Exploration of Preferential Bayesian Optimization

Kayoon Kim 6646790

March 25, 2025

Abstract

Preferential Bayesian Optimization (PBO) has emerged as a promising approach for optimizing functions where direct measurements are unavailable, instead relying on pairwise preference feedback. State-of-the-art PBO methods utilize acquisition functions like EUBO and qEUBO to guide the optimization process. However, comparative analyses of these functions in real-world settings remain limited, and the impact of human biases on preference elicitation is poorly understood. This study addresses these gaps through three investigations using synthetic functions and Bosch's industrial dataset. We found that acquisition function performance varies with problem dimensionality. Furthermore, our experiments successfully demonstrated PBO's ability to learn preferences from industrial data, validating its practical applicability. Importantly, we discovered that preference data with even minimal bias significantly compromises optimization performance. These findings provide valuable guidance for PBO implementation and highlight the critical need for bias detection and mitigation strategies to ensure reliable optimization outcomes.

1 Introduction

Bayesian Optimization (BO) is a sequential model-based approach framework to find an optimum of black-box functions with expensive or time-consuming evaluations [10]. It affected various domains such as interactive user interfaces, robotics, and reinforcement learning. Mathematically, it considers the problem of finding a global maximum of an unknown objective function f

$$\mathbf{x}^* = \arg\max_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}) \tag{1}$$

Additionally, the black-box function f isn't assumed as closed form but be able to evaluated at any query point x in the domain.

BO is a stochastic and sequential. It utilizes a probabilistic surrogate model, which consists of a prior distribution of the objective function. We sequentially refine the model based on an observation via Bayesian posterior updating. During the process, we sequentially update acquisition functions that suggest the model which next point should explore. In detail, x_{n+1} is suggested by a maximum acquisition function value. Then we get our updated beliefs, Bayesian posterior.

Although BO has been introduced as a critical solution for various domains, there are still many problems. It is difficult to observe objective function values directly because of computational costs and measurement noise. One potential solution is to provide the decision maker (DM) with a representation of pairwise comparisons, which could assist them in determining the preferred option. In this context, Preferential Bayesian Optimization (PBO) emerges as a robust methodological framework for addressing these challenges, as outlined in the seminal work by Gonzalez et al [6].

As in standard BO, PBO also considers the problem of finding a global maximum of an unknown objective function f where X is a design space of interest. In BO, it observes the function f through direct objective measurements y. However, a key distinction between PBO and BO lies in the accessibility of these observations, y. While BO can directly access the observations, PBO operates under the assumption that the observations are not directly accessible. Instead, PBO observes the preference of values.

Furthermore, similar to BO, a PBO algorithm has two key components. The first one is a probabilistic surrogate model of the DM's latent utility function. The second component is an acquisition function (AF) that calculated from the probabilistic surrogate model. It evaluates the informational utility derived from DM's preference feedback regarding their optimal choice among q alternatives, quantifying the value of preference in the design space.

The evolution of PBO has been marked by significant advances in acquisition functions, with recent developments. As a state-of-the-art, there are two acquisition functions *expected utility of the best option*, EUBO [7] and qEUBO [2] which overcame downsides of previous acquisition functions.

Based on the foundation, the paper aims to explore PBO through three research questions.

1. How do acquisition functions perform in comparison to one another?
2. Is it feasible to utilize real-world data to derive human preferences?
3. Are there human biases that could potentially impede the efficacy of PBO? What are the mitigation strategies?

As a result, the paper provides three key contributions. First, we conduct a comparison of novel acquisition functions, EUBO and qEUBO. Second, we demonstrate the practical viability of PBO using Bosch's industrial data in the real world. Finally, we examine the role of human cognitive factors in preference-based optimization. These research objectives address several challenges in the field. Real-world validation of PBO methods is limited. Although previous studies [2] [7] [9] evaluated their algorithms using real preference data, it's mostly from previous studies' dataset. The paper is meaningful that it validated algorithms with entirely novel and real industrial dataset. Furthermore, our study contributes comprehensive novel acquisition function comparisons. Lastly, the paper suggests unexplored DM's factors in preference elicitation. Here, we defined a term "bias" to describe situations where a comparison is made incorrectly by factors. Investigating the performance of different acquisition functions under different bias levels is introduced by Astudillo et al [2]. However, our study introduces delicate human-related biases which is available enough to happen in human preference feedback-based experiments.

