
DEEP REINFORCEMENT LEARNING

HOMWORK 5 | MULTI-ARMED BANDITS

Professor Mohammad Hossein Rohban

Sharif University of Technology
Spring 2026



Contents

1	The ETC Algorithm with Heterogeneous Noise	1
2	How to Model with Bandits?	2
3	Reduction	4
4	One-Armed Bandits	4
5	Mirror Descent and FTRL	5
5.1	Legendre Regularizers	5
5.2	Follow-the-Leader, FTRL, and Mirror Descent	6
5.3	FTRL for Adversarial Bandits	7
6	Contextual Bandits for Text Moderation	9

Important Notes

- **Grading Policy:** This homework is considered complete once you reach a total of **100 points**. Up to *5 additional bonus points* may be earned; however, any score above 100 will still be recorded as 100. Bonus points do *not* transfer or overflow to other assignments.
- **Homework Deadline:** The assignment is due on the posted deadline. Please make sure to submit your work on time.
- **Delay Usage Policy:** You may use *at most 5 days* of your allowed late-day budget for this homework. Late days will be automatically applied if you submit after the deadline.
- **Cheating and Collaboration Policy:** You must write all solutions in your own words and produce all code independently. Discussing general ideas is allowed, but sharing written solutions, code, or using someone else's work (including from online sources, previous semesters, or AI tools without proper authorization) is strictly prohibited. Violations will result in academic misconduct procedures.

Advice

Please avoid asking LLMs for the answer. Sure, doing so might give you an immediate reward of 5 points, but in the long run, it won't help you become the next Rockstar of DRL. Remember the key lesson of this course: chasing immediate rewards is rarely the best strategy. Instead, struggle with the problems, read the papers, and truly understand the algorithms. The satisfaction of solving a challenging problem yourself is far greater than any grade. Good luck! [1]

Grading Breakdown

Task / Criteria	Points
The ETC Algorithm with Heterogeneous Noise	10 pts
Pure Exploration Phase	2 pts
Arm Selection Probability	2 pts
Combining The Results	2 pts
Optimal Exploration Allocation	2 pts
Optimal Number of Exploration Rounds	2 pts
How to Model with Bandits?	10 pts
Adversarial Bandits	2 pts
The Adversary	1 pts
The Learner	1 pts
The Objective	2 pts
Non-Stationary Bandits	2 pts
Contextual Bandits	2 pts
Reduction	10 pts
Reduction	5 pts
The Confidence Ellipsoid	5 pts
One-Armed Bandits	15 pts
Mirror Descent and FTRL	30 pts
Legendre Functions	3 pts
Failure of Follow-the-Leader	3 pts
Adding Quadratic Regularization	3 pts
Mirror Descent Comparison	3 pts
The FTRL Distribution	4 pts
Effect of Observing a Loss	2 pts
Hessian of the Potential	2 pts
Diameter of the Regularizer	2 pts
Regret Bound	4 pts
Mirror Descent Instead of FTRL	2 pts
Implementation of FTRL for Adversarial Bandits	2 pts
Contextual Bandits for Text Moderation	25 pts
Notebook Implementation	20 pts
Analysis Questions	5 pts
Bonus	+5 pts
Clarity and Quality of Code	+2 pts
Clarity and Quality of Report	+3 pts
Total	100+5 pts

1 The ETC Algorithm with Heterogeneous Noise

The Explore-Then-Commit (ETC) algorithm has two phases. We start by exploration and after a fixed number of rounds, we stop exploring and commit to the arm with the larger empirical mean.

In the previous setting, all arms had the same sub-Gaussian parameter. We now consider a simple two-armed bandit problem where the two arms have different noise levels. Arm 1 and arm 2 have unknown means μ_1 and μ_2 , respectively. Assume arm 1 is the unique optimal arm and define the suboptimality gap as

$$\Delta = \mu_1 - \mu_2 > 0.$$

The reward distribution of arm 1 is σ_1 -sub-Gaussian, and the reward distribution of arm 2 is σ_2 -sub-Gaussian, where $\sigma_1 \neq \sigma_2$. For $i \in \{1, 2\}$, let $\hat{\mu}_i$ denote the empirical mean of arm i after the exploration phase.

