinical drug development fails and how to improve it? *Acta Pharmaceutica Sinica B*, 2022.

Svenson, J. *Computer experiments: Multiobjective optimization and sensitivity analysis*. PhD thesis, The Ohio State University, 2011.

Tagasovska, N., Frey, N. C., Loukas, A., Hötzel, I., Lafrance-Vanasse, J., Kelly, R. L., Wu, Y., Rajpal, A., Bonneau, R., Cho, K., et al. A pareto-optimal compositional energy-based model for sampling and optimization of protein sequences. *NeurIPS AI for Science workshop*, 2022.

Tagasovska, N., Ozdemir, F., and Brando, A. Retrospective uncertainties for deep models using vine copulas. In *International Conference on Artificial Intelligence and Statistics*, pp. 7528–7539. PMLR, 2023.

Thawani, A. R., Griffiths, R.-R., Jamasb, A., Bourached, A., Jones, P., McCorkindale, W., Aldrick, A. A., and Lee, A. A. The photoswitch dataset: A molecular machine learning benchmark for the advancement of synthetic chemistry. *arXiv preprint arXiv:2008.03226*, 2020.

Tu, B., Gandy, A., Kantas, N., and Shafei, B. Joint entropy search for multi-objective bayesian optimization. *arXiv preprint arXiv:2210.02905*, 2022.

Van Breemen, R. B. and Li, Y. Caco-2 cell permeability assays to measure drug absorption. *Expert opinion on drug metabolism & toxicology*, 1(2):175–185, 2005.

Villemonteix, J., Vazquez, E., and Walter, E. An informational approach to the global optimization of expensive-to-evaluate functions. *Journal of Global Optimization*, 44:509–534, 2009.

Wang, N.-N., Dong, J., Deng, Y.-H., Zhu, M.-F., Wen, M., Yao, Z.-J., Lu, A.-P., Wang, J.-B., and Cao, D.-S. Adme properties evaluation in drug discovery: prediction of caco-2 cell permeability using a combination of nsga-ii and boosting. *Journal of Chemical Information and Modeling*, 56(4):763–773, 2016.

Wilson, J., Hutter, F., and Deisenroth, M. Maximizing acquisition functions for bayesian optimization. *NeurIPS*, 31, 2018.

Yang, K., Emmerich, M., Deutz, A., and Bäck, T. Efficient computation of expected hypervolume improvement using box decomposition algorithms. *Journal of Global Optimization*, 75:3–34, 2019.

Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. M., and Da Fonseca, V. G. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Transactions on evolutionary computation*, 7(2):117–132, 2003.

Zuo, Y., Qin, M., Chen, C., Ye, W., Li, X., Luo, J., and Ong, S. P. Accelerating materials discovery with bayesian optimization and graph deep learning. *Materials Today*, 51:126–135, 2021.

A. Related work in multi-objective optimization

Table 2. Comparison of BOtied with related work

Type of groundwork	Scoring method	MO criteria	Scalability with M	Scale invariance	Bayesian optimization	Non GP surrogates
Multivariate ranks/CDF, Copula, Copula space	BOtied (this work)	✓	✓	✓	✓	✓
Copula space, Game theory	Kalai-Smorodinsky MO (Binois et al., 2020)	✓	✓	✓	✓	✗
Multivariate ranks	Aggregate Rank (Kukkonen & Lampinen, 2007)	✓	✓	✓	✗	✗
	Ordinal BO (Picheny et al., 2019)	✓	✗	✗	✓	✓
Information Theoretic	Joint Entropy Search (Tu et al., 2022; Hvarfner et al., 2022)	✓	✗	✓	✓	✗
	Predictive Entropy Search (Hernández-Lobato et al., 2016a)	✓	✗	✗	✓	✗
	Max-Value Entropy Search (Belakaria et al., 2019)	✓	✗	✗	✓	✗
Hypervolume	EHVI variants (Daulton et al., 2021; 2022)	✓	✗	✗	✓	✓
Random scalarization	ParEGO (Knowles, 2006)	✓	✓	✗	✓	✓
Boundary distance	SVM-variants (Miranda & Von Zuben, 2016; Shilton et al., 2018)	✓	✓	✗	✓	✗
Maxmin, Pareto Indicator	Pareto improvement , EmaX (Bautista, 2009) Maximin improvement (Svenson, 2011)	✓	✓	✗	✓	✓
Completeness	Averaged completeness indicator (Svenson, 2011)	✓	✗	✓	✓	✗
	Estimated completeness indicator improvement (Svenson, 2011)	✓	✓	✓	✓	✓