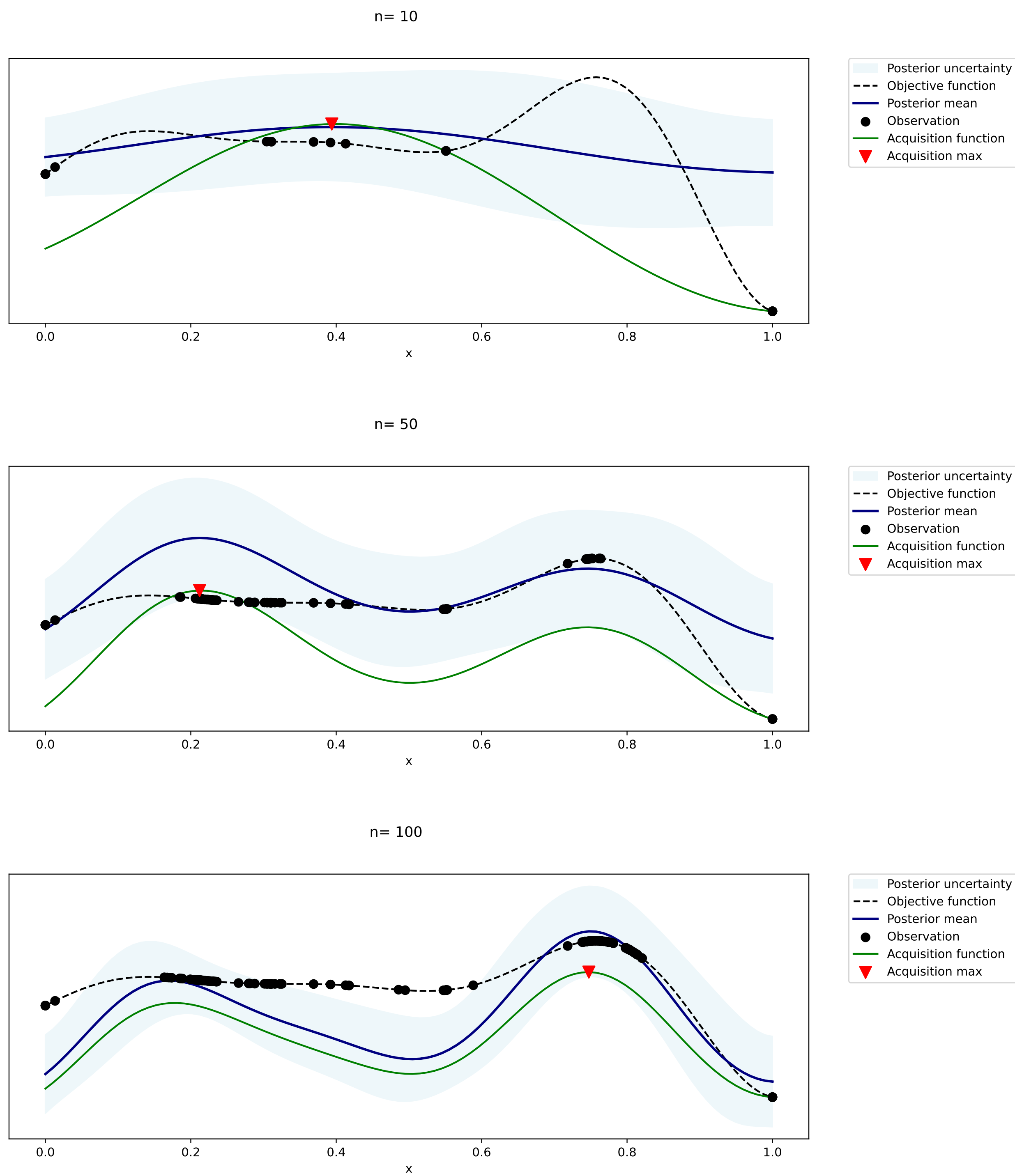

Figure 1: Forrester function optimization process with different numbers of observations ($n = 10, 50, 100$). The plots show the posterior uncertainty, objective function, posterior mean, observations, acquisition function, and acquisition maximum at different stages of the optimization. Given that PBO is incapable of evaluating true objective function, it tries to capture its shape.

2 Methods

2.1 Data

The present study utilizes synthetic functions-that is, artificial test problems designed to evaluate optimization algorithms. Specifically, the Forrester function and the Six-hump camel function were employed. Forrester function is 1-dimensional in both the x and y variables. To negate the loss values, the function was as illustrated in Figure 2 (left) given that Botorch [3], Bayesian optimization library built on PyTorch, is set to maximize by default for PBO. Six-hump Camel function is 2-dimensional in the x variable and 1-dimensional in the y variable.

