Probability HW3

109350008 張詠哲

Problem1

Problem 1.

P1

Case 1. $Z=0 \Rightarrow P(X=x, Z=0) = \begin{cases} 1-p & \text{if } x=0 \\ 0 & \text{if } x>0. \end{cases}$

Case 2. $Z=1 \Rightarrow P(X=x, Z=1) = \begin{cases} p\cdot\frac{\lambda^x}{x!}e^{-\lambda} & \text{if } x\geq 0. \\ 0 & \text{otherwise.} \end{cases}$

$$\therefore P(X=x, Z=z) = \begin{cases} 1-p & \text{if } x=0 \text{ and } z=0 \\ p\cdot\frac{\lambda^x}{x!}e^{-\lambda} & \text{if } x\geq 0 \text{ and } z=1 \\ 0 & \text{otherwise} \end{cases}$$

a). Case $x=0$: $P(X=x) = (1-p)+p\cdot e^{-\lambda}$

Case $x>0$: $P(X=x) = p\cdot\frac{\lambda^x}{x!}e^{-\lambda}$

$$\therefore P(X=x) = \begin{cases} (1-p)+p\cdot e^{-\lambda} & \text{if } x=0. \\ p\cdot\frac{\lambda^x}{x!}e^{-\lambda} & \text{if } x>0. \\ 0 & \text{otherwise.} \end{cases}$$

b). When $\begin{cases} I=1, \ \tilde{X}=0 \ (\text{analogous to the case } Z=0) \\ I=0, \ \tilde{X}=Y \ (\text{analogous to the case } Z=1) \end{cases}$

Since $I \sim \text{Bernoulli}(1-p)$, the probability align

$$P(\tilde{X}=0) = P(I=1) + P(I=0)\cdot P(Y=0) = (1-p) + p\cdot e^{-\lambda}$$

$$P(\tilde{X}=x) = P(I=0)\cdot P(Y=x) = p\cdot \frac{\lambda^x}{x!}e^{-\lambda}, \text{ for } x>0.$$

$\therefore$ $\tilde{X}$ and X have the same PMF.

c). $E[UV] = \sum_u \sum_v uv \cdot P(U=u, V=v)$

$\because$ U & V are independent r.v., we can know

$$P(U=u, V=v) = P(U=u)\cdot P(V=v)$$

$$\therefore E[UV] = \sum_u \left(u\cdot P(U=u) \sum_v v\cdot P(V=v)\right)$$

$$= \sum_u \left(u\cdot P(U=u)\cdot E[V]\right)$$

$$= E[V]\cdot \sum_u u\cdot P(U=u) = \underline{E[V]\cdot E[U]}$$

Problem2

[P3]

Problem 2.

1). For X, Y ~ Possion($\lambda=2$), the MGF are

$$M_X(t) = M_Y(t) = \exp(2(e^t-1))$$

1. MGF of 3X $\Rightarrow$ $M_{3X}(t) = M_X(3t) = \exp(2(e^{3t}-1))$
2. MGF of 4Y $\Rightarrow$ $M_{4Y}(t) = M_Y(4t) = \exp(2(e^{4t}-1))$

Since X and Y are independent, we have

$$M_Z(t) = M_{3X}(t) \cdot M_{4Y}(t)$$

$$\Rightarrow M_Z(t) = \exp(2(e^{3t}-1)) \cdot \exp(2(e^{4t}-1))$$

$$= \exp(2(e^{3t}+e^{4t}-2))$$

For r.v. W ~ Posson(u), the MGF is

$M_W(t) = \exp(u(e^t-1))$, which is not match the $\exp(2(e^{3t}+e^{4t}-2))$. So Z is NOT a Posson r.v.