B. Algorithm

Algorithm 1 MOBO with BOtied: a CDF-based acquisition function

```
Input: Surrogate model $\hat{f}$, initial data $\mathcal{D}_0 = \{(\boldsymbol{x}_n, \boldsymbol{y}_n)\}_{n=1}^{N_0}$, $\mathcal{X} \subset \mathbb{R}^d, \mathcal{Y} \subset \mathbb{R}^M$, number of MOBO iterations $T$, size of the candidate pool used in each inner-loop optimization of the acquisition function $N$, number of posterior predictive sample $L$
Output: Optimal selected subset $\mathcal{D}_T$.
for $\{t = 1, \ldots, T\}$ do
    while converged do
        Sample the candidate pool $X \coloneqq [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N] \subset \mathcal{X}$
        Obtain the predictive distribution $p(f|\mathcal{D}_{t-1}, X)$
        Draw $L$ predictive samples $\hat{f}^{(j)} \sim p(f|\mathcal{D}_{t-1}, X)$, for $j \in [L]$
        Version 1: Fit a CDF $\hat{F}$ on the pooled samples, $\{\hat{f}^{(j)}\}_{j\in[L]}$.
        Version 2: Fit a CDF $\hat{F}$ on the mean-aggregated samples, $\frac{1}{L}\sum_{j=1}^{L} \hat{f}^{(j)}$ (or posterior mean parameters if they are directly available from the parameterization of the $\hat{f}$ posterior).
        for $\{i = 1, \ldots, N\}$ do
            Version 1: Evaluate the fit CDF $\hat{F}$ on the samples and take the mean across the samples $\mathcal{S}(\boldsymbol{x}_i) = \frac{1}{L}\sum_{j=1}^{L} \hat{C}\left(\hat{f}_i^{(j)}\right)$
            Version 2: Evaluate the fit CDF $\hat{F}$ on the posterior means $\mathcal{S}(\boldsymbol{x}_i) = \hat{C}\left(\frac{1}{L}\sum_{j=1}^{L} \hat{f}_i^{(j)}\right)$
        end for
    end while
    $i^\star \leftarrow \arg\max_{i\in[N]} \mathcal{S}(\boldsymbol{x}_i)$
    $\mathcal{D}_t \leftarrow \mathcal{D}_{t-1} \cup \{(\boldsymbol{x}_{i^\star}, \boldsymbol{y}_{i^\star})\}$
end for
return $\mathcal{D}_T$
```

C. Properties of the CDF indicator

C.1. Theorem 1: Pareto compliance of the CDF indicator

We state Theorem 4.3 again and provide the proof here.

Theorem 4.3: For any arbitrary approximation sets $A \in \mathcal{X}$ and $B \in \mathcal{X}$ where $\mathcal{X} \subset \mathbb{R}^d$, the following holds:

$$A \preccurlyeq B \land B \not\preccurlyeq A \Rightarrow I_F(A) \leq I_F(B).$$

Proof. If we have $A \preccurlyeq B \land B \not\preccurlyeq A$, then the following two conditions hold: $\forall \boldsymbol{x}' \in B\ \exists \boldsymbol{x} \in A : \boldsymbol{x} \preccurlyeq \boldsymbol{x}'$ and $\exists \mathbf{x} \in A\ s.t.\ \nexists \boldsymbol{x}' \in B : \boldsymbol{x}' \preccurlyeq \mathbf{x}$. Recall that the weak Pareto dominance $\mathbf{x} \preccurlyeq \mathbf{x}'$ implies that $\forall i \in [M] : f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{x}')$. From the definition and fundamental property of the CDF being a monotonic non-decreasing function, it follows that $\forall i \in [M] : f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{x}') \Rightarrow F_Y(\boldsymbol{x}) \leq F_Y(\boldsymbol{x}')$.

Define the set of non-dominated solutions in B, $\mathcal{P}_B \coloneqq \{\boldsymbol{x} \in B, \forall \boldsymbol{x}' \in B : \boldsymbol{x} \preceq \boldsymbol{x}'\}$. Note that $I_F(B) = I_F(\mathcal{P}_B) = I_F(\{\boldsymbol{z}\})$ for any $\boldsymbol{z} \in \mathcal{P}_B$. Now let $\boldsymbol{x}_B \in \mathcal{P}_B$. There is $\boldsymbol{x}_A \in A$ such that $\boldsymbol{x}_A \preceq \boldsymbol{x}_B$, and we have that $F_Y(\boldsymbol{x}_A) \leq F_Y(\boldsymbol{x}_B)$. By definition, $I_F(A) \leq I_F(\{\boldsymbol{x}_A\})$ so we have $I_F(A) \leq I_F(\{\boldsymbol{x}_A\}) \leq I_F(\{\boldsymbol{x}_B\}) = I_F(B)$ as desired. □

C.2. Corollary 2: Invariance under monotonic transformations

This proof closely follows the one in (Haugh, 2016).

Corollary 2: Let Y_1, Y_2 be continuous random variables with copula C_{Y_1,Y_2}. If $\alpha, \beta : \mathbb{R} \rightarrow \mathbb{R}$ are strictly increasing functions, then:

$$C_{\alpha(Y_1),\beta(Y_2)} = C_{Y_1,Y_2} \tag{13}$$

where $C_{\alpha(Y_1),\beta(Y_2)}$ is the copula function corresponding to variables $\alpha(Y_1)$ and $\beta(Y_2)$.