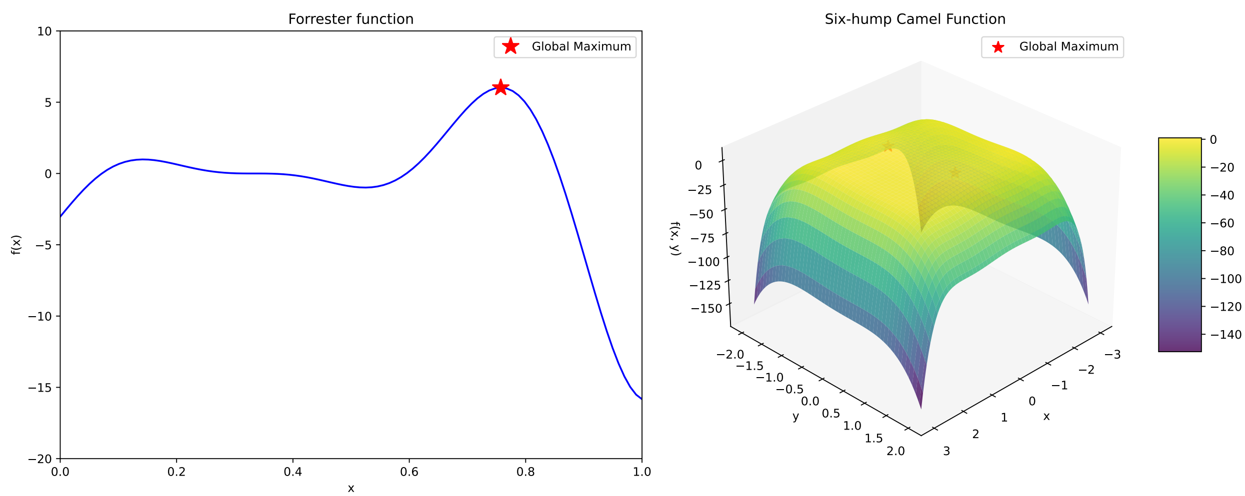

Figure 2: Forrester function (left) is represented by a graph with one maximum point, Six-hump camel function (right) with two maximum points, indicated by red stars.

In addition, the algorithms were evaluated using Bosch's internal tabular benchmark. The design space of the benchmark is 5-dimensional including integer and continuous values while the losses y, were 4-dimensional with range of 0 to 1. Given that the data is not subject to user preferences and is sparse, a surrogate model based on the data was implemented. The process involved selecting a specific data point as a reference point. This reference point was determined by the experts in the field, as an ideal trade-off. Next, we projected each loss value to this reference point, thereby acquiring the corresponding direction. These directions were then designated as weights. Subsequent to this, each loss was multiplied by its respective normalized weight, resulting in a 1-dimensional scalarized loss. This can be represented by

$$\text{scalarized\_loss} = w_1 \cdot W + w_2 \cdot X + w_3 \cdot Y + w_4 \cdot Z \tag{2}$$

where W, X, Y, Z denote the individual loss components from the dataset, and $w_i \in \mathbb{R}$ for $i = 1, \ldots, 4$ are the corresponding weights that determine the relative importance of each loss. With 5-dimensional features and 1-dimensional scalarized loss, the K-Nearest Neighbors (KNN) model was implemented as a surrogate model.

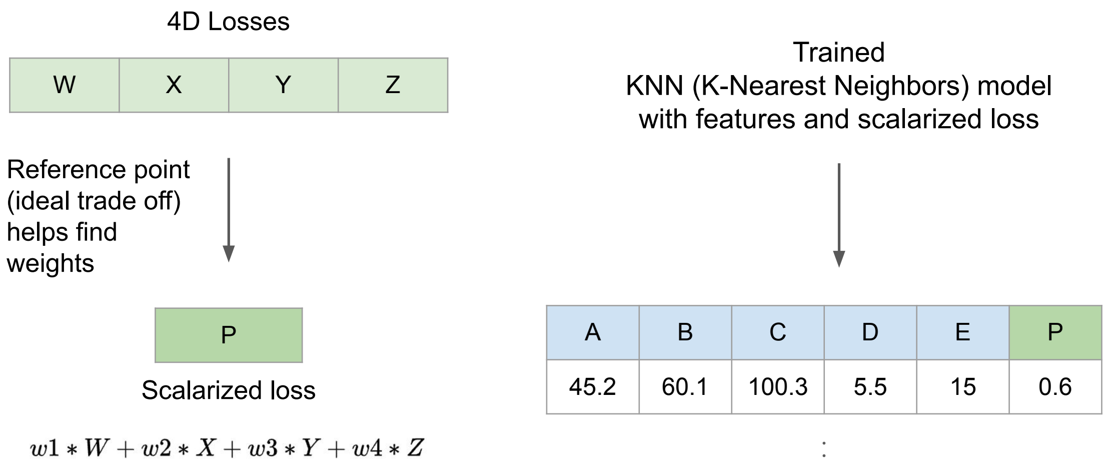

Figure 3: Visual explanation of generating a surrogate model using Bosch internal benchmark data where W, X, Y, Z are the loss components, A, B, C, D, E are feature components and P represents the scalarized loss.