We consider the following modified ETC algorithm:

- **Exploration Phase:** Pull arm 1 exactly m_1 times and arm 2 exactly m_2 times.
- **Commitment Phase:** For the remaining $n - m_1 - m_2$ rounds, commit exclusively to the arm with the larger empirical mean. Ties are broken arbitrarily.

Assume throughout that $m_1 + m_2 \leq n$.

Question 1: Pure Exploration Phase

[2 Points]

During the exploration phase, arm 1 is pulled m_1 times and arm 2 is pulled m_2 times. Determine the expected cumulative pseudo-regret over the exploration phase. Briefly explain why only the pulls of arm 2 contribute to the regret.

Suppose that we are now at the end of the exploration phase. The algorithm commits to arm 2 only if

$$\hat{\mu}_2 \geq \hat{\mu}_1.$$

Question 2: Arm Selection Probability

[2 Points]

Prove the following upper bound on the probability of selecting the suboptimal arm:

$$\mathbb{P}(\hat{\mu}_2 \geq \hat{\mu}_1) \leq \exp \left(- \frac{\Delta^2}{2 \left(\frac{\sigma_1^2}{m_1} + \frac{\sigma_2^2}{m_2} \right)} \right).$$

Briefly discuss how this probability changes as m_1 , m_2 , σ_1 , σ_2 , or Δ changes.

Question 3: Combining The Results

[2 Points]

Combine the results from the last two questions to obtain an upper bound on the expected pseudo-regret R_n of the modified ETC algorithm.

Your bound should be written in terms of n , m_1 , m_2 , Δ , σ_1 , and σ_2 .

Discuss the trade-off between exploration and commitment. What happens if m_1 and m_2 are too small? What happens if they are too large?

Now suppose that the total exploration budget

$$m = m_1 + m_2$$

is fixed. We want to allocate this budget between the two arms in a way that makes the error probability from Question 2 as small as possible.

Question 4: Optimal Exploration Allocation

[2 Points]

Find the optimal ratio

$$\frac{m_1}{m_2}$$

that minimizes

$$\frac{\sigma_1^2}{m_1} + \frac{\sigma_2^2}{m_2}.$$

Then express the optimal values of m_1 and m_2 in terms of m , σ_1 , and σ_2 .

Hint: Since $m_1 + m_2 = m$, you may write $m_2 = m - m_1$ and minimize over m_1 .

Question 5: Optimal Number of Exploration Rounds

[2 Points]

Assume that Δ is known. Use the optimal allocation from Question 4 to choose the total exploration budget m that minimizes the pseudo-regret upper bound from Question 3.

Show that, up to universal constants and lower-order terms, the optimized regret scales as

$$O\left(\frac{(\sigma_1 + \sigma_2)^2}{\Delta} \log n\right).$$

Explain why this result is intuitive: why should the total amount of exploration increase when the noise levels σ_1 and σ_2 are larger, and decrease when the gap Δ is larger?

2 How to Model with Bandits?

The goal of stochastic bandit learning is to maximize cumulative reward, or equivalently to minimize cumulative regret. Let

$$\nu = (P_a : a \in \mathcal{A})$$

be a stochastic bandit, where each arm $a \in \mathcal{A}$ is associated with a reward distribution P_a . Define

$$\mu_a(\nu) = \int_{-\infty}^{+\infty} x dP_a(x), \quad \mu^*(\nu) = \max_{a \in \mathcal{A}} \mu_a(\nu).$$

The regret of a policy π on the bandit instance ν is

$$R_n(\pi, \nu) = n\mu^*(\nu) - \mathbb{E} \left[\sum_{t=1}^n X_t \right],$$

where $X_t \in \mathbb{R}$ is sampled from P_{A_t} and the expectation is taken over the interaction between π and ν .

George E. P. Box famously wrote that “all models are wrong, but some are useful.” In the stochastic bandit model, the reward distribution depends only on the selected action. This is a useful idealization, but in many real problems rewards may also depend on time, context, hidden variables, or strategic behavior.