Proof. We first note that for the distribution function of $\alpha(Y_1)$ it holds that

$$F_{\alpha(Y_1)} = P(\alpha(Y_1) \leq y_1) = P(Y_1 \leq \alpha^{-1}(y_1)) = F_{Y_1}(\alpha^{-1}(y_1)) \tag{14}$$

and analogously,

$$F_\beta(Y_1)(y_1) = F_{Y_1}(\beta^{-1}(y_1)) \tag{15}$$

From Sklar's theorem, we have that for all $y_1, y_2 \in \mathbb{R}$

$$\begin{aligned}
C_{\alpha(Y_1)\beta(Y_2)}(F_{\alpha(Y_1)}(y_1), F_{\beta(Y_2)}(y_2)) &= F_{\alpha(Y_1)\beta(Y_2)}(y_1, y_2) \\
&= P(\alpha(Y_1) \leq y_1, \beta(Y_2) \leq y_2) \\
&= P(Y_1 \leq \alpha^{-1}(y_1), Y_2 \leq \beta^{-1}(y_2)) \\
&= F_{Y_1,Y_2}(\alpha^{-1}(y_1), \beta^{-1}(y_2))) \\
&= C_{Y_1,Y_2}(F_{Y_1}(\alpha^{-1}(y_1)), F_{Y_2}(\beta^{-1}(y_2))) \\
&= C_{Y_1,Y_2}(F_{\alpha(Y_1)}(y_1), F_{\beta(Y_2)}(y_2))
\end{aligned}$$

Equalities one and five follow from Sklar's theorem. In the third equality we make use of fact that α and β are increasing functions. The last equality follows from Equation C.2 and Equation C.2. □

D. (Vine) copula overview and example

According to Sklar's theorem (Sklar, 1959), the joint density of any bivariate random vector (X_1, X_2), can be expressed as

$$f(x_1, x_2) = f_1(x_1) f_2(x_2) c\left(F_1(x_1), F_2(x_2)\right) \tag{16}$$

where f_i[4] are the marginal densities, F_i the marginal distributions, and c the copula density.

That is, any bivariate density is uniquely described by the product of its marginal densities and a *copula density*, which is interpreted as the *dependence structure*. For self-containment of the manuscript, we borrow an example from (Tagasovska et al., 2023). Figure D.7 illustrates all of the components representing the joint density.

As a benefit of such factorization, by taking the logarithm on both sides, one can estimate the joint density in two steps, first for the marginal distributions, and then for the copula. Hence, copulas provide a means to flexibly specify the marginal and joint distribution of variables. For further details, please refer to (Aas et al., 2009; Joe et al., 2010).

There exist many parametric representations through different copula families, however, to leverage even more flexibility, in this paper, we focus on the kernel-based nonparametric copulas of (Geenens et al., 2017).

Equation 16 can be generalized and holds for any number of variables. To be able to fit densities of more than two variables, we make use of the pair copula constructions, namely *vines*; hierarchical models, constructed from cascades of bivariate copula blocks (Nagler et al., 2017).

According to (Joe, 1997; Bedford & Cooke, 2002), any M-dimensional copula density can be decomposed into a product of $\frac{M(M-1)}{2}$ bivariate (conditional) copula densities. Although such factorization may not be unique, it can be organized in a graphical model, as a sequence of $M-1$ nested trees, called *vines*. We denote a tree as $T_m = (V_m, E_m)$ with V_m and E_m the sets of nodes and edges of tree m for $m = 1, \ldots, M-1$. Each edge e is associated with a bivariate copula. An example of a vine copula decomposition is given in Figure D.

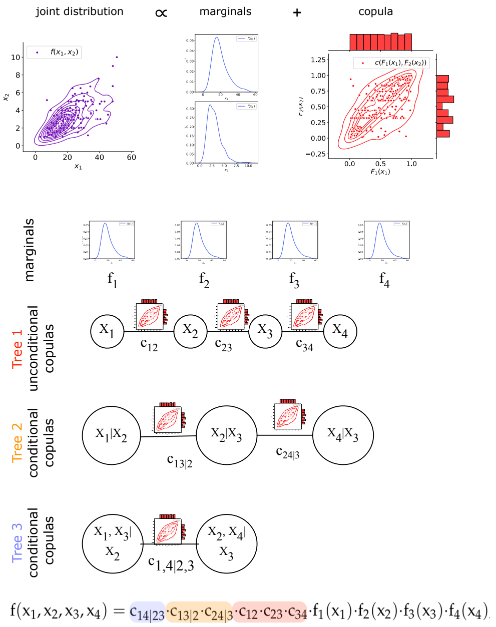


Figure 8. Top: expressing joint densities with copulas; Bottom: Multivariate joint density factorized with a vine copula.

In practice, in order to construct a vine, one chooses two components: (1) the structure, the set of trees $T_m = (V_m, E_m)$ for $m \in [M-1]$ and (2) the pair copulas, the models for $c_{j_e,k_e|D_e}$ for $e \in E_m$ and $m \in [M-1]$.

Corresponding algorithms exist for both of those steps and in the rest of the paper, we assume consistency of the vine copula estimators for which we use the implementation by (Nagler & Czado, 2016), namely its Python version -`pyvinecopulib`.