2.2 Algorithms

2.2.1 EUBO (Expected Utility of Best Options)

In the EUBO paper [7], authors propose Bayesian Optimization with Preference Exploration (BOPE) as a novel PBO algorithm framework. The algorithm proceeds in two stages: preference exploration (PE) and experimentation with two models, f and g. In this context, f represents the outcome function, i.e, an approximation of true objective function, while g represents the preference function, i.e. an approximation of utility function. The objective of the algorithm is to learn and approximate the true f (true objective function) and true g (true utility function) in detail, thereby solving $max(g_{true}(f_{true(x)})$. By making use of preferences, the algorithm is able to optimize expensive multi-output functions with fewer resources. In our case, the true outcome function (f_{true}) will serve as a ground-truth function, a surrogate model. A Gaussian Process (GP) approximation

f is employed to train on available data, then making predictions on f_{true}. The true utility function (g_{true}) will be a function with parameter settings input and a true preference value (e.g., ratings, scores, etc.) provided by experts or users which sometimes cannot be measured or expensive to test all settings. The preference model g will be a function with parameter settings input and a predicted preference value learned from pairwise comparisons. Obtaining a f_{true} and g_{true} is costly, and the models approximate these efficiently.

In order to model the outcome function f_{true}, the paper employs a multi-output GP. The input is the design points x, which are in d dimensions, and the output is the outcome vectors y, which are in k dimensions. To illustrate, the outcome model f of our surrogate model f_{true} comprises 5-dimensional design points and 1-dimensional loss. The outcome model was implemented with a Matérn 5/2 ARD covariance function. Subsequently, the posterior is updated in order to predict outcomes at novel design points. Similarly, the preference function g_{true} is also modeled with a GP, g. In the case of a query constituted by two outcome vectors (y_1, y_2), which indicate the preference of the DM, the Laplace approximation posterior g. To illustrate, the model is capable of discerning the preference of the DM, indicating which parameter they favor over the others. The preference model was implemented with a radial basis function (RBF) ARD kernel.

During a PE stage, an algorithm generates a query consisting of two outcome vectors for the DM to compare. DM interactively expresses preferences over multiple pairs of outcome vectors that does not involve the collection of new values from f_{true}. In regard to the acquisition function, EUBO is the expected value of the maximum utility between two outcomes under the current preference model, g. In detail, EUBO identifies informative pairs where the model is confident about preference, thereby enabling the model to learn the preference more effectively. For practical reason, BoTorch implementation of EUBO utilizes one of PE strategies that samples many plausible achievable regions based on data from true objective function f_{true}.

In comparison to non-preferential strategies such as EI and UCB in BO, there are notable distinctions. First, the search space of EUBO is the outcome space y, distinct from the design space x as seen in EI and UCB. Second, the objective of EUBO is to identify the most informative preference queries for the model, whereas non-preferential strategies aim to balance exploration and exploitation in the search space. Third, EUBO utilizes the DM's pairwise preferences, whereas non-preferential strategies rely on direct function evaluations.

During an experimentation stage, an experimentation strategy chooses a set of points in the design space where f_{true} is evaluated. They used the qNEIUU as an acquisition function. Thus, in a PE stage, previous DM queries and previous experiment evaluations are used to select the outcome vectors over which DM preference is elicited. In an experiment stage, all the information gathered up to that point is used to determine the design points to evaluate.

2.2.2 qEUBO

qEUBO extends to queries with $q > 2$ alternatives. The primary distinction between EUBO and qEUBO lies in their respective acquisition strategies. While EUBO employs a sequential approach, optimizing individual points, qEUBO extends this framework through batch processing, enabling parallel evaluations. This approach allows for a much broader exploration of the parameter space, thereby enhancing computational efficiency. Moreover, the batch-based framework generates a diverse range of candidates and information gains across multiple points. In our experiment however, we utilized single-point selection strategy for both methodologies which they are now the same method.

