Kernelized Offline Contextual Dueling Bandits

Viraj Mehta [1]   **Ojash Neopane** [2]   **Vikramjeet Das** [2]   **Sen Lin**   **Jeff Schneider** [1 2]   **Willie Neiswanger** [3]

Abstract

Preference-based feedback is important for many applications where direct evaluation of a reward function is not feasible. A notable recent example arises in reinforcement learning from human feedback on large language models. For many of these applications, the cost of acquiring the human feedback can be substantial or even prohibitive. In this work, we take advantage of the fact that often the agent can choose contexts at which to obtain human feedback in order to most efficiently identify a good policy, and introduce the *offline contextual dueling bandit* setting. We give an upper-confidence-bound style algorithm for this setting and prove a regret bound. We also give empirical confirmation that this method outperforms a similar strategy that uses uniformly sampled contexts.

1. Introduction

In many decision making problems in information retrieval, question answering, clinical trials, advertising, and other fields, feedback about the performance of a particular choice is only available in the form of a preference between several options. Such feedback often takes the form of a user clicking on a link in the wild but can also be collected from labelers in an attempt to understand their underlying preferences and train a system to optimize them. This is an especially useful method for collecting human feedback because humans are unreliable in giving scalarized feedback compared to the accuracy of their preferences (Ouyang et al., 2022). Often, these settings come with some context that can provide information about the associated distribution over preferences. This could be the search term, some information about a trial participant, or the prompt in a question

*Equal contribution [1]Robotics Institute, Carnegie Mellon University, USA [2]Machine Learning Department, Carnegie Mellon University, USA [3]Department of Computer Science, Stanford University, USA. Correspondence to: Viraj Mehta <virajm@cs.cmu.edu>.

answering setting.

Recently, these techniques have seen large amounts of attention when popularized through reinforcement learning from human feedback (RLHF). RLHF is one of the major techniques for aligning large language models (LLMs) for use as a chat assistant or for other specialized applications after pretraining on a sequence modeling objective. In these applications, human raters are provided with a prompt and several possible responses taken from the LLM. They are asked to rank the human responses based on their preferences given the prompt. This process requires a relatively large number of samples from human raters (tens of thousands) in order for the alignment process to succeed. This can incur large costs for the data collection. For more specialized problems than a general chatbot assistant, the feedback required may be impractical or expensive relative to the desired application.

In many applications, the feedback is collected *online* as the policy is learned. Under these circumstances, the contexts are typically assumed to be drawn from an (unknown) stationary probability distribution and the agent's object is to quickly find a near optimal decision rule. Currently, in the RLHF setting, the prompts presented to the model in order to sample responses and then to the raters are typically sampled uniformly from a response set intended to be representative of the test-time distribution.

In such cases, the aforementioned online setting does not allow us to fully take advantage of the problems structure – we can control which prompts and responses are presented to the human raters for feedback and we are not interested in the performance of the actions chosen during labeling, just the performance of the policy at test time afterwards. Instead of the standard contextual bandit problem it is then more appropriate to consider the so-called *offline contextual* bandit (Char et al., 2019) where we are additionally allowed to select the contexts for which we receive feedback. By leveraging this control over this less restrictive data-generating process, we show how to select contexts and actions in order output policies with stronger optimality guarantees without the need to collect more data.

Here, we tackle the special case of pairwise feedback where a pair of actions are compared given a particular context. Following Xu et al. (2020), we first reduce the problem

of finding the optimal action given pairwise feedback to finding the action that optimizes the *Borda function* given a particular context. The Borda function is the probability that for a particular context, a selected action is preferred over another action selected uniformly at random. We select contexts which maximize the uncertainty over the Borda 'value function' and then select one action optimistically and the other uniformly.

In this work, we show that our method provably achieves suboptimality at most $O\left(\frac{L_1}{\sqrt{T}}\left(B + \Phi_T\sqrt{\log\frac{1}{\delta}}\right)\right)$ everywhere in the context space after T iterations with probability $1 - \delta$ under the assumption that the Borda function is bounded by B in RKHS norm. We also demonstrate on a distribution of synthetic problems that it performs well when implemented and outperforms a baseline with uniformly selected contexts and an optimistic policy as well as entirely uniform sampling.

2. Related Work

Learning from Comparative Feedback   There is a rich literature on reinforcement learning from comparative human feedback, including work by Fürnkranz et al. (2012), Akour (2014) and Christiano et al. (2017). Many of these works grappled with the heightened need for sample efficiency given the cost of acquiring human feedback. In particular, Christiano et al. (2017) made it feasible to use human feedback for deep reinforcement learning by training a reward model that is then used as the target for reinforcement learning. In their Atari test case, where naive deep RL would have required thousands of hours of gameplay, they were able to achieve superior performance with only 5,500 or several hours of human queries.

