Supplement to ”Preferential Bayesian Optimization Improves the Efficiency of Printing Objects with Subjective Qualities”

Acquisition Functions for PBO

The performance of acquisition functions for preferential Bayesian optimization (PBO) has not been studied as extensively as those used in regular Bayesian optimization. We performed a simulation study that compared eight PBO algorithms using different acquisition functions, some proposed in the literature (e.g., Brochu, M. Cora, & De Freitas, 2010; Siivola et al., 2021; Lin, Astudillo, Frazier, & Bakshy, 2022) and others our own variations on those, to identify one to use in the current study. Six were variants of q Noisy Expected Improvement (qNEI; Brochu et al., 2010; Siivola et al., 2021) and two were variants of Expected Utility of Best Option (EUBO; Lin et al., 2022). All assume Gaussian Process (GP) modeling of a latent utility function (Chu & Ghahramani, 2005; Rasmussen & Williams, 2006). Given noisy observations of preferential choices, qNEI selects the next q (≥ 2) candidate points in the design space (to be presented to the oracle) based on the highest qNEI value by posterior simulation, whereas EUBO directly maximizes the expected utility of the best option and can be applied only to the case of two design points (i.e., $q = 2$).

The acquisition functions tested were implemented using BoTorch, a framework for Monte-Carlo Bayesian optimization in Python (Balandat et al., 2020). The variants of qNEI that we implemented differed in two factors: (a) the choice of a reference point in the design space from which further improvement (i.e., the expected improvement) is calculated; and (b) whether such a reference point is included in the set of candidate points or not. In defining the expected improvement, reference points of the following three types were compared.

- *Updated Best* (UB): The point that maximizes the current, estimated utility function (i.e., $\mathrm{UB}_k = \arg\max_x \mu_k(x)$, where $\mu_k(x)$ is the GP posterior mean function estimated after k experiments), the expected improvement from which defines an acquisition function known as a *knowledge gradient* (P. Frazier, Powell, & Dayanik, 2009; P. I. Frazier & Wang, 2016).
- *Incumbent Best* (IB): The point, among the previously selected points so far, that has the greatest value on the current, estimated utility function (i.e., $\mathrm{IB}_k = \arg\max_{x \in X_k} \mu_k(x)$, where $\mu_k(x)$ is the posterior mean function and X_k is the set of all previously selected points after k experiments).
- *Previous* (Prev): Any one of the previously selected points, which is sampled according to the GP posterior, and the qNEI of which is averaged over all such samples in a Monte-Carlo simulation.

Below is a list of the eight acquisition functions along with short descriptions of them.

- *qNEI-UB-addBest*: Selects $q - 1$ points that jointly maximize expected improvement from the updated best (UB) point, and add the UB to make q points.

- **qNEI-UB-woBest**[1]: Selects q points that jointly maximize expected improvement from the UB point.
- *qNEI-IB-addBest*: Select $q-1$ points that jointly maximize expected improvement from the incumbent best (IB) point, and add the IB to make q points.
- *qNEI-IB-woBest*: Selects q points that jointly maximize expected improvement from the IB point.
- *qNEI-Prev-addBest*: Selects $q-1$ points that jointly maximize expected improvement from a posterior-simulated point within previously selected points, and add the IB to make q points.
- *qNEI-Prev-woBest*: Selects q points that jointly maximize expected improvement from a posterior-simulated point within previously selected points.
- *EUBO-woBest*: Selects 2 points that maximize the expected utility of the best option (EUBO). Supports only $q = 2$.
- *EUBO-UB-addBest*: Selects 1 point that maximizes the expected utility of the best option (EUBO) using the UB as a previous winner, and add the UB to make 2 points. Supports only $q = 2$.