Question 1: Adversarial Bandits**[2 Points]**

In the adversarial bandit model, the environment may choose the rewards directly. At each round $t \in [n]$, the adversary secretly chooses rewards $x_{t,a}$ for every arm $a \in \mathcal{A}$. The regret is defined as

$$R_n(\pi, x) = \max_{a \in \mathcal{A}} \sum_{t=1}^n x_{t,a} - \mathbb{E} \left[\sum_{t=1}^n x_{t,A_t} \right].$$

Explain how this regret notion differs from stochastic regret. Why is the stochastic regret definition no longer appropriate in this setting?

Question 2: The Adversary**[1 Points]**

Explain the role of the adversary in the regret definition. What happens if, at every round, all arms receive the same reward, for example all rewards are zero or all rewards are maximal?

Question 3: The Learner**[1 Points]**

Can algorithms such as ETC or UCB be used directly in the adversarial setting? Explain how an adversary could exploit a deterministic policy.

Question 4: The Objective**[2 Points]**

What should be considered a successful learning strategy in adversarial bandits? Give an intuitive explanation of how sublinear regret can still be possible, and describe one real-world situation where adversarial modeling is useful.

Question 5: Non-Stationary Bandits**[2 Points]**

Suppose the reward distribution of each arm changes over time. Are adversarial bandits an appropriate model for this form of non-stationarity? Your answer should discuss how the choice of regret benchmark affects the modeling decision.

Question 6: Contextual Bandits**[2 Points]**

Consider a contextual bandit with contexts $c \in \mathcal{C}$, actions $a \in \mathcal{A}$, a feature map $\psi : \mathcal{C} \times \mathcal{A} \rightarrow \mathbb{R}^d$, and an unknown parameter vector $\theta^* \in \mathbb{R}^d$ such that

$$r(c, a) = \langle \theta^*, \psi(c, a) \rangle.$$

In LinUCB/OFUL, one often chooses an action by optimizing over available feature vectors. In the contextual interpretation, however, the observed context c restricts the feasible set to

$$\{\psi(c, a) : a \in \mathcal{A}\}.$$

Does this restriction create a real problem for applying LinUCB-style reasoning? Explain why, or propose a suitable modification.

3 Reduction

Consider the linear bandit with action set

$$\mathcal{A} = \{e_1, e_2, \dots, e_d\},$$

where e_i is the i -th unit vector in $\mathbb{R}^d$.

Question 1: Reduction

[5 Points]

Explain how similar or different LinUCB and UCB are for this d -armed bandit instance. In particular, compare the confidence bounds obtained by the two algorithms.

Question 2: The Confidence Ellipsoid

[5 Points]

In LinUCB, the confidence set $\mathcal{C}_t \subset \mathbb{R}^d$ is refined as more observations are collected. Using the ridge least-squares estimator, one obtains a confidence ellipsoid of the form

$$\mathcal{C}_t \subseteq \mathcal{E}_t = \left\{ \theta \in \mathbb{R}^d : \|\theta - \hat{\theta}_{t-1}\|_{V_{t-1}}^2 \leq \beta_t \right\}.$$

Give a geometric interpretation of this ellipsoid and explain how its volume changes over time. For a positive definite matrix $G \in \mathbb{R}^{d \times d}$, recall that $\|x\|_G^2 = x^\top G x$ is the Mahalanobis norm.

4 One-Armed Bandits

A *one-armed bandit* is a two-armed bandit in which only one arm has an unknown reward distribution. Let

$$\mathcal{M}_1$$

be a set of probability distributions on $(\mathbb{R}, \mathfrak{B}(\mathbb{R}))$ with finite means, and let

$$\mathcal{M}_2 = \{\delta_{\mu_2}\}$$

be the singleton class containing the Dirac distribution at a known value $\mu_2 \in \mathbb{R}$. The set of bandit instances is

$$\mathcal{E} = \mathcal{M}_1 \times \mathcal{M}_2.$$

Thus arm 1 has an unknown reward distribution, while arm 2 always gives the known deterministic reward μ_2 .