More recently, methods of using comparative human feedback have gained prominence as a means of improving the performance of language models. These methods have been shown to be effective at improving stylistic continuation (Ziegler et al., 2019), text summarization (Stiennon et al., 2020), translation (Kreutzer et al., 2018), semantic parsing (Lawrence & Riezler, 2018), review generation (Cho et al., 2018), and evidence extraction (Perez et al., 2019). However, while effective, incorporating human feedback brought substantial costs. For example, Stiennon et al. (2020) achieved significant improvements to baseline, but needed summaries on 123,169 posts from the TL;DR dataset generated by a small team of labelers (more than 21 persons) from Upwork, Scale, and Lionbridge to train.

This heavy-resource requirement is again reflected even in later, state-of-the-art work. Ouyang et al. (2022) focused on using RLHF to improve alignment of the GPT-3 model (at 175B parameters) with human values on a variety of directions, including toxicity, hallucinations, moral opinion,

and overall quality. The results are spectacular, with the 1.3B parameter InstructGPT matching the 175B GPT-3 in performance on a variety of tasks. Because the focus was ensuring representation both in the inputs to models used in real life and in the human feedback received, the team used 40 labelers and worked with a dataset of more than 100,000 examples.

A related paper from Zhu et al. (2023) also explores a simplified version of this problem and showed that under the strong assumption of a linear model given a known feature mapping, the policy obtained by optimizing the pessimistic MLE given a fixed dataset is provably optimal for learning in the k-wise comparison context. Given these strong assumptions, the authors point out that a G-optimal experimental design for online data collection as in Soare et al. (2014) would be maximally informative. However, these assumptions are unrealistic and do not represent the methods used in practice as reward model training is usually conducted over all layers of a deep model.

Dueling Bandits   At the same time, the bandit literature has also explored the effectiveness of comparative feedback ("dueling bandit") while considering the cost of acquiring such information. This was first studied by Yue et al. (2012) in settings where comparative information is relatively easy to extract but absolute rewards (*i.e.*, direct queries) are ill-defined and have no absolute scale. Later, Bengs et al. (2021) surveyed methods used in the online learning setting, where the trade off with cost of information is most acute, including those used in the online contextual dueling bandit setting by Dudík et al. (2015). These constraints motivate a kernelized approach that can incorporate the nonlinearities in the models used in practice.

Offline Contextual Bandit Optimization   When there are distinct phases of learning and then deployment, an agent can make maximal use of every example during learning to acquire information that can be exploited once deployed.

Char et al. (2019) introduce this idea for black-box function approximation by considering a setting where at test time the goal is to perform well on average across a context distribution while during learning the goal is to choose contexts and actions that are most useful for that goal. Given a reward function for each task, the authors proposed a multi-task version of Thompson sampling during the offline training phase, which allows provable regret bounds in that problem setting. We extend this setting from cardinal to ordinal rewards as is appropriate for comparative feedback.

In Li et al. (2023), the agent queries the states (or contexts, in a bandit setting) where the value function is most uncertain and acts optimistically. Combined with least-squares value iteration, this method leads to provable polynomial-

sample convergence in the worst-case error of the value function estimate in reinforcement learning in general, and as a corollary the setting from Char et al. (2019) as a special case. This sets the foundation that we will adapt to the comparative feedback setting.

3. Problem Setting

In this paper, we consider a dueling variant of the so-called offline contextual bandit problem introduced in Char et al. (2019). An instance of this problem is defined by a tuple $(\mathcal{X}, \mathcal{A}, f)$ where $\mathcal{X}$ denotes the context space, $\mathcal{A}$ denotes the action space and $f : \mathcal{X} \times \mathcal{A} \times \mathcal{A} \to [0, 1]$ is a preference function so that $f(x, a, a')$ denotes the probability that the action a is preferred to the action a' when the underlying context is x. We will design algorithms that operate under the following interaction protocol, which occurs for T time steps. During each time step $t \in [T]$, the agent selects a context $x_t \in \mathcal{X}$ and a pair of actions $a_t, a'_t \in \mathcal{A}$ and observes a binary random variable $R_t \sim \text{Bern}(f(x_t, a_t, a'_t))$ which equals one if a_t is preferred to a'_t (denoted $a_t \succ a'_t$) and zero otherwise.

We assume that the preference function takes the following form

$$f(x, a, a') = \sigma\left(r(x, a) - r(x, a')\right), \quad (1)$$

where $\sigma : \mathbb{R} \to [0, 1]$ is the *link function* and $r : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ is the reward function. Common link functions include the logistic function, which leads to the Bradley-Terry-Luce (BTL) model (Bradley & Terry, 1952) as well as the Gaussian CDF (Thurstone, 1927). We also place some additional assumptions on the reward function which we discuss at the end of this section.

Our objective within this protocol is to design algorithms that are able to quickly identify policies with small suboptimality. We define the suboptimality of a policy π as

$$\text{SubOpt}(\pi) = \sup_{x \in \mathcal{X}} \left(\sup_{a \in \mathcal{A}} r(x, a) - r(x, \pi(x)) \right). \quad (2)$$

We remark that this notion of suboptimality is much stronger than usual notions that look at the expected suboptimality of the final policy when the contexts are sampled from some known distribution. In contrast, the form of suboptimality we consider here looks at the worst-case context for each policy.

