Homework 1: bandits with IID rewards

- All problems can be solved by a fairly basic application of concepts covered in class. Problem (2b) requires a somewhat careful argument (but see the hint).
- Please feel free to refer to facts proved in lecture notes. It would probably be useful to state the “clean event” once and use it for several problems.
- OK to discuss solutions with others, but write your own solutions separately.

Notation. We will use some notation from the class. $T$ is the time horizon, $K$ is the number of arms, $a_t$ is the arm chosen at time $t$, $\mu^*$ is the expected reward of the best arm. For each arm $a$, $\mu(a)$ is the expected reward, and $\Delta(a) = \mu^* - \mu(a)$ is the “badness”.

Problem 1: rewards from a small interval. Consider a version of the problem in which all the realized rewards are in the interval $[\frac{1}{2}, \frac{1}{2} + \epsilon]$ for some $\epsilon \in (0, \frac{1}{2})$. Define versions of UCB1 and Successive Elimination attain improved regret bounds (both logarithmic and root-$T$) that depend on the $\epsilon$.

Hint: Use a more efficient version of Hoeffding Inequality in the slides from the first lecture. It is OK not to repeat all steps from the analysis in the class as long as you explain which steps in the analysis are changed.

Comments after due date: The confidence radius and the $\sqrt{T}$ regret bound can be improved by the factor $\epsilon$. The $\log(T)/\Delta$ regret bound can be improved by the factor of $\epsilon^2$. (The $\epsilon^2$ improvement is a little surprising; but $\Delta \leq \epsilon$, so, in some sense, one factor of $\epsilon$ cancels out.)

An alternative solution is to transform all rewards from $r$ to $(r - \frac{1}{2})/\epsilon$. Then we obtain the problem with $[0, 1]$ rewards, and regret in the original problem is $\epsilon$ times regret in the transformed problem. (And $\Delta$ in the original problem is $\epsilon$ times the $\Delta$ in the transformed problem, hence the $\epsilon^2$ improvement in the $\log(T)$ regret bound.)

Problem 2: instantaneous regret. Recall: instantaneous regret at time $t$ is defined as $\Delta(a_t)$.

(a) Prove that Successive Elimination achieves “instance-independent” regret bound of the form

$$\mathbb{E}[\Delta(a_t)] \leq \frac{\text{polylog}(T)}{\sqrt{t/K}} \text{ for each round } t \in [T].$$  \hspace{1cm} (1)

Comments after due date: A solution should mention which “clean event” is being used, and that in each round $t$ each remaining arm is played at least $t/K$ times.

(b) Let us argue that UCB1 does not achieve the regret bound in (1). More precisely, let us consider a version of UCB1 with $UCB_t(a) = \bar{\mu}_t(a) + 2 \cdot r_t(a)$, where $\bar{\mu}_t(a)$ and $r_t(a)$ are as defined in class. (It is easy to see that the analysis from the class carries over to this version.) Focus on two arms, and prove that this algorithm cannot achieve a regret bound of the form

$$\mathbb{E}[\Delta(a_t)] \leq \frac{\text{polylog}(T)}{t^\gamma}, \gamma > 0 \text{ for each round } t \in [T].$$  \hspace{1cm} (2)
Hint: Fix reward function $\mu$. Focus on the clean event. If (2) holds, then the bad arm cannot be played after some time $T_0$. Consider the last time the bad arm is played, call it $t_0 \leq T_0$. Derive a lower bound on the UCB of the best arm at $t_0$ (stronger lower bound than the one proved in class). Consider what this lower implies for the UCB of the bad arm at time $t_0$. Observe that eventually, after some number of plays of the best arm, the bad arm will be chosen again, assuming a large enough time horizon $T$. Derive a contradiction with (2).

Comments after due date: The take-away is that for “bandits with predictions”, the simple solution of predicting the last-played arm to be the best arm does not always work, even for a good algorithm such as UCB1. This is a tricky problem, even with the hint.

If (2) holds, then the bad arm cannot be played after some time $T_0$ which depends only on the reward function $\mu$, but not on the time horizon. The remainder of the proof shows that the bad arm will be played once more, arriving at a contradiction. This argument works as long as $T_0$ is smaller than a constant fraction of $\log T$ (which holds for a sufficiently large $T$).

We know from the analysis of UCB1 that the bad arm cannot be played more than $O(\log(T)/\Delta^2)$ times. So the "bad arm will be played once more" argument cannot be used indefinitely. But there is no contradiction, because this argument works only if $T_0$ is small enough.

(c) Derive a regret bound for Explore-first with $N$ steps of exploration, namely: an “instance-independent” upper bound on the instantaneous regret. (There are two cases: $t \leq N$ and $t > N$, the first case being trivial.)

Comments after due date: There was a confusion on whether “$N$ steps of exploration” means that each arm was played $N$ times, or each arm was played $N/K$ times. I counted both versions as correct.

Problem 3: bandits with predictions. In “bandits with predictions”, after $T$ rounds the algorithm outputs a prediction: a guess $y_T$ for the best arm. We are mainly interested in the instantaneous regret $\Delta(y_T)$ for the prediction.