At each round t , the learner chooses an action $A_t \in \{1, 2\}$ and observes the corresponding reward. A policy $\pi = (\pi_t)_t$ is called a *retirement policy* if, once it plays arm 2, it never returns to arm 1. Equivalently, for every history for which $A_t = 2$,

$$\pi_{t+1}(2 \mid A_1, X_1, \dots, A_t, X_t) = 1.$$

For a horizon n and bandit instance $\nu \in \mathcal{E}$, let $R_n(\pi, \nu)$ denote the expected cumulative pseudo-regret of policy π .

We also consider the Bernoulli one-armed bandit, where

$$\mathcal{M}_1 = \{\text{Bernoulli}(\mu_1) : \mu_1 \in [0, 1]\}, \quad \mathcal{M}_2 = \{\delta_{\mu_2}\},$$

with known $0 < \mu_2 < 1$. For an infinite-horizon retirement policy $\pi = (\pi_t)_{t=1}^\infty$, define the retirement time

$$\tau = \inf\{t \geq 1 : A_t = 2\},$$

with the convention that $\tau = \infty$ if arm 2 is never played.

Question 1: One-Armed Bandits

[15 Points]

- (a) Let n be fixed, and let $\pi = (\pi_t)_{t=1}^n$ be an arbitrary policy. Prove that there exists a retirement policy $\pi' = (\pi'_t)_{t=1}^n$ such that, for every bandit instance $\nu \in \mathcal{E}$,

$$R_n(\pi', \nu) \leq R_n(\pi, \nu).$$

Your proof should explain why the known deterministic arm can be simulated without actually pulling it, and why all pulls of arm 1 can be moved before the final commitment to arm 2 without increasing regret.

- (b) Let $\pi = (\pi_t)_{t=1}^\infty$ be any retirement policy. Prove that there exists a bandit instance $\nu \in \mathcal{E}$ such that

$$\limsup_{n \rightarrow \infty} \frac{R_n(\pi, \nu)}{n} > 0.$$

Your argument should address both cases $\mu_1 > \mu_2$ and $\mu_1 < \mu_2$. In particular, relate linear regret to the two events

$$\{\tau < \infty\} \quad \text{and} \quad \{\tau = \infty\},$$

and explain why a retirement policy cannot avoid linear regret uniformly over all Bernoulli parameters $\mu_1 \in [0, 1]$.

5 Mirror Descent and FTRL

This problem studies two closely related ideas in online learning: Mirror Descent and Follow-the-Regularized-Leader (FTRL). We first examine Legendre regularizers, then compare Follow-the-Leader with regularized variants, and finally apply FTRL to adversarial bandits with importance-weighted loss estimates.

Throughout this section, the learner plays actions from a convex action set and observes a sequence of linear losses.

5.1 Legendre Regularizers

Legendre functions are useful regularizers because they are strictly convex on the interior of their domain and their gradients become large near the boundary. This behavior is especially useful in constrained online optimization.

Question 1: Legendre Functions

[3 Points]

For each of the following real-valued functions, decide whether it is Legendre on the given domain. For each case, briefly justify your answer by discussing convexity, differentiability on the interior, and the behavior of the gradient near the boundary.

- (a) $f(x) = x^2$ on $[-1, 1]$.
- (b) $f(x) = -\sqrt{x}$ on $[0, \infty)$.
- (c) $f(x) = \log(1/x)$ on $[0, \infty)$, with $f(0) = \infty$.

- (d) $f(x) = x \log x$ on $[0, \infty)$, with $f(0) = 0$.
- (e) $f(x) = |x|$ on $\mathbb{R}$.
- (f) $f(x) = \max\{|x|, x^2\}$ on $\mathbb{R}$.

5.2 Follow-the-Leader, FTRL, and Mirror Descent

Consider a one-dimensional online linear optimization problem with action set $\mathcal{A} = [-1, 1]$. The learner faces the loss sequence $y_1 = \frac{1}{2}$, and for $s > 1$,

$$y_s = \begin{cases} 1, & \text{if } s \text{ is odd,} \\ -1, & \text{if } s \text{ is even.} \end{cases}$$

At each time t , the learner chooses $a_t \in \mathcal{A}$ and suffers loss $\langle a_t, y_t \rangle$.