Before discussing the assumptions we place on the reward function, we first introduce a closely related function, called the *contextual borda function* f_r, which generalizes the borda function introduced in by Xu et al. (2020). The Borda function as introduced in Xu et al. (2020) for dueling-choice optimization is defined as the probability that a particular action a will be preferred over a random action a' uniformly sampled from the action space. We generalize this definition

to the contextual setting as follows, given as $f_r : \mathcal{X} \times \mathcal{A} \to [0, 1]$ where $f_r(x, a) = \mathbb{E}_{a' \sim U(\mathcal{A})}[P(a \succ a' \mid x)]$, where $U(\mathcal{A})$ is the uniform measure over the action space. It is clear from the definition that f_r and r will have the same maximizers.

We conclude this section by discussing the structural assumptions we place on the reward function as well as the contextual Borda function. Our first assumption restricts the reward and contextual Borda functions to be 'smooth' in an underlying Reproducing Kernel Hilbert Space (RKHS).

Assumption 3.1. Let κ denote a Positive Semi-Definite kernel and let $\mathcal{H}_\kappa$ denote its associated RKHS. We assume that $\|r\|_\kappa, \|f_r\|_\kappa \leq B$, where B is a known constant.

Note that this assumption is different than the standard assumption which only requires that r has bounded RKHS norm. This is due to the generality of our setting which allows for multiple different link functions. While this assumption is not ideal, it is difficult to bound the norm of f_r given a bound on the norm of r. We investigate this issue more in Section B where we empirically find that the norm of the Borda function is almost always smaller than the norm of the reward function.

Our second assumption relates the reward function to the contextual Borda function.

Assumption 3.2. Let $f_r^*(x) = \max_a f_r(x, a)$ and $r^*(x) = \max_a r(x, a)$. There exists a constant L_1 such that for every $x \in \mathcal{X}$, $a \in \mathcal{A}$ we have $\frac{1}{L_1}(r^*(x) - r(x, a)) \leq f_r^*(x) - f_r(x, a)$.

This assumption implies that differences in r will cause a similar magnitude of difference in f_r. In fact, when σ is Lipschitz continuous, it is sufficient for the Lipschitz constant of σ to be at least $1/L_1$ for this condition to hold.

4. Method and Analysis

At a high level, our approach reduces the dueling feedback problem to contextual optimization over a single action via the *contextual Borda function* introduced in Section 3. Once reduced appropriately, we apply techniques adapted from recent work on active exploration in reinforcement learning to construct a sampling rule and policy selection rule which allows us to output a policy with provably small suboptimality. Broadly, our sampling rule samples contexts at which there is maximum uncertainty over the Borda 'value function' and then compares the optimistic action with an action sampled uniformly from the action set.

4.1. Estimating The Contextual Borda Function

By design, we can easily estimate the contextual Borda function from data of the form $\{x_t, a_t \overset{?}{\succ} a'_t\}$, where the

contexts x_t and first actions a_t are arbitrary and the second actions a'_t are uniformly selected. In this work, we model the contextual Borda function using standard kernelized ridge regression (KRR) (Rasmussen et al., 2006). The key feature of KRR is that besides an estimate of the contextual Borda function after t observations $\mu_t(x, a)$, we can also estimate the uncertainty over the prediction $\sigma_t(x, a)$ using standard results; under the assumptions in Section 3 and given an value for β appropriate chosen for our confidence level, we can bound $|f_r(x, a) - \mu_t(x, a)| \le \beta \sigma_t(x, a)$ with the desired confidence.

4.2. Selecting Contexts and Actions

Our sampling rule builds on top of the one established in Li et al. (2023) —put simply the rule is to sample the state with the maximum uncertainty over the value function and then act optimistically. We will now present our algorithm which highlights how to extend these idea to the dueling setting via the contextual Borda function f_r.

For now, we assume that there is a known bonus term $\beta_t^{(r)}$ for all t. We can then define upper and lower confidence bounds $\overline{f_r^t}(x, a) = \mu_t(x, a) + \beta_t^{(r)} \sigma_t(x, a)$ and $\underline{f_r^t}(x, a) = \mu_t(x, a) - \beta_t^{(r)} \sigma_t(x, a)$. Our rule is to sample a context

$$x_t \in \arg\max_{x \in \mathcal{X}} \left(\max_{a \in \mathcal{A}} \overline{f_r^t}(x, a) - \max_{a \in \mathcal{A}} \underline{f_r^t}(x, a) \right). \quad (3)$$

Here, we are choosing a context that maximizes the difference between the optimistic 'value function' and the pessimistic 'value function' (both of which require a maximization over actions to compute).