2.3 Experiments

A total of three experiments were conducted in line with our research questions. Initially, a comparison was made between EUBO and qEUBO as acquisition functions utilizing BoTorch's PairwiseGP model. The PairwiseGP model is a probit likelihood GP that learns via pairwise comparison data, with two other baselines, random search and Upper Confidence Bound (UCB) in BO. The BO was utilized as a baseline for two reasons. First, for a computational comparison. For the optimizer, BO requires a single computation of the objective function, whereas PBO requires two computations due to the necessity of determining which value is promising for the next iteration. In detail, when both BO and PBO identify a subsequent candidate, BO only has one promising candidate. Conversely, PBO has two candidates, and selects which one is more promising, resulting in a more computational process. To illustrate this, consider the scenario in which PBO evaluates 100 queries, whereas BO would only evaluate 50. In addition, the model has been included to compare the relative performance of the two approaches. PBO is limited to acquiring pairwise input preferences, whereas BO is capable of directly accessing the ground truth from synthetic functions. This suggests that BO is a superior approach to PBO. The objective of this study was to investigate the performance of PBO in comparison to a known ground truth model. We report results across two synthetic test functions.

Second, we employed the Bosch internal benchmark, which is a test function constructed from real-world data, to determine whether the PBO could infer human preferences. The experimental setup was identical to the first one. Finally, aligning with the model comparison experiment, the objective was to assess the model's performance in the presence of potential bias, which could arise in use cases or real-world experiments. For instance, when the options are very similar or the differences are subtle, DMs tend to make arbitrary choices because they struggle to identify meaningful differences between options. In bias experiment, the PairwiseGP model was used. The EUBO strategy, an acquisition function designed for use with biased data (1, 3, 5, 7, 10%), was employed. The term "biased data" is used to describe situations where, on average, a comparison is made incorrectly, when presented with random pairs of points with comparable function values. This occurs at each percent (1, 3, 5, 7, or 10%) of the time. To illustrate, consider a 100-iteration scenario, where a data set comprising 10% inaccurate comparisons is expected.

In order to ascertain whether our implementation aligns with the same result, we also conducted an identical experiment based on the Astudillo et al. [2], which utilized biased data. It has a difference of generating biased data in previous experiment. Instead of utilizing fixed probability for generating biased data, the paper adopts a logistic likelihood function assuming that the DM's responses may not always align with the underlying utility function. For instance, if proposal queries are too similar, DM is less likely to make the right choice. Among q proposals, $X = (x_1, \ldots, x_q) \in \mathbb{X}^q$, the DM responds to the preferred alternative that is denoted by $r(X) \in \{1, \ldots, q\}$, where $r(X) = i$ if x_i is the one selected by the DM. A parametric likelihood function $L(\cdot\,; \lambda) : \mathbb{R}^q \to \mathbb{R}^q$ was modeled such that

$$\mathbb{P}(r(X) = i \mid f(X)) = L_i(f(X); \lambda), \tag{3}$$

$$L_i(f(X); \lambda) = \frac{\exp(f(x_i)/\lambda)}{\sum_{j=1}^{q} \exp(f(x_j)/\lambda)}, \tag{4}$$

for $i = 1, \ldots, q$, where $\lambda \geq 0$ is the the bias level parameter, which controls how consistent the choices are. A total of three test problems with varying bias levels were examined: low, medium, and high. The value of λ was set as 0.0575, 0.1416 and 0.2943, correspondingly. The experimental setting was configured to match the paper setting. Utilizing a single-point selection strategy with $q = 2$ alternatives, a 6-dimensional Ackley benchmark was employed, and a high bias level was tested. For the sake of comparison, the code from the original paper was utilized, which can be found at `https://github.com/facebookresearch/qEUBO`. During the experiment, the number of queries was set to 150, with 50 replications.

With the exception of the previous comparison experiment, all experiments were run for a total of 100 queries with 20 trials. The number of queries represents the number of times we ask for feedback or comparisons between different options. Consequently, the execution of 100 queries results in the comparison of the values 100 times. The metrics employed included simple regret and the best observed value were used. Simple regret is defined as the difference between the best value found by the optimization and the true global optimum. It denotes as

$$r_t = f(x^*) - f(x_t) \tag{5}$$

where $f(x^*)$ is the true global optimum and $f(x_t)$ is the best value found by the algorithm at time t. The best observed value, on the other hand, involves the logging of the best value determined by the optimization.