(a) Take any bandit algorithm with an instance-independent regret bound $E[R(T)] \leq f(T)$, and construct an algorithm for “bandits with predictions” such that $E[\Delta(y_T)] \leq f(T)/T$.

Comments after due date: Taking $y_T = a_t$ does not work in general, see problem 2(b). There is a simple and easy-to-analyze solution: pick arm $a_t$ with $t$ chosen uniformly at random.

Another solution was proposed: pick the arm with maximal average reward. I am not sure it works in general. More precisely: I don’t know how to complete the proof, for the reason that I explained in class.

Take-away: we easily obtain $E[\Delta(y_T)] = O(\sqrt{K \log(T)/T}$ from standard algorithms such as UCB1 and Successive Elimination. However, as parts (bc) show, one can do much better!

(b) Consider Successive Elimination with $y_T = a_T$. Prove that (with a slightly modified definition of the confidence radius) this algorithm can achieve

$$E[\Delta(y_T)] \leq T^{-\gamma} \text{ if } T > T_{\mu,\gamma},$$

where $T_{\mu,\gamma}$ depends only on the mean rewards $\mu(a) : a \in A$ and the $\gamma$. This holds for an arbitrarily large constant $\gamma$, with only a multiplicative-constant increase in regret.
**Hint:** Put the $\gamma$ inside the confidence radius, so as to make the “failure probability” sufficiently low.

**Comments after due date:** State the clean event, argue that for $T$ large enough, only the best arm remains.

(c) Prove that alternating the arms (and predicting the best one) achieves, for any fixed $\gamma < 1$:

$$\mathbb{E}[\Delta(y_T)] \leq e^{-\Omega(T^\gamma)}$$

if $T > T_{\mu, \gamma}$,

where $T_{\mu, \gamma}$ depends only on the mean rewards $\mu(a) : a \in A$ and the $\gamma$.

**Hint:** Consider Hoeffding Inequality with an arbitrary constant $\alpha$ in the confidence radius. Pick $\alpha$ as a function of the time horizon $T$ so that the failure probability is as small as needed.

**Comments after due date:** Let $\Delta = \min_a \Delta(a)$. Pick $T$ large enough so that the confidence radius in Hoeffding Inequality becomes smaller than (say) $\Delta/2$. Define clean event just for this $T$ (in particular, no need to take a union bound over all rounds $t < T$).

In fact, the result holds even for $\gamma = 1$. However, for $\gamma < 1$ one can prove the same result not only in expectation for each $T$, but w.h.p. for each large enough round $t \in [T]$.

**Problem 4:** doubling trick. Take any bandit algorithm $A$ for fixed time horizon $T$. Convert it to an algorithm $A_\infty$ which runs forever, in phases $i = 1, 2, 3, \ldots$ of $2^i$ rounds each. In each phase $i$ algorithm $A$ is restarted and run with time horizon $2^i$.

**Comments after due date:** So the modified algorithm $A_\infty$ does not have a time horizon, but allows regret bounds for any given round $t$ (i.e., as if the $t$ were the time horizon).

(a) State and prove a theorem which converts an instance-independent upper bound on regret for $A$ into similar bound for $A_\infty$ (so that this theorem applies to both UCB1 and Explore-first).

**Comments after due date:** So if $A$ has a regret bound $O(T^\gamma \log T)$, for any given $\gamma \in (0, 1)$, then one can prove the same regret bound for $A_\infty$.

(b) Do the same for $\log(T)$ instance-dependent upper bounds on regret.

**Note:** in this case, regret increases by a $\log(T)$ factor.

**Note:** consider a regret bound of the form $C \cdot f(T)$, where $f(\cdot)$ does not depend on the reward function $\mu$ and $C$ does not depend on $T$. Such regret bound is called instance-independent if $C$ does not depend on $\mu$, and instance-dependent otherwise.

**Problem 5:** lower bound for non-adaptive exploration. Consider a algorithm such that:

- in the first $N$ rounds (“exploration phase”) the choice of arms does not depend on the observed rewards, for some fixed $N$;
- in all remaining rounds (“exploitation phase”) the algorithm only uses observed rewards from the exploration phase.

Prove that any such algorithm for two arms must have regret $\mathbb{E}[R(T)] \geq \Omega(T^{2/3})$ in the worst case.

**Note:** In particular, regret bound for Explore-First cannot be improved.

**Hint:** Assume a deterministic algorithm and use the “bandits with predictions” impossibility result from class (Lemma 1.1 in Lecture note 4).
Comments after due date: Regret is a sum of regret from exploration, and regret from exploitation. To lower-bound the two summands, we can use different problem instances (because $N$ is fixed ahead of time, and therefore cannot be adjusted to a particular problem instance). So, for “regret from exploration”, we can use two instances: $(\mu_1, \mu_2) = (1, 0)$ and $(\mu_1, \mu_2) = (0, 1)$ (i.e., one arm is very good and another arm is very bad). For “regret from exploitation” we invoke Lemma 1.1: assume the specific randomized problem instance, as in Lemma 1.1, and pick $\epsilon$ such that $cK/\epsilon^2 = N$ to make sure that this lemma applies.

Take-away: allowing Explore-first to pick different arms in exploitation does not help.