We then optimistically choose an action by

$$a_t \in \arg\max_{a \in \mathcal{A}} \overline{f_r^t}(x_t, a). \quad (4)$$

After repeating this process T times, we output a pessimistic policy against the tightest lower bound we can find, which is the maximizer of all our lower bounds through the optimization process. Put formally we return $\hat{\pi}_T : \mathcal{X} \to \mathcal{A}$ such that

$$\hat{\pi}_T(x) \in \arg\max_{a \in \mathcal{A}} \max_{t \le T} \underline{f_r^t}(x, a). \quad (5)$$

From these pieces we construct the full algorithm, Borda-AE, which we present in Algorithm 1.

4.3. Bounding the regret of Borda-AE

Before proceeding with our algorithm's formal guarantees, we first introduce the *maximal-information gain* which plays an important role in our results. The maximum information gain over t rounds, denoted Φ_t, is defined as

$$\Phi_t = \max_{A \subset \mathcal{X} \times \mathcal{A}:|A|=t} I(r_A + \epsilon_A; r_A), \quad (6)$$

Algorithm 1 Borda-AE

1: **Input:** kernel function $\kappa(\cdot, \cdot)$, exploration parameters $\beta_t^{(r)}$, number of inital data n_0

2: Let $D_{n_0} = \{s_i, a_i \overset{?}{\succ} a'_i\}_{i=1}^{n_0}$ for s_i, a_i, a'_i uniform.

3: **for** $t = n_0 + 1, \ldots, T$ **do**

4:     Compute $\mu_t(\cdot, \cdot)$, $\sigma_t(\cdot, \cdot)$ using KRR.

5:     Choose x_t according to (3).

6:     Choose a_t according to (4), $a'_t \sim U(\mathcal{A})$.

7:     Let $D_t = D_{t-1} \cup \{(x_t, a_t \overset{?}{\succ} a'_t)\}$.

8: **end for**

9: Output a final policy $\hat{\pi}_T$ according to (5).

---

where $r_A = [r(x)]_{x \in A}$, $\epsilon_A \sim N(0, \eta^2 I)$ and $I(X; Y) = H(X) - H(X|Y)$ is the mutual information. With this definition, we are now ready to state our result.

Theorem 4.1. *Suppose we run Algorithm 1 with*

$$\beta_t^{(r)} = 2B + \sqrt{2\Phi_t + 1 + \log\left(\frac{2}{\delta}\right)}, \quad (7)$$

then, with probability at least $1 - \delta$, we have that

$$\text{SubOpt}(\hat{\pi}_T) \le O\left(\frac{L_1}{\sqrt{T}} \left(B + \Phi_T \sqrt{\log \frac{1}{\delta}}\right)\right). \quad (8)$$

Proof Sketch. At a high-level the proof of this result is as follows. First, we use standard results on KRR to show that our choice of $\beta^{(r)}$ guarantees that our confidence bands contain $f^r(x, a)$ with high probability simultaneously for all t and $x, a \in \mathcal{X} \times \mathcal{A}$. Next, we use assumption 3.2 to show that the suboptimality of the pessimistic policy induced by our estimated contextual borda function is small whenever we are able to estimate the contextual borda function well. Finally, we conclude the proof by showing that our sampling rule indeed allows us to estimate thet contextual borda function well.

Concrete Performance Bounds. To more concretely understand the performance of our algorithm, we instantiate our results for three commonly studied kernels: the linear, squared-exponential. For both of these settings, the scaling of the information gain is well known (see for example (Srinivas et al., 2010)). In the linear setting, we have that $\Phi_T = d \log T$ leading to a bound of $O\left(\frac{L_1}{\sqrt{T}} \left(dB \log T \sqrt{\log \frac{1}{\delta}}\right)\right)$. For squared exponential kernels we have $\Phi_T = O\left(\log(T)^{d+1}\right)$ leading to a suboptimality bound of $O\left(\frac{L_1}{\sqrt{T}} \left(B \log(T)^{d+1} \sqrt{\log \frac{1}{\delta}}\right)\right)$.

When compared to existing results for dueling bandits (Xu et al., 2020) as well as standard bandits (Chowdhury & Gopalan, 2017), we see that our suboptimality bounds

match, thus showing that our algorithm is able to achieve the same performance under a stronger performance metric.

5. Experiments

In order to assess the validity of our theory we have begun an experimental campaign starting with synthetic experiments that allow us to come as close as possible to the theoretical setting and empirically confirm our results. To do so, we implemented the regression using the BernoulliGP model provided by GPyTorch (Gardner et al., 2018). We use a Matérn kernal with automatic relevance detection with hyperparameters fit via maximum a posteriori optimized by the Adam algorithm (Kingma & Ba, 2014). We tested on distributions of synthetic reward functions generated by sampling a random linear combination of Random Fourier Features (Rahimi & Recht, 2007) derived from a squared exponential kernel. For each sampled reward function r, we used the Bradley-Terry model where $p(a \succ a' \mid x) = \frac{1}{1+\exp(r(x,a')-r(x,a))}$ to generate comparison data. For each trial we uniformly sampled $n_0 = 25$ datapoints and then selected data to observe until $T = 500$ total datapoints had been collected according to one of three methods:

- **Borda-AE**: our method, as described in Section 4.

