

Homework 5

● Graded

Student

John Stephen Gutam

Total Points

48 / 50 pts

Question 1

EM

25 / 25 pts

✓ + 25 pts Correct

+ 20 pts Most of the way there. Some minor details missing.

+ 15 pts About halfway there. Some effort made.

+ 7 pts Solution started

+ 0 pts Incorrect/Missing solution.

Question 2

Bandits

■ 23 / 25 pts

✓ + 25 pts Correct

+ 15 pts Implementation

+ 15 pts Writeup

💬 - 2 pts Your graph for greedy is off but looks good otherwise.

Question assigned to the following page: [1](#)

Name: John Stephen Gutam
Email: jgutam@gmu.edu

Problem 1:

G01413212

Homework - 5

CS688 - Machine Learning

Name: John Stephen Gutam
Email: jgutam@gmu.edu

1. (a) For T time steps, both the coin that I flip ($z_t \in \{1, 2\}$ at time step t) and the outcome variable x_t which is 1 if the coin comes up heads and 0 otherwise.

To write the log-likelihood function, we need to consider the probability for parameters $\theta = (p_1, p_2, \pi_1)$

The log likelihood function can be expressed as:

$$\log L(\theta) = \sum_t \left[I(z_t=1) \log(\pi_1 \cdot p_1^{x_t} \cdot (1-p_1)^{(1-x_t)}) + I(z_t=2) \log((1-\pi_1) \cdot p_2^{x_t} \cdot (1-p_2)^{(1-x_t)}) \right]$$

To find the maximum likelihood estimators (MLEs) of $\theta = (p_1, p_2, \pi_1)$, we take the partial derivative with respect to each parameter and set them equal to zero.

For p_1

$$\frac{\partial \log L(\theta)}{\partial p_1} = 0 \quad \rightarrow \text{eq (1)}$$

$$\sum_t \left[I(z_t=1) \frac{\partial}{\partial p_1} (\log(\pi_1 \cdot p_1^{x_t} \cdot (1-p_1)^{(1-x_t)})) + 0 \right] = 0$$

Let this derivative be M

solve M

$$M = \frac{\partial}{\partial p_1} (\log \pi_1 \cdot p_1^{x_t} \cdot (1-p_1)^{(1-x_t)}) = \log \pi_1 \frac{\partial}{\partial p_1} [x_t \log p_1 + (1-x_t) \log (1-p_1)]$$

Question assigned to the following page: [1](#)

$$= x_t \cdot \frac{1}{p_1} + (1-x_t) \cdot \left(\frac{1(-1)}{1-p_1} \right)$$

$$M = \frac{x_t}{p_1} - \frac{(1-x_t)(-1)}{1-p_1} \rightarrow \text{eq (2)}$$

substitute eq (2) in eq (1)

$$\sum_t \left[I(z_t=1) \cdot \left(\frac{x_t}{p_1} - \frac{(1-x_t)(-1)}{1-p_1} \right) \right] = 0$$

$$\sum_t \left[I(z_t=1) \cdot \frac{x_t}{p_1} \right] - \sum_t \left[I(z_t=1) \cdot \frac{(1-x_t)}{(1-p_1)} \right] = 0$$

multiply by $p_1 + (1-p_1)$

$$\sum_t \left[I(z_t=1) \cdot x_t \right] - \frac{p_1}{(1-p_1)} \sum_t \left[I(z_t=1) \cdot (1-x_t) \right] = 0$$

$$\frac{p_1}{(1-p_1)} = \frac{\sum_t [I(z_t=1) \cdot x_t]}{\sum_t [I(z_t=1) \cdot (1-x_t)]}$$

$$p_1 = \frac{\sum_t [I(z_t=1) \cdot x_t]}{\sum_t [I(z_t=1)]}$$

similarly for p_2 :

$$p_2 = \frac{\sum_t [I(z_t=2) \cdot x_t]}{\sum_t [I(z_t=2)]}$$

Question assigned to the following page: [1](#)

2

for π_1 :

$$\frac{\partial(\log L(\theta))}{\partial \pi_1} = \sum_t \left[\frac{I(z_t=1)}{\pi_1} - \frac{I(z_t=2)}{(1-\pi_1)} \right] = 0$$

solving

$$\sum_t \left[\frac{I(z_t=1)}{\pi_1} \right] = \sum_t \left[\frac{I(z_t=2)}{(1-\pi_1)} \right]$$

solving for π_1 ,
we get

$$\pi_1 = \frac{\sum_{t=1}^T I(z_t=1)}{T}$$

(b) In this case, since we don't observe which coin was flipped, we need to consider the probability of observing the outcome x_t under either coin, weighted by the probability of selecting each coin.