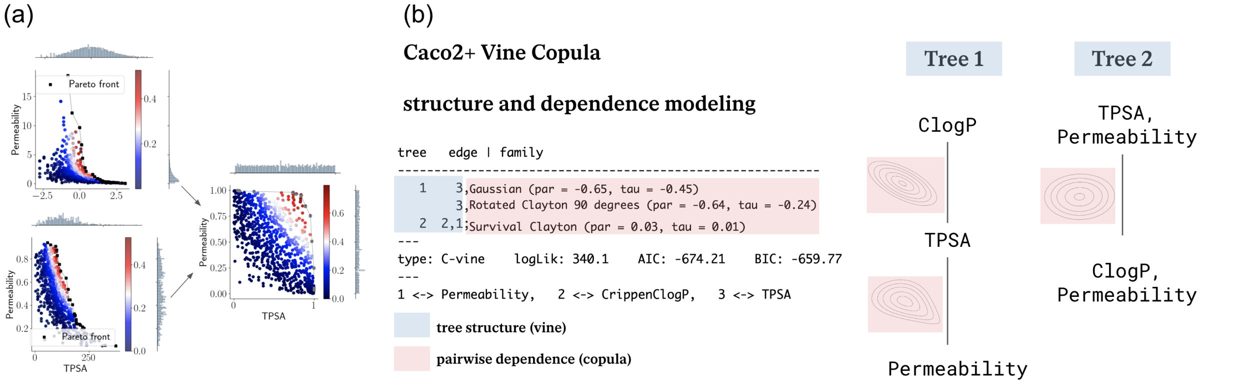
D.1. Complexity of the copula estimation

The complexity for fitting the vine copulas as currently implemented scales as $O(n_{\text{total}} Mp)$ in the case of density estimation, where n_{total} is the number of points being fit and p is the vine depth. Both estimation and sampling involve a double loop over M and p with an internal step scaling linearly with n_{total}. The computational complexity is linearly impacted by L (number of predictive samples). For BOtied v1, we have $n_{\text{total}} = n$, so this translates to $O(nLMp)$, where n is the number of query candidates, while for BOtied v2, we use the expectation of the posterior samples only, so $n_{\text{total}} = nL$ and the complexity remains as $O(nMp)$. Note that $p \in [M]$ can be truncated for additional efficiency.

D.2. Copulas in BO

In the low-data regime, empirical Pareto frontiers tend to be noisy. When we have access to domain knowledge about the objectives, we can use it to construct a model-based Pareto frontier using vine copulas. This section describes how to incorporate (1) the known correlations among the objectives to specify the tree structure (vine) and (2) the pairwise joint distributions (including the tail behavior), approximately estimated from domain knowledge, when specifying the copula models.

[4] In this section, we use the standard notations for densities (f) and distributions (F) as commonly done in the copula literature.



```
tree   edge | family
------------------------------------------------------------
   1     3,Gaussian (par = -0.65, tau = -0.45)
         3,Rotated Clayton 90 degrees (par = -0.64, tau = -0.24)
   2   2,1;Survival Clayton (par = 0.03, tau = 0.01)
---
type: C-vine    logLik: 340.1    AIC: -674.21    BIC: -659.77
---
1 <-> Permeability,   2 <-> CrippenClogP,   3 <-> TPSA
```

Figure 9. (a). Regardless of the distributions of the marginals, the CDF score from a copula is the same. (b) An example of explicitly encoding domain knowledge in a BO procedure by imposing the blue tree structure (specifying the matrix representation of the vine) and pink selection of pairwise dependencies (choice of parametric/non-parametric family).

The advantages of integrating copula-based estimators for our metric and acquisition function are threefold: (i) scalability from the convenient pair copula construction of vines, (ii) robustness wrt marginal scales and transformations thanks to inherent copula properties Theorem 4.7 and Equation 4.8, and (iii) domain-aware copula structures from the explicit encoding of dependencies in the vine copula matrix, including choice of dependence type (e.g., low or high tail dependence).

Figure 9 illustrates the use of copulas in the context of optimizing multiple objectives in drug discovery, where data tends to be sparse. In panel (a) we see that, thanks to the separate estimation of marginals and dependence structure, different marginal distributions have the same Pareto front in the PIT space, in which we evaluate our CDF scores. Hence, with copula-based estimators, we can guarantee robustness without any overhead for scalarization or standardization of the data as required by other counterparts. In panel (b) we show how we can encode domain knowledge of the interplay between different molecular properties in the Caco2+ dataset. Namely, permeability is often highly correlated with ClogP and TPSA, with positive and negative correlation, respectively, which is even more notable at the tails of the data (see panel (a) and Appendix F). Such dependence can be encoded in the vine copula structure and in the choice of copula family for each pair. For example, we specified a rotated Clayton copula so that the tail dependence between TPSA and permeability is preserved.

E. Other multivariate CDF estimators

Copulas are not the only statistical tool we can use for estimating multivariate CDFs. Here we include three more alternatives for the CDF acquisition function based on: empirical CDF, kernel density estimation and multivariate Gaussian. However, not all of them enjoy the fast computation in higher dimensions as vine copulas, and they all lack the guarantees for invariance to scale and transformation. The sensitivity analysis doesn't show significant difference between the performance of the estimators, thus, the choice can be made based on users' preference.