- **Borda-Uniform**: uniform sampling of context and actions.

- **Borda-UCB**: uniform sampling of contexts with UCB actions as in Borda-AE.

This last method reduces to the method presented in Xu et al. (2020) when naïvely generalized to the contextual setting. All methods have the same test-time behavior of executing the action found by optimizing the pessimistic Borda function estimate for the test context. By optimizing the ground-truth reward function we were able to approximate the optimal policy and therefore estimate the regret of our policy against it. We give an example of the progression of our method for 1D context and 1D actions in Figure 2 as well as a comparison against Borda-Uniform and Borda-UCB in Figure 1. One can see that Borda-AE performs best both on median regret and on the maximum regret, which was the metric of interest in our theoretical analysis.

It is clear the method is quickly able to concentrate samples in regions that could plausibly be the optimum and it is similarly clear that the peaks in the acquisition function over contexts are sensible given the mean and uncertainty estimates of f_r.

6. Conclusion and Future Work

In this work we introduced the offline contextual dueling bandit setting and presented a first efficient algorithm for solving it in the kernelized setting. Though our method is
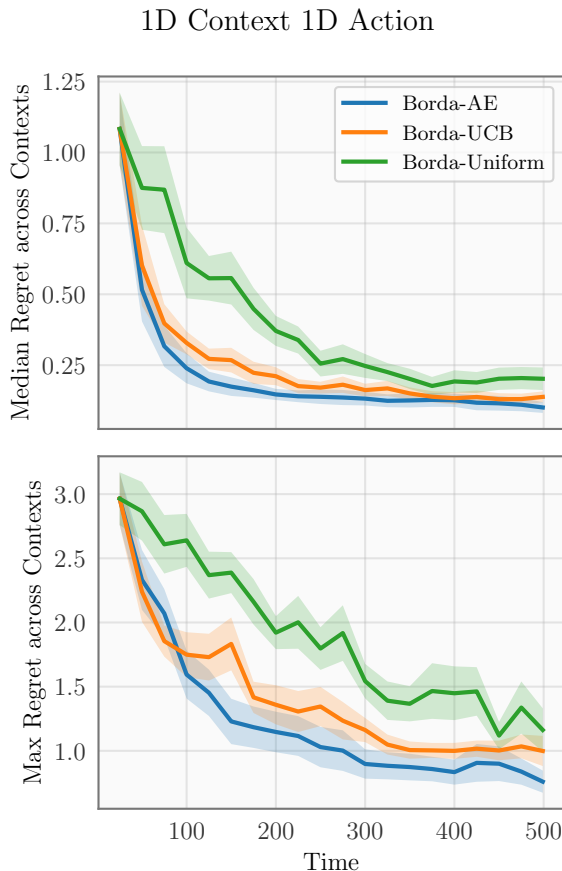

Figure 1. Performance of all methods across 10 random functions r with 1D Context and 1D action. The top plot shows the median regret across contexts and the bottom shows the maximum. Error regions are standard errors.

able to in theory and in controlled practical settings achieve promising performance, it has a number of issues that prevent its wider application. First, we reflect on the fact that the Borda function is a blunt tool: in order to make sampling tractable we must sample a' uniformly during exploration. Since we have the possibility of choosing both a and a' this seems somewhat wasteful of the a' samples. In the context of RLHF with language models this is especially grim as uniformly chosen sequences are likely to be gibberish and the obvious preference for plausible sequences vs uniform ones will mean that the data selected via this strategy would be less likely to be useful.

Many of these applications also benefit from the parallel nature of modern hardware and only make sense when presented with large batches of data. The sequential nature of Borda-AE makes it unsuitable for these applications as well.