The likelihood of observing x_t given by:

$$P\left(\frac{x_t}{\theta}\right) = P_1(x_t | z_t=1, \theta) \times P_1(z_t=1 | \theta) + P_1(x_t | z_t=2, \theta) \times P_1(z_t=2 | \theta)$$

from eq (2) and (3)

$$= p_1^{(x_t)} \cdot (1-p_1)^{(1-x_t)} \times \pi_1 + p_2^{(x_t)} \cdot (1-p_2)^{(1-x_t)} \times (1-\pi_1)$$

eq (1)

Question assigned to the following page: [1](#)

where $\theta = (p_1, p_2, \pi_1)$

① $P(x_t | z_t, \theta)$:

↳ if $z_t = 1$, the prob. of observing x_t is p_1 if $x_t = 1$ (heads)
and $1 - p_1$ if $x_t = 0$ (tails).

↳ if $z_t = 2$, the prob. of observing $x_t = 1$ is p_2 if $x_t = 1$ (heads)
and $1 - p_2$ if $x_t = 0$ (tails).

we can express this as:

$$P(x_t | z_t, \theta) = \begin{cases} p_1^{x_t} (1-p_1)^{1-x_t} & \text{if } z_t = 1 \\ p_2^{x_t} (1-p_2)^{1-x_t} & \text{if } z_t = 2 \end{cases} \rightarrow \text{eq (2)}$$

② $P(z_t | \theta)$:

↳ The prob. of choosing coin $z_t = 1$ is π_1 , and $z_t = 2$ is $1 - \pi_1$.

↳ we can express this as:

$$P(z_t | \theta) = \begin{cases} \pi_1 & \text{if } z_t = 1 \\ 1 - \pi_1 & \text{if } z_t = 2 \end{cases} \rightarrow \text{eq (3)}$$

Taking log of the likelihood for the eq (1)

$$\log L(\theta) = \sum_{t=1}^T \log (p_1^{x_t} (1-p_1)^{1-x_t} \pi_1 + p_2^{x_t} (1-p_2)^{1-x_t} (1-\pi_1))$$

Hence proved.

Question assigned to the following page: [1](#)

(3)

(c) Using the above, In the E-step of the EM algorithm, we need to compute the expected value of the complete data log likelihood function, given the observed data and the current parameter estimates.

To do this we need to calculate $P_\lambda(z_t=1|x_t, \theta)$, which is the probability that coin 1 was flipped, given the observed outcome x_t and the current parameter estimates θ .

Using Bayes' rule, we can derive:

$$\begin{aligned} \hookrightarrow P_\lambda(z_t=1|x_t, \theta) &= \frac{P_\lambda(x_t|z_t=1, \theta) * P_\lambda(z_t=1|\theta)}{P_\lambda(x_t|\theta)} \\ &= \frac{P_1^{x_t} * (1-P_1)^{(1-x_t)} * \pi_1}{P_1^{x_t} (1-P_1)^{(1-x_t)} * \pi_1 + P_2^{x_t} (1-P_2)^{(1-x_t)} * (1-\pi_1)} \end{aligned}$$

Similarly

$$P_\lambda(z_t=2|x_t, \theta) = \frac{P_\lambda(x_t|z_t=2, \theta) * P_\lambda(z_t=2|\theta)}{P_\lambda(x_t|\theta)}$$

(8)

$$\hookrightarrow P_\lambda(z_t=2|x_t, \theta) = 1 - P_\lambda(z_t=1|x_t, \theta)$$

Question assigned to the following page: [1](#)

(d) In the M-step, we maximize the expected complete log likelihood function to update p_1, p_2, π_1

~~$Q(\theta)$~~ The expected complete log likelihood function can be written as

$$Q\left(\frac{\theta}{\theta^t}\right) = \sum_{t=1}^T \left[P_1(z_t=1 | x_t, \theta^t) * \log(\pi_1 * p_1^{x_t} * (1-p_1)^{(1-x_t)}) \right. \\ \left. + P_2(z_t=2 | x_t, \theta^t) * \log((1-\pi_1) * p_2^{x_t} * (1-p_2)^{(1-x_t)}) \right]$$

where θ^t represents the current parameter estimates at iteration t of the EM algorithm.