The above acquisition functions, except the two 'EUBO' functions, allow any number of options to be compared against one another. The preference given in the form of a rank order among all options can be expressed as an exhaustive set of pairwise preference comparisons and fed to the pairwise comparison model of Chu and Ghahramani (2005). For instance, a rank order among three options given as $a > b > c$ (i.e., the option a is preferred to the option b, which itself is preferred to the option c) is equivalent to three pairwise preference judgements of $a > b, b > c$, and $c < a$, assuming transitivity of comparisons.

Simulation Setup

Two variables were manipulated: (a) the number of options to choose from (q); and (b) the dimensionality of the design space or input domain (d). A simple, unimodal function (i.e., with a single peak on a multidimensional domain) was chosen as the target utility function that was to be learned and maximized in an experiment. Since the true utility is known in this simulation, each algorithm's performance was measured simply by the utility value at the updated best point after each experiment (i.e., true utility at $\arg\max_x \mu_k(x)$, where $\mu_k(x)$ is the GP posterior mean function estimated after k experiments).

Each simulated campaign consisted of 80 experiments, in each of which a simulated oracle indicated preferences in response to q options generated by an acquisition function. Noisy preferential choices were simulated by adding Gaussian noise (with $\sigma = 5\%$ of the true utility function's range) to each option's true

[1] This is the algorithm that was implemented in the PBO-AM campaigns of the present study.

utility and rank-ordering q options' noisy utility values. For each simulation condition (i.e., a combination of q and d), performance measured in each experiment was averaged across 100 independent campaigns with as many random number generator seeds.

Results

Figure 1 shows the performance of acquisition functions put to the test in four different simulation conditions. Also tested was 'random' selection of q points from a uniform distribution on the design space, which served as a baseline. Each line in the figure indicates how closely, on average, the updated best points across experiments approach the true best point in terms of true utility normalized between 0 and 1. The upper two panels, showing the results from the $q = 2$ and $q = 3$ conditions in a design space of $d = 6$, convey two main points: (a) Except a few acquisition functions (*qNEI-Prev-addBest*, *qNEI-Prev-addBest*, and *Random*) that performed consistently worse, the performance of all others was indistinguishable, with all showing similar trajectories that have a similar maximum; and (b) having the oracle compare three, as opposed to two, choice alternatives led to greater efficiency of inference.

Comparison between Panels b and c shows that the efficiency decreases (slightly lower asymptote), as expected, as the dimensinality of design features increases (i.e., $d = 6$ vs. $d = 8$). Lastly, inspection of Panel d in comparison to Panel c suggests that providing more choice alternatives beyond $q = 3$ marginally increases the efficiency of inference, especially early in a campaign, but its practical benefit is negligible.[2]

We selected *qNEI-UB-woBest* along with $q = 3$ for our PBO-AM campaigns because it balances performance and task difficulty. The algorithm is reasonably efficient when there are three options, and comparison of three options is relatively simple for an oracle. When $q > 3$, asking a participant to give a full rank order can be cognitively taxing and lead to inconsistencies in ordering when options are difficult to judge. Moreover, a participant responding with a single 'batch winner' and coding the response into pairwise comparisons (e.g., Option a being the winner of a, b, c and d is encoded into $a > b, a > c$, and $a < d$ and fed to the model) turns out not to be substantially lower in the efficiency of inference (Siivola et al., 2021). Theoretical considerations also influenced our decision. Defining the expected improvement from the updated best point, rather than confining the reference point within previously selected points, made better sense. The reference point being added to the set of candidates seemed to be redundant because the q points selected by qNEI almost always included a point close to the updated best.

References

Balandat, M., Karrer, B., Jiang, D. R., Daulton, S., Letham, B., Wilson, A. G., & Bakshy, E. (2020). BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization. In *Advances in neural*

[2]Another reason why the performance gap between the $q = 3$ and $q = 5$ conditions is small must be that the oracle provided only a batch winner in the case of $q = 5$ but full rank ordering when $q = 3$.