We want to highlight the general form of our proposed score, by showing how the CDF estimator as well as the BOTIED acquisition function can be computed with other parametric and non-parametric estimators. In what follows we include:

- **Multivariate Gaussian CDF** ($\mathrm{BOtied}_{\mathrm{mvn}}$) We compute the sample mean and covariance (μ, Σ) from the training data, and than use a closed-form analytical solution to obtain the multivariate Gaussian distribution with which we can compute our CDF scores. $\hat{F}(\mathbf{x}) = P(\mathbf{X} \leq \mathbf{x}) \quad where \quad \mathbf{X} \sim (\mu, \mathbf{\Sigma})$

- **Empirical CDF** ($\mathrm{BOtied}_{\mathrm{empirical}}$) The empirical cumulative distribution function is a step function that jumps up by $\frac{1}{n}$ at each of the n data points. Its value is the fraction of observations of the measured variable that are less than or equal to the specified value: $\hat{F}_n(t) = \frac{\#elements \quad in \quad sample \quad < t}{n} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}_{X_i < t}$

- **Kernel density estimation** ($\mathrm{BOtied}_{\mathrm{KDE}}$) Finally, we can also use a mixture of density estimators, such as KDE. Since the density is $\hat{f}(x) = \frac{1}{m}\sum_{i=1}^{M} f_i(x)$, then the joint CDF is the mixture of CDFs, $\hat{F}(x) = \frac{1}{m}\sum_{i=1}^{M} F_i(x)$. With a Gaussian kernel we have $f_i(x) = \frac{\phi(x-x')}{\sigma}$ and analogously $F_i = \frac{\Phi(x-x')}{\sigma}$ where σ is the kernel bandwidth.

Table 3. HV indicators (computed in the original units) and I_{CDF} different batch size on for the Penicillin dataset. Higher is better and best per column is marked in bold. We report the average metric across twenty random seeds along with their standard error in parentheses.

	Penicillin (M=3, q=1)		**Penicillin (M=3, q=2)**		**Penicillin (M=3, q=4)**	
	I_{CDF}	**HV**	I_{CDF}	**HV**	I_{CDF}	**HV**
BOtied v1	0.15(0.06)	**32.69e4(1.78e4)**	**0.33(0.09)**	**34.08e4(2.7e4)**	**0.31(0.08)**	33.55e4(2.4e4)
BOtied v2	**0.26(0.08)**	31.05e4(2.11e4)	0.31(0.07)	30.06e4(2.21e4)	0.29(0.07)	**33.34e4(2.2e4)**
NParEGO	0.19(0.08)	31.31e4(1.96e4)	0.18(0.06)	30.79e4(2.26e4)	0.18(0.08)	31.67e4(1.4e4)
NEHVI	0.16(0.09)	30.72e4(2.06e4)	0.19(0.08)	32.04e4(1.85e4)	0.19(0.01)	32.2e4(2.8e4)
Random	0.34(0.05)	10804(112305)	0.21(0.11))	30.8e4(1.86e4)	0.18(0.09)	30.65e4(1.5e4)

F. Experimental detail and additional results

We executed batched BO simulations with sequential greedy optimization and varying batch sizes $q \in \{1, 2, 4\}$. The number of iterations T varied across the experiments.

Sequential greedy optimization In batch BO, we seek joint optimization over the q design points, so the decision variable is effectively $q \times d$-dimensional. When q is large, we may employ a sequential greedy scheme, where the q designs are selected in series by fantasizing observations at the predictive mean of already-selected designs and conditioning on them to select the next design (Wilson et al., 2018). For the baseline acquisition functions supported in the BoTorch, we use the `optimize_acqf` function with `sequential=True`.

Other parameters include: the initial data size N_0, the size of the pool N, and the number of predictive posterior samples L. We fixed the size of the pool relative to the selected batch, at $N/B = 100$. We also fixed $L = 20$, which was found to yield good sample coverage and a stable BOtied acquisition value.

Unless otherwise stated, the surrogate model was a multi-task Gaussian process (MTGP) with a Matern kernel implemented in `BoTorch` (Balandat et al., 2020) and `GPyTorch` (Gardner et al., 2014). The inputs and outputs were both scaled to the unit cube for fitting the MTGP, but the outputs were scaled back to their natural units for evaluating the respective acquisition functions.

F.1. Branin-Currin

Branin-Currin (d=2, M=2 Belakaria et al., 2019) is a composition of the Branin and Currin functions featuring a concave Pareto front (in the maximization setting). We maximize

$$f_1(x_1, x_2) = -\left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - r\right)^2 + 10(1 - \frac{1}{8\pi})\cos(x_1) + 10$$

$$f_2(x_1, x_2) = -[1 - \exp\left(-\frac{1}{2x_2}\right)]\frac{2300x_1^3 + 1900x_1^2 + 2092x_1 + 60}{100x_1^3 + 500x_1^2 + 4x_1 + 20},$$

where $x_1, x_2 \in [0, 1]$. We used $T = 30$.