To update the parameters in the M-step, we take the partial derivatives of $Q(\theta | \theta^t)$ with respect to each parameter and set them to zero.

For p_1 :

$$\frac{\partial Q}{\partial p_1} = \sum_t P_1(z_t=1 | x_t, \theta^t) * \left[\left(\frac{x_t}{p_1} \right) - \frac{(1-x_t)}{(1-p_1)} \right] = 0 \\ \sum_t P_1(z_t=1 | x_t, \theta^t) * \left(\frac{x_t}{p_1} \right) = \sum_t P_1(z_t=1 | x_t, \theta^t) * \frac{(1-x_t)}{(1-p_1)} \\ \Rightarrow p_1 = \frac{\sum P_1(z_t=1 | x_t, \theta^t) * x_t}{\sum P_1(z_t=1 | x_t, \theta^t)}$$

Question assigned to the following page: [1](#)

Therefore, the solution for p_1 is

$$p_1^{(t+1)} = \frac{\sum_t p_k(z_t=1|x_t, \theta^t) \cdot x_t}{\sum_t p_k(z_t=1|x_t, \theta^t)}$$

Similarly for p_2

$$\frac{\partial Q}{\partial p_2} = 0 \Rightarrow p_2^{(t+1)} = \frac{\sum_t p_k(z_t=2|x_t, \theta^t) \cdot x_t}{\sum_t p_k(z_t=2|x_t, \theta^t)}$$

For π_1

$$\frac{\partial Q}{\partial \pi_1} = \sum_t \left[\frac{p_k(z_t=1|x_t, \theta^t)}{\pi_1} - \frac{p_k(z_t=2|x_t, \theta^t)}{(1-\pi_1)} \right] = 0$$

solving for π_1 , we get:

$$\pi_1^{(t+1)} = \frac{\sum_t p_k(z_t=1|x_t, \theta^t)}{T}$$

\therefore These updates are based on the expectations computed in the E-step.

Problem 2:

Question assigned to the following page: [2](#)

```

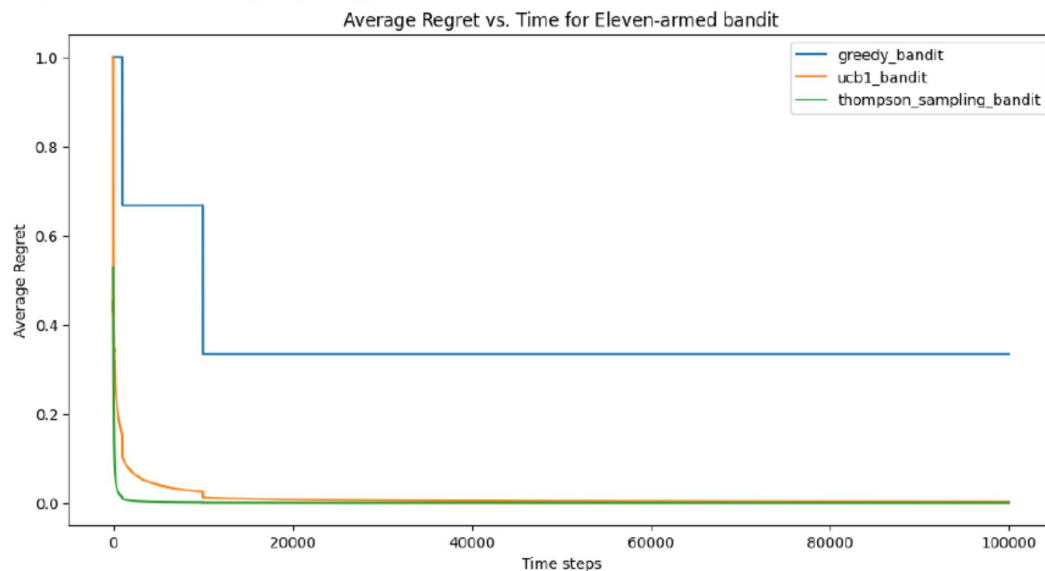
Testing on Eleven-armed bandit:
Algorithm: greedy_bandit
Iterations: 1000, Mean reward: 0.0, Mean regret: 1.0
Iterations: 10000, Mean reward: 0.0, Mean regret: 1.0
Iterations: 100000, Mean reward: 0.0, Mean regret: 1.0
Algorithm: ucb1_bandit
Iterations: 1000, Mean reward: 846.12, Mean regret: 0.15328999999999926
Iterations: 10000, Mean reward: 9634.36, Mean regret: 0.03638260000000045
Iterations: 100000, Mean reward: 99418.14, Mean regret: 0.005800630000000247
Algorithm: thompson_sampling_bandit
Iterations: 1000, Mean reward: 986.08, Mean regret: 0.014254999999999999
Iterations: 10000, Mean reward: 9986.1, Mean regret: 0.0013903
Iterations: 100000, Mean reward: 99986.17, Mean regret: 0.00013779999999999999
Testing on Five-armed bandit:
Algorithm: greedy_bandit
Iterations: 1000, Mean reward: 299.48, Mean regret: 0.55000000000000044
Iterations: 10000, Mean reward: 3007.03, Mean regret: 0.55000000000001022
Iterations: 100000, Mean reward: 29998.07, Mean regret: 0.55000000000008705
Algorithm: ucb1_bandit
Iterations: 1000, Mean reward: 790.84, Mean regret: 0.05796530000000053
Iterations: 10000, Mean reward: 8298.44, Mean regret: 0.01960542000000157
Iterations: 100000, Mean reward: 84448.09, Mean regret: 0.005631584999998115
Algorithm: thompson_sampling_bandit
Iterations: 1000, Mean reward: 830.4, Mean regret: 0.019289899999999863
Iterations: 10000, Mean reward: 8453.91, Mean regret: 0.004518010000000033
Iterations: 100000, Mean reward: 84918.16, Mean regret: 0.0008003910000000199