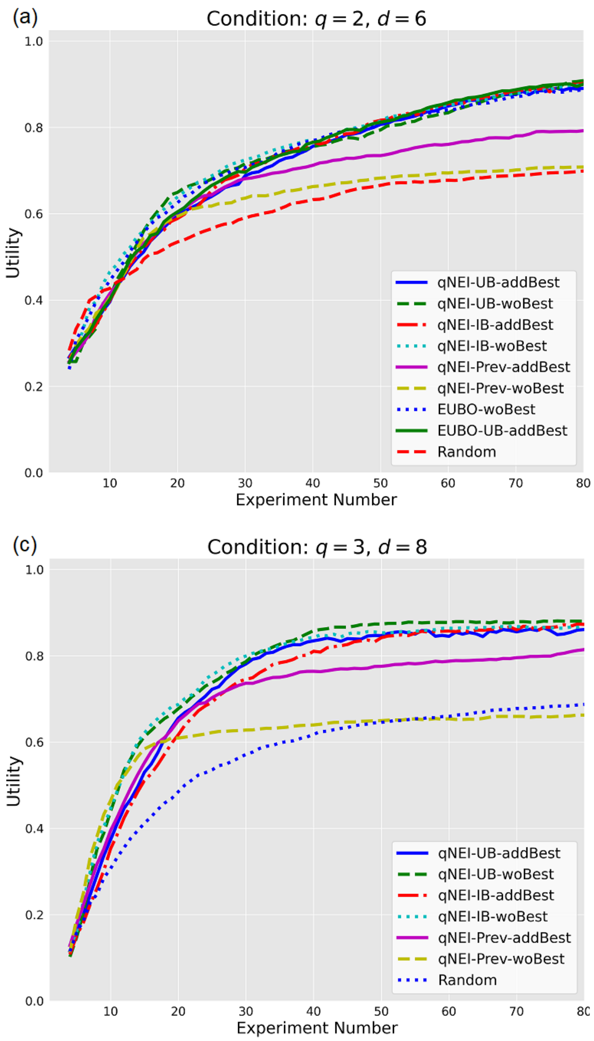

Figure 1: Performance comparison of the PBO algorithms under various simulation conditions. The symbol q represents the number of alternatives to choose from and d represents the dimensionality of design space. Each curve is the mean averaged over 100 independent simulation runs.

information processing systems 33. Retrieved from `http://arxiv.org/abs/1910.06403`

Brochu, E., M. Cora, V., & De Freitas, N. (2010). A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *ArXiv:1012.2599*, 1–49. Retrieved from `https://doi.org/10.48550/arXiv.1012.2599`

Chu, W., & Ghahramani, Z. (2005). Preference learning with Gausian processes. In *Proceedings of the twenty-second international conference on machine learning (ICML)* (pp. 137–144). Bonn, Germany.

Frazier, P., Powell, W., & Dayanik, S. (2009). The knowledge-gradient policy for correlated normal beliefs. *INFORMS Journal on Computing*, *21*, 599–613.

Frazier, P. I., & Wang, J. (2016). Bayesian optimization for materials design. In T. Lookman (Ed.), *Information science for materials discovery and design* (pp. 35–75). Springer International Publishing Switzerland.

Lin, Z. J., Astudillo, R., Frazier, P. I., & Bakshy, E. (2022, 28–30 Mar). Preference exploration for efficient bayesian optimization with multiple outcomes. In G. Camps-Valls, F. J. R. Ruiz, & I. Valera (Eds.), *Proceedings of the 25th international conference on artificial intelligence and statistics* (Vol. 151, pp. 4235–4258). PMLR.

Rasmussen, C. E., & Williams, C. K. I. (2006). *Gaussian processes for machine learning.* Cambridge, MA: MIT Press.

Siivola, E., Dhaka, A. K., Andersen, M. R., Gonzalez, J., Moreno, P. G., & Vehtari, A. (2021). Preferential batch Bayesian optimization. In *IEEE international workshop on machine learning for signal processing* (pp. 1–6). Gold Cost, Australia. doi: 10.1109/MLSP52302.2021.9596494