F.2. DTLZ2

We took two configurations of DTLZ2 with d=6, M=4 and d=7, M=6 (Deb & Gupta, 2005).

F.3. Penicillin production

The penicillin production problem (d=7, M=3; Liang & Lai, 2021) simulates the penicillin yield, time to production, and undesired byproduct for seven input parameters of the production process.

F.4. Caco2+

For the Caco2 problem ($M = 3$; Wang et al., 2016) the objective is to identify molecules with maximum cell permeability. Here, permeability describes the degree to which a molecule passes through a cellular membrane. This property is critical for drug discovery (DD) programs where the disease protein being targeted resides within the cell (intracellularly). In each experiment, a molecule $\boldsymbol{x}_i$ is applied to a monolayer of Caco2 cells and, after incubation, the concentration c of $\boldsymbol{x}_i$ is measured on both the input and output side of the monolayer, giving c_{in} and c_{out}(Van Breemen & Li, 2005). The ratio $c_{\text{out}}/c_{\text{in}}$ is then treated as the final permeability label y_i^p.

Cellular membranes are composed of a complex mixture of lipids and other biomolecules. In order to enter and (passively) diffuse through a membrane, molecule $\boldsymbol{x}_i$ should interact favorably with these biomolecules and/or avoid disrupting their packing structure. Increasing the lipophilicity (logP) of $\boldsymbol{x}_i$ is thus one strategy to increase permeability. However, increasing logP often results in promiscuous binding of $\boldsymbol{x}_i$ to non-disease related proteins, which can lead to undesired side-effects. As such, we seek to minimize the computed logP (clogP, y_i^l) in our optimization task and note that this could directly compete with (i.e., harm) permeability.

Lastly and related, common objectives during MPO in DD settings include increasing the affinity and specificity of target binding. As opposed to non-specific lipophilic interactions as above, polar contacts (such as hydrogen bonds) between drug molecules and proteins often result in higher affinity and more specific binding. We compute the topological polar surface area (TPSA, y_i^t) of each candidate $\mathbf{x}_i$ as one indicator of its ability to form such interactions and seek to maximize it in our optimization. As with decreasing logP, increasing TPSA can negatively impact permeability and we thus consider it a competing objective.

It is important to note that the treatment of each of these optimization tasks as unidirectional (max or min) is a simplification of many practical DD settings. There is often an acceptable range of each value that is targeted, and leaving the bounds in either direction can be problematic for complex reasons. We direct the reader to (D. Segall, 2012) for a comprehensive review.

For fitting the MTGP on the Caco2+ data, we represent each input molecule as a concatenation of fingerprint and fragment feature vectors, known as fragprints (Thawani et al., 2020) and use the Tanimoto kernel implemented in `GAUCHE` (Griffiths et al., 2022).

F.5. Real-world datasets for data-driven evolutionary multi-objective optimization

To evaluate BOtied in real-world scenarios, we include experiments over three datasets from the DDMOP benchmark (He et al., 2020). Differently from the synthetic test functions which have analytical solutions, DDMOP, proposes a testbed of complex objective functions, approximated by expensive numerical simulations, formulated as Data-Driven Multi-objective Optimization Problems, hence the name DDMOP. We select the three scenarios [5]:

- **Car cab** from (Deb et al., 2009), optimization of vehicle frontal structure, $d = 11, M = 9, N = 120$. The objectives represent the performance of the car cab, through weight of the car, fuel economy, acceleration time, road noise at different speed, and roominess of the car.
- **Power system** (Kavasseri & Srinivasan, 2011), $d = 11, M = 3, N = 120$. The objectives relate to the performance of a power system, active power loss, voltage deviation and generation cost based on the optimal joint placement of phasor measurement units.
- **Neural network performance** (Jin & Sendhoff, 2008), $d = 17, M = 2, N = 186$. One objective denotes the complexity of the network in terms of nonzero weights, while the second objective is the classification error rate of the neural network.

We negate all three problems to turn them into maximization objectives. As we approach these problems from a realistic perspective, we ran the experiments with batch size $q = 4$, $T = 20$ and initial set of $24, 9, 10$ points respectively. The reference points for each dataset were chosen as the minimum per objective decreased by 1e-3.

F.6. Details on wall clock time

Details for Figure 7. For all acquisition functions, we report the wall clock time per single acquisition function evaluation as computed on a Tesla V100 SXM2 GPU (16GB RAM) and an Intel Xeon CPU @ 2.30GHz (240GB RAM). A single call

[5] Which have more than 100 data points in the latest version of that datasets we were provided by the authors at the time of writing this paper

Table 4. HV and I_{CDF} across three DDMOP datasets. Mean and sterr in brackets.