3 Results

3.1 Comparative Analysis of Acquisition Functions

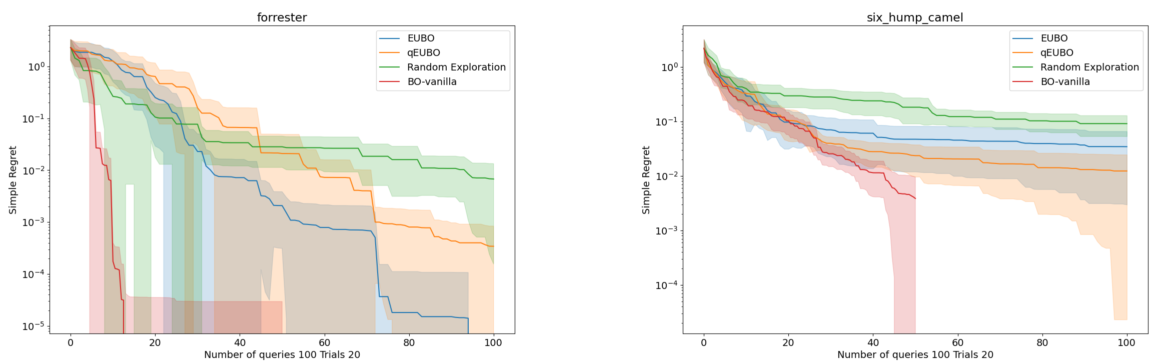

Figure 4: Comparison of EUBO and qEUBO against random search and UCB (BO) on a 1-dimensional problem (Forrester, left) and 2-dimensional problem (Six-hump camel, right)

For the Forrester benchmark, EUBO demonstrated superior performance in comparison to qEUBO. Conversely, in the context of Six-hump camel, the efficacy of qEUBO was better than that of EUBO. The UCB (BO) exhibited a more superior performance on both benchmarks.

3.2 Real-world Preference Learning with Industrial Data

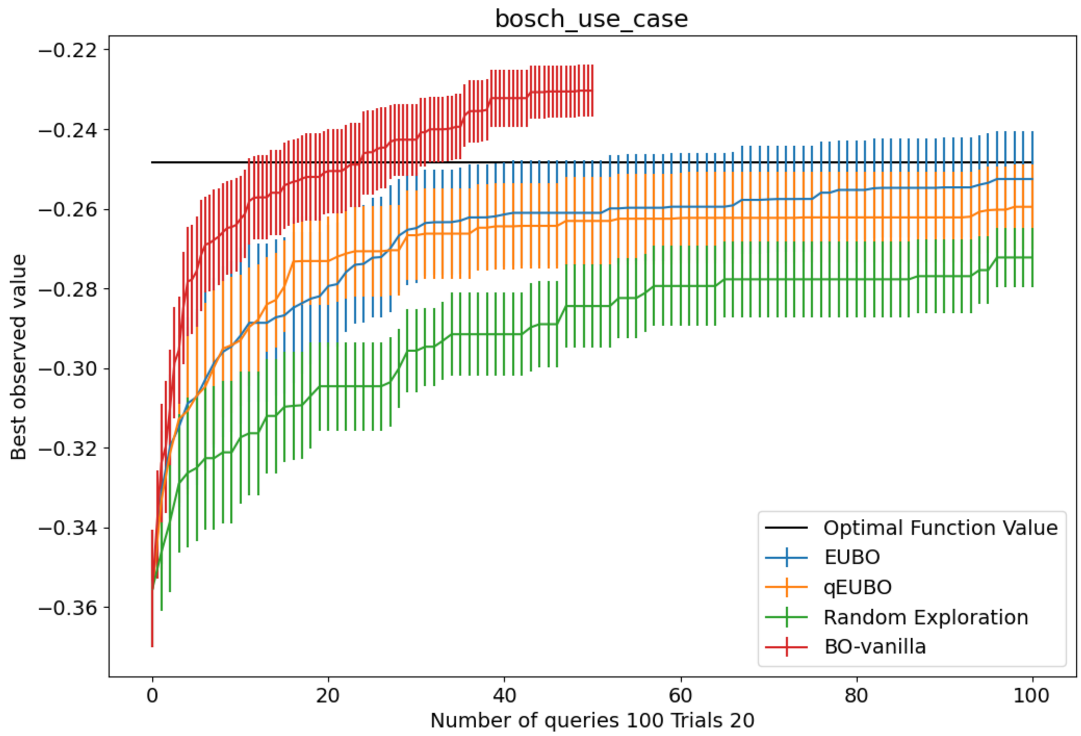

Figure 5: Comparison of EUBO and qEUBO with random search and UCB (BO). The black line indicates the optimal function value of the model

In this experiment, EUBO demonstrated a marginally superior performance in comparison to qEUBO. Notably, UCB (BO) surpasses the optimal value of the KNN model. It should be noted that the value attained by BO corresponds to the optimal value of the data prior to the implementation of the KNN model. Although this suggests that it has found the optimum, it is important to note that it may not align with the reference point (preference).