With all this in mind, however, we still believe that there is tremendous potential in finding methods for solving these problems, especially with the worst-case style guarantees such as the ones provided here. We hope to continue the progress as we work towards solving these issues.
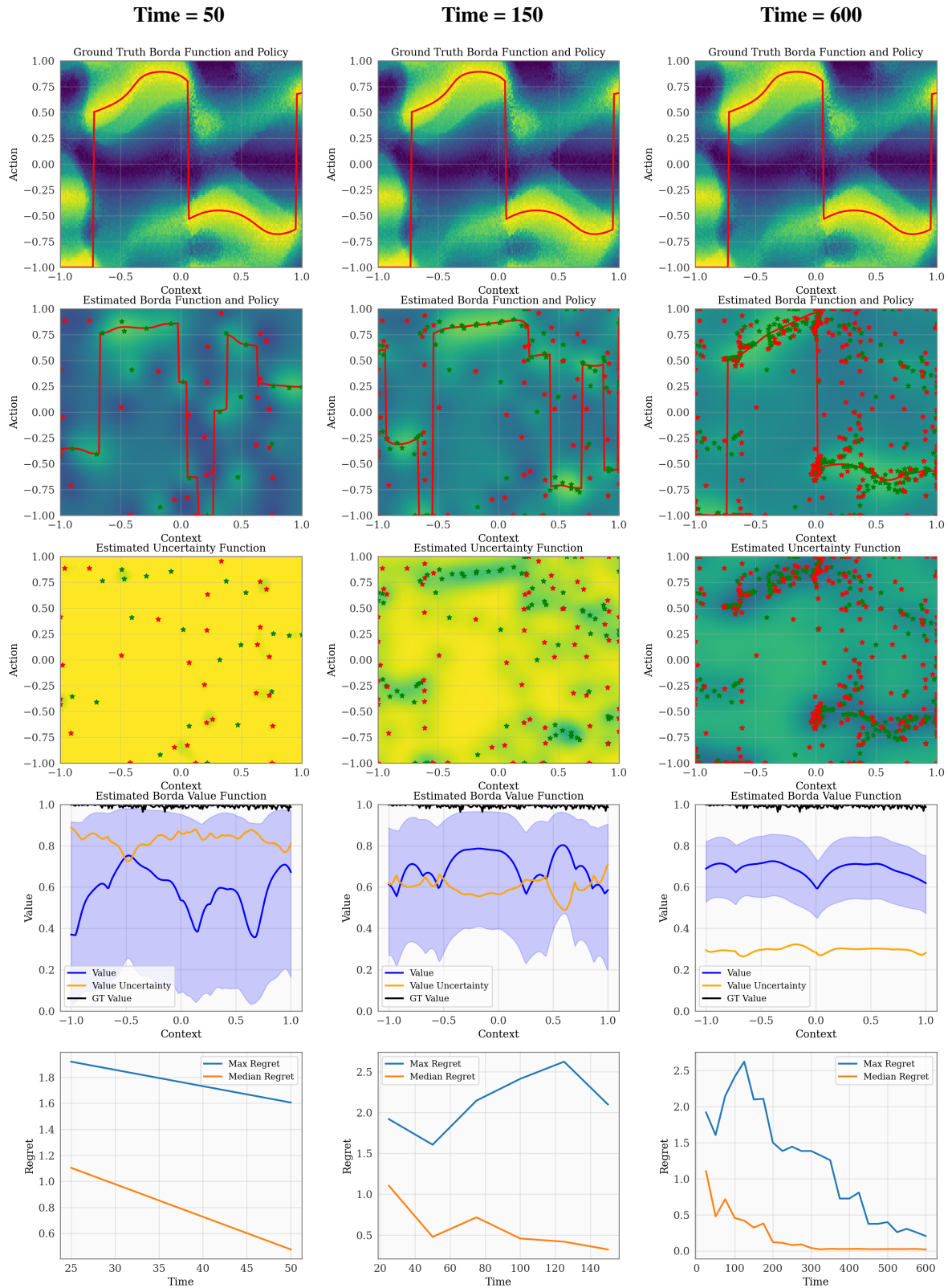
Figure 2. Progress of Borda-AE across 50, 150, and 600 datapoints. From the top the charts show the ground truth function, the mean of the posterior estimate of f_r, the uncertainty function, the estimate of the value function as well as the acquisition function given in (3), and the regret.

References

Akour, R. Robust preference learning-based reinforcement learning. 2014.

Bengs, V., Busa-Fekete, R., Mesaoudi-Paul, A. E., and Hüllermeier, E. Preference-based online learning with dueling bandits: A survey, 2021.

Bradley, R. A. and Terry, M. E. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

Char, I., Chung, Y., Neiswanger, W., Kandasamy, K., Nelson, A. O., Boyer, M., Kolemen, E., and Schneider, J. Offline contextual bayesian optimization. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/7876acb66640bad41f1e1371ef30c180-Paper.pdf.

Cho, W. S., Zhang, P., Zhang, Y., Li, X., Galley, M., Brockett, C., Wang, M., and Gao, J. Towards coherent and cohesive long-form text generation. 2018.

Chowdhury, S. R. and Gopalan, A. On kernelized multi-armed bandits. In *International Conference on Machine Learning*, pp. 844–853. PMLR, 2017.

Christiano, P., Leike, J., Brown, T. B., Martic, M., Legg, S., and Amodei, D. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, 2017. URL https://papers.nips.cc/paper_files/paper/2017/hash/d5e2c0adad503c91f91df240d0cd4e49-Abstract.html.

Dudík, M., Hofmann, K., Schapire, R. E., Slivkins, A., and Zoghi, M. Contextual dueling bandits, 2015.

Fürnkranz, J., Hüllermeier, E., Cheng, W., and Park, S.-H. Preference-based reinforcement learning: a formal framework and a policy iteration algorithm. *Mach Learn*, 2012.

Gardner, J. R., Pleiss, G., Bindel, D., Weinberger, K. Q., and Wilson, A. G. Gpytorch: Blackbox matrix-matrix gaussian process inference with gpu acceleration. In *Advances in Neural Information Processing Systems*, 2018.