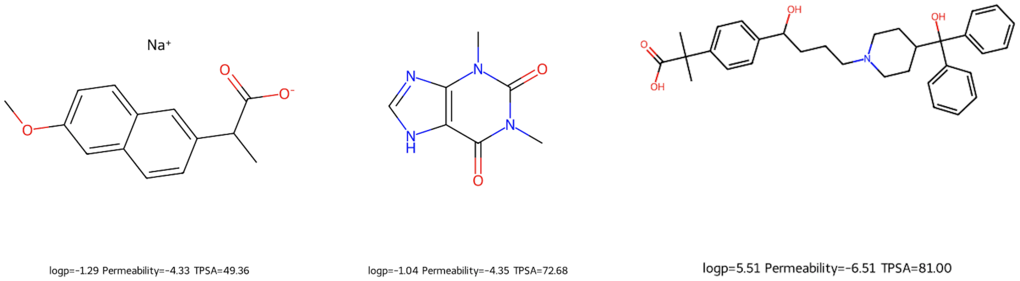
	CarCab (M=9)		**PowerSystem (M=3)**		**NeuralNetwork (M=2)**	
	I_{CDF}	**HV**	I_{CDF}	**HV**	I_{CDF}	**HV**
BOtied v1	**0.095(0.01)**	2.39e5 (0.21e5)	**0.732(002)**	**0.0235(0.001)**	**0.852(0.01)**	**0.512(0.02)**
BOtied v2	0.091(0.02)	**2.44e5(0.30e5)**	**0.732(002)**	**0.0235(0.001)**	0.849(0.02)	0.511(0.02)
NPareGo	0.094(0.02)	2.36e5(0.23e5)	0.729(0.02)	0.0233(0.001)	0.847(002)	0.509(0.03)
NEHVI	0.090(0.02)	2.3e5(0.35e5)	0.724(0.02)	0.0218(0.002)	0.847(0.02)	0.502(0.03)
random	0.079(0.02)	2.24e5(0.31e5)	0.721(0.05)	0.0222(0.002)	0.850(0.02)	0.504(0.03)

takes in the surrogate inference results for the candidate pool as well as the previously evaluated points and computes the acquisition scores.

- BC $M{=}2$: q batch size = 4, number of predictive samples=40, initial n = 10, pool size = 40

- DTLZ $M{=}4$: q batch size = 4, number of predictive samples=20, initial n = 50, pool size = 40

- DTLZ $M{=}6$: q batch size = 4, number of predictive samples=20, initial n = 50, pool size = 40.

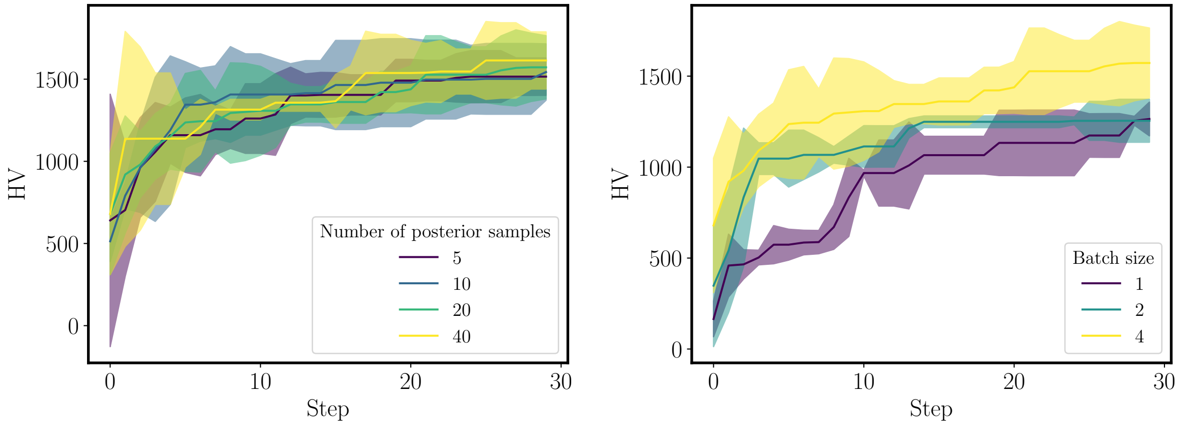

Na+
O
O-
O
N
N
O
N
NH
O
O
OH
OH
N
OH
logp=-1.29 Permeability=-4.33 TPSA=49.36
logp=-1.04 Permeability=-4.35 TPSA=72.68
logp=5.51 Permeability=-6.51 TPSA=81.00

(a) Desirable properties

(b) Undesirable properties (high log p, low permeability)

Figure 10. Examples of molecules in the Caco2+ dataset. The goal for the Caco2+ problem is to minimize log p, maximize permeability, and maximize TPSA.

