Thank you for the reviews of our paper. We appreciate that you like the simplicity of our approach and see its potential impact on the bandit community. We will revise the paper accordingly. Our rebuttal is below.

Reviewer 1

The goal of our work was easy reproducibility and clearly showing the benefits of learning to explore over the state of the art. Therefore, we focus on non-contextual bandits, where the optimal policy (Gittins index) can be sometimes computed and Thompson sampling (TS) is the state of the art. We discuss a contextual extension in Section 8.

Your main concern seems to be how the performance of `GradBand` depends on horizons n and batch sizes m. We observe empirically that doubling of n requires doubling of m, to get policies of a similar quality. The run time of `GradBand` is linear in n and m, and this currently limits what we can do. To show the robustness of our reported results, we decrease batch sizes up to $m = 100$ and increase horizons up to 5 fold.

New horizon Batch size m	n 100	n 200	n 500	n 1000	$2n$ 1000	$5n$ 1000
2 Bernoulli arms, $n = 200$ (Figure 2b)	4.85 ± 0.23	4.86 ± 0.23	4.75 ± 0.08	4.75 ± 0.05	5.93 ± 0.14	6.68 ± 0.18
10 Bernoulli arms, $n = 1000$ (Figure 2c)	27.36 ± 1.35	23.65 ± 0.96	23.29 ± 0.61	24.88 ± 0.76	30.97 ± 1.66	39.22 ± 2.75
10 beta arms, $n = 1000$ (Figure 2c)	15.75 ± 2.86	14.38 ± 1.99	10.68 ± 0.35	10.64 ± 0.25	14.05 ± 0.59	18.36 ± 1.02

The above results are for `SoftElim` and all problems in Figure 2. We observe that the regret increases as m decreases, since the gradients are more noisy. But even at $m = 100$, our policies outperform TS (Figure 2) and are computed 10 times faster than in the paper. The policies for longer horizons also perform well and outperform TS.

Feedback 1: See above.

Feedback 2: Theorem 4 is an instance-dependent upper bound on the n-round regret of `SoftElim`. It is proved for $\theta = 8$, which was obtained by tuning constants. An analogous bound, with worse constants, holds for any $\theta \in (1, 8]$. This can be seen in the proof in Appendix C, which only requires that $\gamma = 1/\theta \in [1/8, 1)$.

Feedback 3: Existing variance minimizing techniques are hard to apply to our problem because 1) our state space, the space of all histories, is at least exponential in n; and 2) the shape of the value function, the future regret as a function of history, is unknown and likely hard to approximate. The baseline b^{SELF} is an independent run of bandit policy θ on the same rewards. When the policy is conservative and over-explores, two of its independent runs are likely to have similar cumulative rewards, and thus b^{SELF} is a good baseline. This is how we choose the initial θ in `GradBand`.

Feedback 4: Conditioned on history H_{t-1}, $S_{i,t}$ is a constant independent of θ. Thus the proof is correct.

Reviewer 2

The average case is not always limiting. For instance, a standard objective in recommender systems is to personalize well on average over users. When each user is viewed as a bandit and $\mathcal{P}$ is a distribution over them, we get our setting.

Reviewer 3

We believe that the reviewer misunderstood our approach. We have two learning algorithms: the bandit policy (agent) in (1), which adapts to an unknown problem instance $P \sim \mathcal{P}$ over n rounds; and a meta-learner `GradBand`, which optimizes the agent by gradient ascent in L iterations. The agent in (1) is a standard bandit policy, which a function of its history H_{t-1} and parameters θ, and does not use rewards of non-pulled arms. In each iteration, `GradBand` runs the agent m times. In each run $j \in [m]$, the agent is executed on rewards $Y^j \in [0, 1]^{K \times n}$ sampled by `GradBand`, for all K arms in n rounds in bandit instance $P^j \sim \mathcal{P}$. The ability to sample Y^j is a weaker assumption than knowing the prior $\mathcal{P}$, as in Thompson sampling. In that case, the meta-learner could sample bandit instance $P^j \sim \mathcal{P}$ and then generate all its rewards over n rounds. The priors are common in practice and can be learned from historic data.

Weaknesses 1 and 5: We optimize θ in a class of bandit policies parameterized by θ. In Sections 6.1 and 6.2, $\mathcal{P}$ is a distribution over two symmetric bandit instances. A single instance would be trivial, since then the optimal solution would be pulling a single arm, irrespective of the history. In Section 6.3, $\mathcal{P}$ is a distribution over bandit instances whose means are drawn independently from a beta prior. That is, there are uncountably many instances.

Weakness 2: We assume independence of rewards over round $t \in [n]$, as in stochastic bandits.

Weakness 3: See the first paragraph.

Weakness 4: `GradBand` is an offline algorithm that optimizes the Bayes reward, which a function of θ. It does not have regret. Does it have any guarantee on optimizing θ? In simple policy classes (Theorem 1), where the Bayes reward is concave in θ, `GradBand` has the same guarantees as gradient ascent and converges to θ_*. In general, the Bayes reward is non-concave in θ and only good empirical performance can be established. The regret in experiments is measured on m sampled bandit instances that are independent of those used in optimization by `GradBand`. So no cheating.