Kandasamy, K., Dasarathy, G., Oliva, J., Schneider, J., and Poczos, B. Multi-fidelity gaussian process bandit optimisation. *Journal of Artificial Intelligence Research*, 66:151–196, 2019.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Kreutzer, J., Khadivi, S., Matusov, E., and Riezler, S. Can neural machine translation be improved with user feedback? 2018.

Lawrence, C. and Riezler, S. Improving a neural semantic parser by counterfactual learning from human bandit feedback. 2018.

Li, X., Mehta, V., Kirschner, J., Char, I., Neiswanger, W., Schneider, J., Krause, A., and Bogunovic, I. Near-optimal policy identification in active reinforcement learning. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=3OR2tbtnYC-.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.

Perez, E., Karamcheti, S., Fergus, R., Weston, J., Kiela, D., and Cho, K. Finding generalizable evidence by learning to convince q&a models. 2019.

Rahimi, A. and Recht, B. Random features for large-scale kernel machines. In Platt, J., Koller, D., Singer, Y., and Roweis, S. (eds.), *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007. URL https://proceedings.neurips.cc/paper_files/paper/2007/file/013a006f03dbc5392effeb8f18fda755-Paper.pdf.

Rasmussen, C. E., Williams, C. K., et al. *Gaussian processes for machine learning*, volume 1. Springer, 2006.

Soare, M., Lazaric, A., and Munos, R. Best-arm identification in linear bandits. *Advances in Neural Information Processing Systems*, 27, 2014.

Srinivas, N., Krause, A., Kakade, S. M., and Seeger, M. Gaussian process optimization in the bandit setting: No regret and experimental design. *International Conference on Machine Learning*, 2010.

Stiennon, N., Ouyang, L., Wu, J., Ziegler, D. M., Lowe, R., Voss, C., Radford, A., Amodei, D., and Christiano, P. Learning to summarize from human feedback. 2020.

Thurstone, L. L. The method of paired comparisons for social values. *The Journal of Abnormal and Social Psychology*, 21(4):384, 1927.

Xu, Y., Joshi, A., Singh, A., and Dubrawski, A. Zeroth order non-convex optimization with dueling-choice bandits. In *Conference on Uncertainty in Artificial Intelligence*, pp. 899–908. PMLR, 2020.

Yue, Y., Broder, J., Kleinberg, R., and Joachims, T. The k-armed dueling bandits problem. 2012.

Zhu, B., Jiao, J., and Jordan, M. I. Principled reinforcement learning with human feedback from pairwise or k-wise comparisons, 2023.

Ziegler, D. M., Stiennon, N., Wu, J., Brown, T. B., Radford, A., Amodei, D., Christiano, P., and Irving, G. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

A. Proof of Theorem 4.1

In this section we will prove our main Theorem, 4.1. The overall strategy of the proof is to use our Lipschitz assumption on the link function (more precisely, the relative Lipschitzness of the reward r and the Borda function f_r) in order to go to the Borda function, which we can directly model from data. Then, we use our selection criteria as well as confidence bounds taken from Chowdhury & Gopalan (2017) and convergence rates taken from Kandasamy et al. (2019) in order to complete the argument. We give these cited results as lemmas in what follows.

In order to attain a particular policy performance with probability $1 - \delta$, we must bound the error of the estimates given by our KRR process for a particular confidence level. In order to do so, we adapt the result from Chowdhury & Gopalan (2017), Theorem 2.

Lemma A.1. *Let $\beta_t^{(r)} = 2||f_r||_\kappa + \sqrt{2(\Phi_{t-1}(\mathcal{X}) + 1 + \log(2/\delta))}$. Then with probability $1 - \delta/2$ we have for all time t and any point $x \in \mathcal{X}$,*

$$|\mu_{t-1}(x) - f_r(x)| \leq \beta_t^{(r)} \sigma_{t-1}(x).$$

This lemma jointly bounds the modeling error over the Borda function for all time t though it introduces a dependence on the RKHS norm of f_r. This dependence is inherited from prior work, but we empirically study the relationship between the RKHS norm of a particular reward function and that of the associated Borda function in Section B.

We also adapt a result from Lemma 8 of Kandasamy et al. (2019) in order to understand the convergence of our uncertainty function σ_t.

Lemma A.2. *Suppose we have n queries $(q_t)_{t=1}^n$ taken from $\mathcal{X} \times \mathcal{A}$. Then the posterior σ_t satisfies*

$$\sum_{q_t} \sigma_{t-1}^2(q_t) \leq \frac{2}{\log(1 + \eta^{-2})} \Phi_n(\mathcal{X} \times \mathcal{A}).$$

Lemma A.2 gives us a handle on how quickly we can expect the uncertainty function to shrink as additional datapoints are observed.

Now that we have lemmas A.1 and A.2 in place, we can proceed to the proof of the main result.