Question 2: Failure of Follow-the-Leader

[3 Points]

Recall that Follow-the-Leader without regularization chooses

$$a_t = \arg \min_{a \in \mathcal{A}} \sum_{s=1}^{t-1} \langle a, y_s \rangle.$$

Show that Follow-the-Leader suffers linear regret on the sequence above. Briefly explain the intuition behind this failure and why the algorithm keeps choosing bad actions.

Question 3: Adding Quadratic Regularization

[3 Points]

Now consider Follow-the-Regularized-Leader with quadratic regularizer

$$F(a) = a^2.$$

Write the FTRL update explicitly for this problem. That is, compute

$$a_t = \arg \min_{a \in [-1, 1]} \left\{ \eta \sum_{s=1}^{t-1} \langle a, y_s \rangle + F(a) \right\}.$$

Explain how the regularizer changes the behavior of the algorithm compared to ordinary Follow-the-Leader.

Question 4: Mirror Descent Comparison

[3 Points]

Consider Mirror Descent on the same problem with the same quadratic potential

$$F(a) = a^2.$$

State the explicit Mirror Descent update rule. Then compare the Mirror Descent update with the FTRL update from the previous question.

Are the two algorithms exactly the same in this example? If not, explain when and why they differ.

5.3 FTRL for Adversarial Bandits

We now move to the adversarial bandit setting. The learner does not observe the full loss vector. Instead, after choosing an arm, the learner observes only the loss of the selected arm.

Let $(y_t)_{t=1}^n$ be a sequence of loss vectors satisfying

$$y_t \in [0, 1]^k \quad \text{for all } t.$$

At each round t , the learner samples an action

$$A_t \in [k]$$

from a probability distribution $P_t \in \mathcal{P}_{k-1}$, where

$$\mathcal{P}_{k-1} = \left\{ p \in \mathbb{R}^k : p_i \geq 0, \quad \sum_{i=1}^k p_i = 1 \right\}.$$

Since only the loss of the selected arm is observed, we use the importance-weighted estimator

$$\hat{Y}_{ti} = \frac{\mathbf{1}\{A_t = i\} y_{ti}}{P_{ti}}.$$

Consider FTRL with the potential

$$F(p) = -2 \sum_{i=1}^k \sqrt{p_i}.$$

The algorithm chooses

$$P_t = \arg \min_{p \in \mathcal{P}_{k-1}} \left\{ \eta \sum_{s=1}^{t-1} \langle p, \hat{Y}_s \rangle + F(p) \right\},$$

where $\eta > 0$ is the learning rate.

The regret is defined as

$$R_n = \mathbb{E} \left[\sum_{t=1}^n y_{t, A_t} \right] - \min_{i \in [k]} \sum_{t=1}^n y_{ti}.$$

Question 5: The FTRL Distribution

[4 Points]

Let

$$L_{t-1, i} = \sum_{s=1}^{t-1} \hat{Y}_{si}$$

be the cumulative estimated loss of arm i before round t .

Show that the FTRL distribution satisfies

$$P_{ti} = (\lambda_t + \eta L_{t-1,i})^{-2},$$

where $\lambda_t \in \mathbb{R}$ is chosen so that $P_t \in \mathcal{P}_{k-1}$.

Briefly explain why such a value of λ_t exists and is unique.

Question 6: Effect of Observing a Loss

[2 Points]

Show that the probability assigned to the selected arm cannot increase immediately after observing its loss. More precisely, prove that

$$P_{t+1,A_t} \leq P_{t,A_t} \quad \text{for } t = 1, \dots, n-1.$$

Explain the intuition behind this property.

Question 7: Hessian of the Potential

[2 Points]

Compute the Hessian of

$$F(p) = -2 \sum_{i=1}^k \sqrt{p_i}.$$

Show that

$$\nabla^2 F(p) = \frac{1}{2} \text{diag}(p^{-3/2}),$$

where the power is applied coordinate-wise.