```

```

Testing on Eleven-armed bandit:
Algorithm: greedy_bandit
Algorithm: ucb1_bandit
Algorithm: thompson_sampling_bandit

```



Based on the graphs and results, we can analyze the properties of the three different algorithms: Greedy, UCB1, and Thompson Sampling.

1. Average Regret vs. Time:

Question assigned to the following page: [2](#)

- For the eleven-armed bandit setting, the Greedy algorithm performs poorly, with a constant high regret throughout the time steps. UCB1 and Thompson Sampling algorithms have significantly lower regret, with Thompson Sampling having the lowest regret overall.
 - For the five-armed bandit setting, the Greedy algorithm still performs poorly compared to UCB1 and Thompson Sampling. However, the difference in regret between UCB1 and Thompson Sampling is smaller compared to the eleven-armed bandit case.
2. Action Selection Over Time:
- For the eleven-armed bandit setting, the Greedy algorithm quickly converges to selecting the arm with the highest probability (arm 10), but it takes a long time to explore and identify the optimal arm.
 - UCB1 and Thompson Sampling explore more efficiently and converge faster to the optimal arm (arm 10) compared to the Greedy algorithm.
 - For the five-armed bandit setting, all three algorithms converge to the optimal arm (arm 4) relatively quickly, but UCB1 and Thompson Sampling still explore more efficiently and converge faster than the Greedy algorithm.

Interesting Insights:

1. The Greedy algorithm performs poorly in both settings, as it lacks an exploration mechanism and can get stuck on sub-optimal arms, leading to high regret.
2. UCB1 and Thompson Sampling outperform the Greedy algorithm by balancing exploration and exploitation effectively. They have lower regret and converge faster to the optimal arm.
3. Thompson Sampling generally performs better than UCB1, especially in the eleven-armed bandit setting, where the number of arms is larger. This suggests that Thompson Sampling is more efficient in exploring and identifying the optimal arm in complex environments with more choices.
4. The difference in performance between UCB1 and Thompson Sampling is more pronounced in the eleven-armed bandit setting compared to the five-armed bandit setting. This indicates that as the number of arms increases, the advantage of Thompson Sampling over UCB1 becomes more significant.
5. The shape of the average regret curves for UCB1 and Thompson Sampling suggests that they have a logarithmic regret bound, which is a desirable property for bandit algorithms.

Overall, the results demonstrate the superiority of UCB1 and Thompson Sampling over the Greedy algorithm in multi-armed bandit problems, with Thompson Sampling having a slight edge, especially in more complex environments with a larger number of arms.