Proof. In this proof, we condition on the event in Lemma A.1 holding true. Given that occurence, we can say the following

for every $x \in \mathcal{X}$.

$$\max_{a \in \mathcal{A}} r(x,a) - r(x, \hat{\pi}_T(s)) \overset{\text{Assumption 3.2}}{\leq} L_1 \left(\max_{a \in \mathcal{A}} f_r(x,a) - f_r(x, \hat{\pi}_T(x)) \right) \tag{9}$$

$$\overset{\text{Lemma A.1}}{\leq} L_1 \left(\max_{a \in \mathcal{A}} f_r(x,a) - \max_{t \in [T]} \underline{f_r^t}(x, \hat{\pi}_T(x)) \right) \tag{10}$$

$$\overset{\text{Def. of } \hat{\pi}_T}{=} L_1 \left(\max_{a \in \mathcal{A}} f_r(x,a) - \max_{a \in \mathcal{A}} \max_{t \in [T]} \underline{f_r^t}(x,a) \right) \tag{11}$$

$$= L_1 \min_{t \in [T]} \left(\max_{a \in \mathcal{A}} f_r(x,a) - \max_{a \in \mathcal{A}} \underline{f_r^t}(x,a) \right) \tag{12}$$

$$\overset{\text{Lemma A.1}}{\leq} L_1 \min_{t \in [T]} \left(\max_{a \in \mathcal{A}} \overline{f_r^t}(x,a) - \max_{a \in \mathcal{A}} \underline{f_r^t}(x,a) \right) \tag{13}$$

$$\overset{\text{Def. of } x^t}{\leq} L_1 \min_{t \in [T]} \left(\max_{a \in \mathcal{A}} \overline{f_r^t}(x^t,a) - \max_{a \in \mathcal{A}} \underline{f_r^t}(x^t,a) \right) \tag{14}$$

$$\overset{\text{Def. of } a^t}{\leq} L_1 \min_{t \in [T]} \left(\overline{f_r^t}(x^t,a^t) - \underline{f_r^t}(x^t,a^t) \right) \tag{15}$$

$$\leq \frac{L_1}{T} \sum_{t=1}^{T} \left(\overline{f_r^t}(x^t,a^t) - \underline{f_r^t}(x^t,a^t) \right) \tag{16}$$

$$= \frac{L_1}{T} \sum_{t=1}^{T} 2\beta_t^{(r)} \sigma_t(x^t,a^t) \tag{17}$$

$$\overset{\beta_t^{(r)} \text{ is increasing}}{\leq} \frac{2L_1 \beta_T^{(r)}}{T} \sqrt{\left(\sum_{t=1}^{T} \sigma_t(x^t,a^t) \right)^2} \tag{18}$$

$$\overset{\text{Cauchy-Schwarz}}{\leq} \frac{2L_1 \beta_T^{(r)}}{T} \sqrt{T \sum_{t=1}^{T} \sigma_t^2(x^t,a^t)} \tag{19}$$

$$\overset{\text{Lemma A.2}}{\leq} \frac{2L_1 \beta_T^{(r)}}{\sqrt{T}} \sqrt{C_1 \Phi_T} \tag{20}$$

$$\overset{\text{def of } \beta_T^{(r)}}{=} \frac{2L_1}{\sqrt{T}} (2B + \sqrt{2(\Phi_{t-1} + 1 + \log(2/\delta))}) \sqrt{C_1 \Phi_T} \tag{21}$$

$$= O\left(\frac{L_1}{\sqrt{T}} \left(B + \Phi_T \sqrt{\log \frac{1}{\delta}} \right) \right). \tag{22}$$

$\square$

B. RKHS norms of r and f_r

In order to understand the dependence of our estimation bound on the RKHS norm $||f_r||_\kappa$, we ran numerical experiments on sampled reward functions. For a variety of context and action dimensions, we sampled 1000 reward functions as in Section 5 and numerically approximated their RKHS norms. We also made a Monte-Carlo estimate of the Borda function f_r for each of the reward functions sampled and numerically approximated its RKHS norm. To do this, we uniformly sample 1,000 points x_i from the input space, compute the regularized kernel matrix K for this set x_i, solve the KRR problem $K\alpha = f(x)$ for α. Then we compute the quadratic form $\sqrt{\alpha^T K \alpha}$ as an estimate of the RKHS norm.

In Table 1, we present the results of comparing the RKHS norms of 1000 reward functions and their associated Borda functions sampled as in Section 5. A 'win' was counted when the Borda function had smaller RKHS norm and a 'loss' otherwise. The win margin is the average difference in RKHS norms of the reward and Borda functions, with a positive value when the Borda function was of smaller norm. It is clear here that in general (though not always) the RKHS norm of

Context Dimension	Action Dimension	Win Rate	Win Margin
0	1	0.16	-6.3
1	1	0.89	5.1
1	3	1	21.4
3	1	1	21.5
3	3	1	38.7
10	10	1	19.6

Table 1. Comparison of RKHS norms of reward functions and associated Borda functions

the Borda function f_r for a particular reward function r is smaller than the RKHS norm of the reward function r itself. This relationship seems to grow stronger as the input dimensionality of the reward function grows larger.