3.3 Human Cognitive Biases in Preference Elicitation

3.3.1 Fixed biased probability experiment

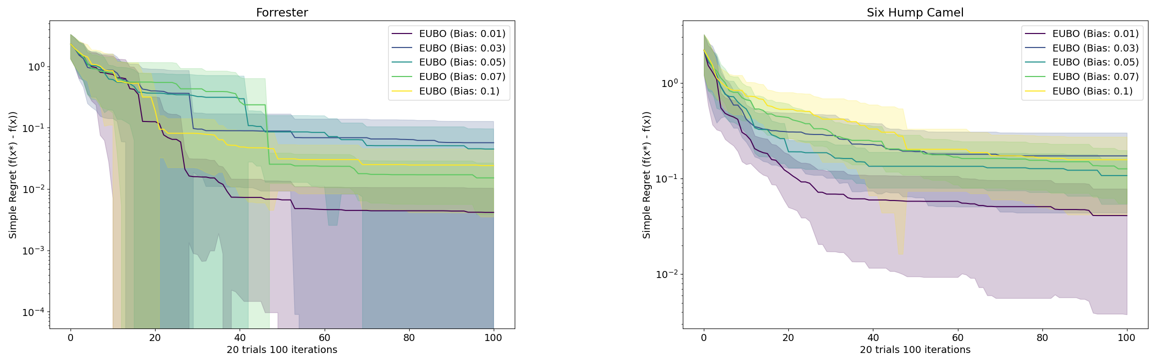

Figure 6: Comparison of different probabilities of fixed biased data using EUBO for each benchmark (1, 3, 5, 7, 10%), Forrester (left) and Six-hump camel (right)

The results indicate that a bias of 1% is optimal. However, the results for other bias ratios indicate that the model becomes confused without clear patterns. For example, 3% biased data is expected to have second-best performance, but it did not. After over 1% of biased data, the model was unable to accurately detect the preference, as shown in the result plot. In detail, except for 1% of biased data, simple regret values increased, suggesting that tracking bias is a crucial aspect for models to consider.

3.3.2 Output based biased probability experiment

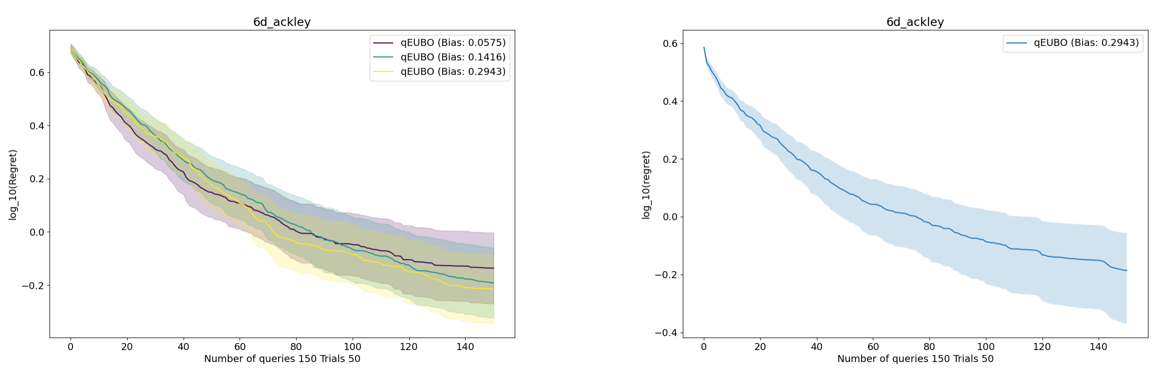

Figure 7: log10(optimum value - objective value at the maximizer of the posterior mean) per DM query with high noise. The result of our code (left), the result of the code from the paper (right) [2]

The results from the original code and our implementation yielded comparable outcomes, approximately -0.2, as shown in Figure 7. However, these values have poorer performance, not aligning with the original result. The observed value from the paper [2] ranged from -0.25 and -0.50. This discrepancy indicates the necessity for further investigation and refinement in future studies.

3.3.3 Human biases in sequential decision-making

Given the evidence that biased data has the potential to hinder the efficacy of PBO, this study proposes human biases in future in-person experiments, a sequential option pair setting, based on the literature research. Consequently, we present mitigation strategies.

Human biases in sequential decision-making are the focus of extensive research in a variety of academic disciplines, including economics, psychology, and cognitive science. Anchoring bias, is a process whereby individuals are influenced by specific information presented prior to making a judgment [5]. In a future experiment setting, for instance, participants set an initial configuration as an anchor to which they compare when making a judgment. Confirmation bias is a type of cognitive bias, defined as the tendency to seek for confirmatory evidence rather than disconfirming information that would challenge existing beliefs [1]. In subsequent experimentation, individuals with a strong conviction against artificial intelligence may exhibit more skepticism toward a proposal from PBO than a human expert. Position bias is defined as the tendency of participants solving pairwise tasks to select left answer or choice [4]. In our case, there is a possibility that participants might select the left choice from the user interface even though they preferred the right one. Order effect bias is the tendency to be influenced by the sequence in which items are presented in a pair [4]. For instance, an experiment investigating a subject's sensitivity to small electrical shocks demonstrated that the response to the second shock could be strongly modified by the first shock. Consequently, in our experiment, it is plausible that participants may exhibit bias in a specific parameter based on the sequence in which the parameters are presented.