b) $p_Y(k) = \begin{cases} \frac{6}{\pi^2 k^2} & \text{if } k \in \mathbb{N} \\ 0 & \text{otherwise} \end{cases}$ P4

$$M_Y(t) = E[e^{tY}]$$

$$= \sum_{k=1}^{\infty} \frac{6}{\pi^2 k^2} \cdot e^{tk}$$

$$\geq \sum_{k=1}^{\infty} \frac{6}{\pi^2 k^2}(1+tk) \quad \because e^x \geq 1+x \text{ for } x \geq 0.$$

$$= \sum_{k=1}^{\infty} \frac{6}{\pi^2 k^2} + \frac{6t}{\pi^2 k}$$

$\because \sum_{k=1}^{\infty} \frac{6t}{\pi^2 k} = \infty$ for all $t > 0$., we can know

$M_Y(t)$ doesn't exist for any $y > 0$.

$\therefore$ MGF of Y doesn't exist #

Problem3

Problem 3. [P5]

a). $F_X(t) = F_{XY}(t,\infty) = \begin{cases} 1-\exp(-t) & \text{if } t>0 \\ 0 & \text{otherwise} \end{cases}$

$F_Y(t) = F_{XY}(\infty, u) = \begin{cases} 1-\exp(-u) & \text{if } u>0. \\ 0 & \text{otherwise} \end{cases}$

For X, Y to be independent, we need $F_{XY}(t,u) = F_X(t)\cdot F_Y(u)$

for all $t, u \in \mathbb{R}$, it will hold iff $\theta = 0$

b). $f_{XY}(t,u) = \frac{\partial}{\partial t \partial u} F_{XY}(t,u)$

$$= \begin{cases} \left(1+\theta(t+u-1)+\theta^2+u\right)\cdot e^{-(t+u+\theta tu)} & \text{if } t\geq 0,\ u>0 \\ 0 & \text{otherwise} \end{cases}$$

Problem4

Problem 4.

a). Let X_i be the r.v. for i-th labeler voting for T_a.

$$X_i = \begin{cases} 1 & \text{if the i-th labeler votes for } T_a \\ 0 & \text{otherwise} \end{cases}$$

$p = P(X_i = 1) = \sigma(R(T_a) - R(T_b))$

Let $S_n = \sum_{i=1}^{n} X_i$, the event E corresponds to $S_n > \frac{n}{2}$

$E[S_n] = \sum_{i=1}^{n} E[X_i] = n \cdot p$, by Hoeffding's inequality.

$P(|S_n - E[S_n]| \geq t) \leq 2\exp\left(-\frac{2t^2}{n}\right)$. For $P(E)$.

we are interested in $P(S_n > \frac{n}{2})$. We can rewrite this as

$$P\left(S_n > \frac{n}{2}\right) = P\left(S_n - E[S_n] > \frac{n}{2} - E[S_n]\right) \geq 1 - \exp\left(-\frac{2\left(\frac{n}{2} - E[S_n]\right)^2}{n}\right)$$

$\because E[S_n] = n \cdot p$. So we can know.

$$P(E) = P\left(S_n > \frac{n}{2}\right) \geq 1 - \exp\left(-\frac{2\left(\frac{n}{2} - np\right)^2}{n}\right)$$

$$= 1 - \exp\left(-\frac{2\left(\frac{n}{2} - n \cdot \sigma(R(T_a) - R(T_b))\right)^2}{n}\right)$$

lower bound for $P(E)$

b).

Chebyshev:

Follow from (a). we know $var(X_i) = p(1-p)$, $Var(S_n) = n \cdot Var(X_i) = n \cdot p(1-p)$. For $P(E)$, we want $P(S_n > \frac{n}{2}) = P(S_n - E[S_n] > \frac{n}{2} - E[S_n]) \geq 1-\delta$.