Briefly explain what this Hessian says about the curvature of the regularizer near the boundary of the simplex.

Question 8: Diameter of the Regularizer

[2 Points]

Show that the diameter of the simplex with respect to F satisfies

$$\text{diam}_F(\mathcal{P}_{k-1}) \leq 2\sqrt{k}.$$

Here,

$$\text{diam}_F(\mathcal{P}_{k-1}) = \sup_{p,q \in \mathcal{P}_{k-1}} |F(p) - F(q)|.$$

Question 9: Regret Bound

[4 Points]

Using the previous parts, prove that for a suitable choice of learning rate η , the regret of this FTRL algorithm satisfies

$$R_n \leq \sqrt{8kn}.$$

Clearly state the value of η that you use. Briefly explain why the dependence on k and n is reasonable

for adversarial bandits.

Question 10: Mirror Descent Instead of FTRL

[2 Points]

Suppose we use Mirror Descent with the same potential

$$F(p) = -2 \sum_{i=1}^k \sqrt{p_i}$$

instead of FTRL.

Are the resulting algorithms the same? If not, explain the difference between the two update rules. What regret guarantee can be proved for Mirror Descent in this setting?

Question 11: Implementation of FTRL for Adversarial Bandits

[2 Points]

Explain how you would implement the FTRL algorithm described above.

Your answer should include:

- how to compute the distribution P_t at each round;
- how to sample the action A_t ;
- how to construct the importance-weighted estimator $\hat{Y}_t$;
- how to update the cumulative estimated losses.

In particular, describe how to solve the one-dimensional normalization equation

$$\sum_{i=1}^k (\lambda_t + \eta L_{t-1,i})^{-2} = 1.$$

6 Contextual Bandits for Text Moderation

In this section, you will work with a realistic contextual bandit problem inspired by content moderation systems. Unlike the classical multi-armed bandits studied throughout this homework, the learner now receives a context before selecting an action. The objective is to learn a decision-making policy from logged interaction data while carefully balancing reward maximization and safety considerations. The accompanying notebook guides you through the complete pipeline of an offline contextual bandit system.

Implementation 1:

[20 Points]

Complete all notebook cells marked with TODO. Your grade will be based on correctness, completeness, and the quality of your experimental analysis.

Question 1: Analysis Questions**[5 Points]**

Answer the following questions based on your implementations and experimental results.

- (1) **Dataset**
 - (a) What are the limitations of using binary labels as reward ground truth?
 - (b) Why is `tweet_eval/hate` a reasonable but imperfect OOD stress test for `tweet_eval/offensive`?
- (2) **Severity Proxy**
 - (a) Why is a severity proxy dangerous if treated as ground truth?
 - (b) What signals does it overemphasize?
- (3) **Reward, Oracle, Regret, and Cost Sensitivity**
 - (a) Which product profile is most likely to over-block?
 - (b) Why is this a bandit problem rather than a standard classification problem?
 - (c) Why does the oracle action distribution change across profiles?
- (4) **Logged Bandit Data**
 - (a) How might low propensities affect offline policy evaluation?
- (5) **Risk Model for Safety and Calibration**
 - (a) Is it fair to use labels here while the reward model uses bandit feedback?
 - (b) What would change if labels were delayed or missing?
- (6) **Fixed Policies from the Reward Model**
 - (a) Does uncertainty help?
 - (b) Which policy has the highest block rate?
 - (c) Which policy has the worst OOD degradation?
- (7) **Safe Exploration Constraints**
 - (a) What is an ethically acceptable exploration space for content moderation?
- (8) **Online LinUCB**
 - (a) Why can LinUCB be more stable than neural Thompson sampling during the early stages of learning?
- (9) **Offline Policy Evaluation**
 - (a) Which estimator is most variable?
 - (b) Which estimator is most biased when the reward model is poor?
 - (c) What happens when target actions have very low behavior propensities?
- (10) **Calibration**
 - (a) How does miscalibration affect safe exploration?

References

[1] Based on INF8250AE - Introduction to Reinforcement Learning. Fall 2025.