In the following, we propose a series of mitigation strategies grounded in the potential human biases that may be involved. From the participants' perspective, the Consider-the-opposite [8] strategy is employed. This strategy consists of considering alternative viewpoints or outcomes that are opposite to one's initial thoughts. To mitigate potential biases, it is important that participants are made aware of this prior to the experiment. From an experimental perspective, we propose the implementation of masked and shuffled proposals. To mitigate the potential confirmation bias against AI-generated proposals, we propose a strategy of masking and shuffling each human expert and AI-generated configuration pairs. This approach ensures that participants are unable to distinguish between the human expert and the AI-generated configuration. In addition, the validation proposals are useful to determine whether the participants exhibit any biases. Specifically, the validation process can detect uniform spammers [11] who repeatedly submit the same answer. In this scenario, participants can repeatedly select either left or right options, and these selections can be identified through validation suggestions. Finally, the user interface is designed to be straightforward and intuitive, ensuring that participants do not become biased by the placement of options buttons.

4 Discussion

The paper provides answers to all three research questions. A comparative analysis was conducted on the performances of two novel acquisition functions, EUBO and qEUBO, in comparing them with random search and UCB. Additionally, we demonstrated the feasibility of deriving human preferences using real industrial data. Finally, we demonstrated that human biases play a critical role in PBO algorithms. Furthermore, we have identified related human biases and have proposed mitigation strategies.

There are a few points meaningful to discuss based on the results. Initially, the findings raises the following question: Are biased models always not useful? The conclusion is that it depends on which model we want to implement. In scenarios where the objective is to implement customized models for each customer, a biased model might be advantageous. Conversely, if the objective is to employ the model as a public resource, it is important to eliminate the bias. Secondly, future research should prioritize a more comprehensive study of biases in PBO. From an algorithmic perspective, the performance of qEUBO did not reach the levels mentioned in Astudillo et al. [2] during the bias experiment. Detecting and correcting the biases while in the loop could be the future work. Additionally, from the perspective of experiment setting, the occurrence of biases in in-person experiments within PBO could be a subject for future investigation. During the literature review, it was challenging to identify human biases related to the PBO experiment setting, given its quite different from the experiment setting of psychological or cognitive science stimuli, such as electric shocks or short audios. Despite the proposal of mitigation strategies, there are elements that remain beyond our control. One such element pertains to the mood of the participants. Furnham at el [5] has noted that mood significantly influences anchoring effects on affective factors. Specifically, participants in sad mood are more susceptible to the heuristic bias of anchoring compared to their counterparts in neutral or happy mood. Although it is not possible to alter moods, it may be possible to mitigate bias through the use of these unchangeable factors that could affect the performance of PBO.

References

[1] D. Arnott. Cognitive biases and decision support systems development: A design science approach. *Information Systems Journal*, 16(1):55–78, 2006.

[2] R. Astudillo, Z. J. Lin, E. Bakshy, and P. I. Frazier. qeubo: A decision-theoretic acquisition function for preferential bayesian optimization, 2023.

[3] M. Balandat, B. Karrer, D. R. Jiang, S. Daulton, B. Letham, A. G. Wilson, and E. Bakshy. Botorch: A framework for efficient monte-carlo bayesian optimization, 2020.

[4] R. J. Beaver and D. V. Gokhale. A model to incorporate within-pair order effects in paired comparisons. *Communications in Statistics*, 4(10):923–939, 1975.

[5] A. Furnham and H. C. Boo. A literature review of the anchoring effect. *The Journal of Socio-Economics*, 40(1):35–42, 2011.

[6] J. Gonzalez, Z. Dai, A. Damianou, and N. D. Lawrence. Preferential bayesian optimization, 2017.

[7] Z. J. Lin, R. Astudillo, P. I. Frazier, and E. Bakshy. Preference exploration for efficient bayesian optimization with multiple outcomes, 2022.

[8] C. G. Lord, M. R. Lepper, and E. Preston. Considering the opposite: A corrective strategy for social judgment. *Journal of Personality and Social Psychology*, 47(6):1231–1243, 1984.

[9] Q. P. Nguyen, S. Tay, B. K. H. Low, and P. Jaillet. Top-k ranking bayesian optimization, 2020.

[10] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2016.

[11] D. Zhu and B. Carterette. An analysis of assessor behavior in crowdsourced preference judgments. In *Proceedings of SIGIR 2010 CSE Workshop*, pages 21–26, 2010.