From Chebyshev's inequality: $P(|S_n - E[S_n]| \geq t) \leq \frac{Var(S_n)}{t^2}$

we can get: $P(S_n - E[S_n] > t) \geq 1 - \frac{n \cdot p(1-p)}{t^2}$,

Let $t = \frac{n}{2} - E[S_n] = \frac{n}{2} - np$. To achieve $P(E) \geq 1-\delta$

We need to solve: $\delta \geq \frac{n \cdot p(1-p)}{(\frac{n}{2} - np)^2}$

$$\delta \geq \frac{n \cdot p \cdot (1-p)}{(\frac{n}{2} - np)^2} = \frac{\cancel{n} p(1-p)}{n^{\cancel{2}}(\frac{1}{2} - p)^2}$$

$$\therefore n \geq \frac{p(1-p)}{\delta(\frac{1}{2} - p)^2}$$

Hoeffding inequality.

From Part (a). we already have $P(E) \geq 1 - \exp\left(-\frac{2(\frac{n}{2} - np)^2}{n}\right)$

To ensure $P(E) \geq 1-\delta$, we need to solve:

$$\exp\left(-\frac{2(\frac{n}{2} - np)^2}{n}\right) \leq \delta$$

$$\Rightarrow -\frac{2\,\cancel{n}^{n^2}(\frac{1}{2} - p)^2}{n} \leq \ln \delta$$

$$\Rightarrow -2n(\frac{1}{2} - p)^2 \leq \ln \delta$$

$$\Rightarrow n \geq \frac{-2 \ln \delta}{(\frac{1}{2} - p)^2}$$

c). Let $S = \sum_{i=2}^{100} I_i$, where I_i is independent Bernoulli r.v.

We want to know $P(S>90)$ upper bound.

We know $P(S \geq a) \leq e^{-ta} M_S(t)$ for $t>0$.

Because I_i is Bernoulli r.v., so the MGF of I_i is $M_{I_i}(t) = E[e^{tI_i}] = (1-p)+pe^t$.

the MGF of S will be $M_S(t) = \prod_{i=2}^{100} M_{I_i}(t) = \prod_{j=1}^{100} (1-p_j+p_j e^t)$

So by Chernoff bound, we can get

$$P(S>90) \leq e^{-t \cdot 90} \cdot \prod_{j=2}^{100} ((1-p_j)+p_j e^t) \text{ for any } t>0$$

and $p_j = \sigma(R(T_j)-R(T_1))$ for $j=2,3 \ldots 100$

Problem5

Problem 5. P10

a).

Step 1. We know N is the smallest integer s.t $S_N > 1$, which implies for $N = n$, $S_{n-1} < 1$ and $S_n > 1$.

So the $P(N=n) = P(S_{n-1} < 1 \cap S_n > 1)$,
it also means we need all cases that the sum less than 1 after $n-1$, and exclude the cases that less than 1 after n. So it can also write as.

$$P(N=n) = P(S_{n-1} < 1 \cap S_n > 1) = P(S_{n-1} < 1) - P(S_n < 1)$$

Step 2. To proof $P(S_n < 1) = \frac{1}{n!}$ for all $n \geq N$,
let $S_n = U_1 + U_2 + \dots + U_n$, the PDF $f_{S_n}(x)$ is obtained by repeating convolving the PDF of S_{n-1} with uniform distribution.

We first need to proof $f_{S_n}(x) = \frac{x^{n-1}}{(n-1)!}$ for $0 \leq x \leq n$.

1. base case: $S_1 = U_1$, $f_{S_1}(x) = 1 = \frac{x^{1-1}}{(1-1)!} = \frac{x^0}{0!}$ for $0 \leq x \leq 1$

P11

2. Assume $f_{S_{n-1}}(x) = \frac{x^{n-2}}{(n-2)!}$, for $0 \le x \le n-1$

by convolution theorem, we can know $S_n = S_{n-1} + U_n$.

So $f_{S_n}(x) = \int_0^1 1 \cdot f_{S_{n-1}}(x-t)\,dt = \int_0^1 1 \cdot \frac{(x-t)^{n-2}}{(n-2)!}\,dt$, for $0 \le x-t \le n-1$

which $t \in [\max(0, x-(n-1)), \min(1, x)]$, for $0 \le x \le n$.