G. Ablation studies



(a) Varying the number of posterior samples

(b) Varying the batch size

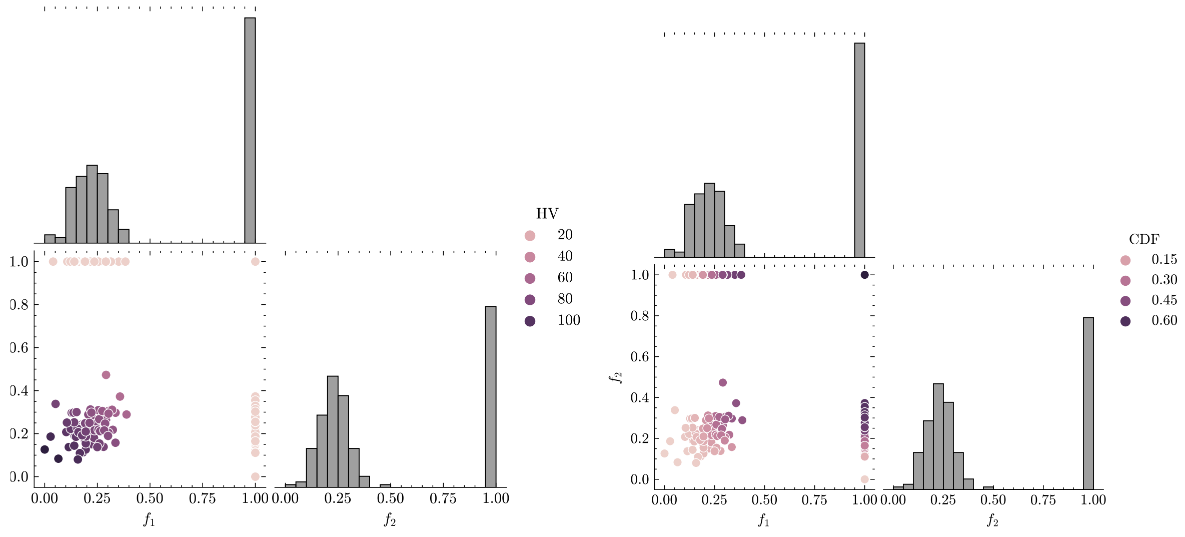
Figure 11. Ablation studies for BOtied v1. (a) BOtied is robust to the number of posterior samples drawn. (b) Increasing the batch size improves acquisition, particularly as it improves the CDF fit quality in earlier iterations.

H. Importance of invariance to scaling and monotonic transformations

Consider a scenario that occurs commonly in drug design, where both objectives are "zero-inflated," meaning that they are distributed with an abundance of zero (null) values plus a wide dispersion of valid, non-null values (Figure 12). We linearly scale the objective values to the $[0, 1]$ range and define the "null" value at 1 for both objectives. The color gradient corresponds to the indicator value at each solution ($q = 1$). With HV, we need to specify a reference point, set at [1.1, 1.1] in this case. Because the having a null value in even one of the objectives makes the HV small for a solution, the HV indicator can only distinguish points with non-null values in *both* objectives (lower left corner) from all other points. It assigns near-zero scores to regions with null values in only one objective (upper left and lower right corners), which should be included in the approximate Pareto front. On the other hand, the CDF indicator effectively identifies the full Pareto front, including the upper left and lower right corners.

I. Discontinuous Pareto fronts

Copulas can flexibly model multi-modal outcome distributions as well, particularly those with discontinuous Pareto fronts. In Figure 13, we consider objectives distributed as a mixture of two well-separated Gaussians. On the 200 simulated observations, we fit a CDF with a Gaussian mixture copula and kernel density estimation (KDE) marginals. The zero level line of the CDF closely traces the true Pareto front (solid red curve).



(a) HV indicator

(b) CDF indicator

Figure 12. A scenario where both objectives are "null-inflated," meaning that they are distributed with an abundance of null values plus a wide dispersion of valid, non-null values. We linearly scale the objective values to the $[0, 1]$ range and define the "null" value at 1 for both objectives. The color gradient corresponds to the indicator value at each solution (q=1). (a) With HV, we need to specify a reference point, set at [1.1, 1.1] in this case. Because the having a null value in even one of the objectives makes the HV small for a solution, the HV indicator can only distinguish points with non-null values in *both* objectives (lower left corner) from all other points. It assigns near-zero scores to regions with null values in only one objective (upper left and lower right corners), which should be included in the approximate Pareto front. (b) The CDF indicator effectively identifies the full Pareto front, including the upper left and lower right corners.

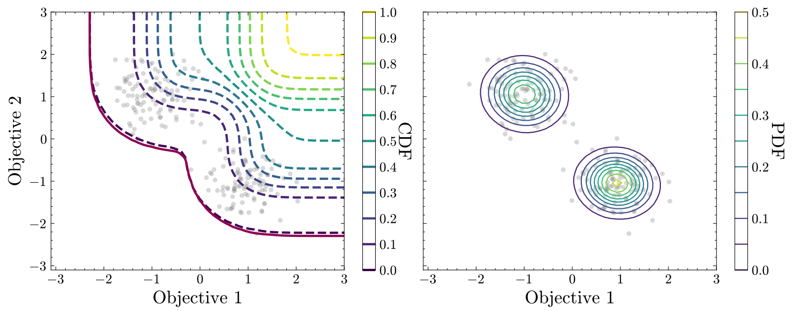


Figure 13. Level lines of the CDF (left) and the PDF (right) from a CDF fit with Gaussian mixture copula and kernel density estimation (KDE) marginals, based on 200 observations simulated from a mixture of two Gaussians (gray dots). The zero level line of the CDF closely traces the true Pareto front (solid red curve).