We can get $t \in [0, x]$, so we can rewrite

$$f_{S_n}(x) = \frac{1}{(n-2)!}\int_0^x (x-t)^{n-2}\,dt$$

Let $u = x-t$, $du = -t$

$$f_{S_n}(x) = \frac{1}{(n-2)!}\int_x^0 u^{n-2}(-du) = \frac{1}{(n-2)!}\int_0^x u^{n-2}\,du$$

$$= \frac{1}{(n-2)!} \cdot \frac{x^{n-1}}{n-1}$$

$$= \frac{x^{n-1}}{(n-1)!}$$

so we can get $f_{S_n}(x) = \frac{x^{n-1}}{(n-1)!}$ by induction.

$\therefore P(S_n < 1) = \int_0^1 \frac{x^{n-1}}{(n-1)!}\,dx = \frac{1}{n!}$ #

Step 3. for step 1 and step 2. we can get

$$P(N=n) = P(S_{n-1}<1) - P(S_n<1)$$
$$= \frac{1}{(n-1)!} - \frac{1}{n!}$$
$$= \frac{n-1}{n!}$$

$$E[N] = \sum_{n=1}^{\infty} n \cdot P(N=n) = \sum_{n=1}^{\infty} n \cdot \frac{n-1}{n!} = \sum_{n=1}^{\infty} \frac{n^2-n}{n!}$$
$$= \sum_{n=1}^{\infty} \frac{n^2}{n!} - \sum_{n=1}^{\infty} \frac{n}{n!}$$
$$= 2e - e$$
$$= e$$

b).
1. Generate uniform r.v. $U_1, U_2, U_3 \ldots$ repeatedly
2. Compute the cummulative sum $S_n = \sum_{i=1}^{n} U_i$
3. Stop at smallest n where $S_n > 1$, and record $N = n$.
4. Repeat for a large number of trials.
5. Compute average mean (N) to estimate e.

(5-b Programming part)

All implementations are in the ZIP file.

Following table is my estimates of e under 10^1, 10^3, 10^5, and 10^7 sample trials

Number of trials	Estimate e
10^1	2.8
10^3	2.728
10^5	2.71844
10^7	2.7183075

We can clearly see that as the number of trials increase, the estimate e is closer to the actual Euler's number e = 2.71828.

Problem6

All implementations are in the ZIP file.

(a) What are the mean regrets of your Epsilon Greedy algorithm under the three MAB problem instances provided in "MAB.ipynb" and under ε = 0.01, 0.03, 0.1, 0.3? Please also compare the performance of Epsilon Greedy algorithm with that of the EmpiricalMeans algorithm.

Following table is the experiment result of my implementation under different ϵ and different three MAB problem:

epsilon	easy but widely adopted problem	An other problem	very hard problem
0.01	**223.65**	**90.288**	417.38
0.03	229.5	149.52	215.47
0.1	442.15	375.87	**140.03**
0.3	1214.9	1028.6	227.17

Table1. Mean regrets of my Epsilon Greedy algorithm in different condition.

We can see that small ϵ (ϵ = 0.01, 0.03) have a good result in simpler environments, I think it's because small ϵ prioritize exploitation (favoring the current best arm) and reduces regret because it quickly converges to high-reward arms. But in hard environments insufficient exploration prevents it from discovering better arms, leading to higher regret. While ϵ = 0.1 has a good result in hard environments. But in simpler environments it causes the increasing of the regret. In large ϵ (ϵ = 0.3) ensures thorough exploration, it comes at a high cost in simpler environments, where high-reward arms are easily identifiable. And the hard environments doesn't benefit much from large ϵ.

So, I think smaller ϵ values might are ideal for environments with clear optimal arms, minimizing unnecessary exploration. Moderate ϵ achieves the best balance in difficult environments with closely spaced arm rewards. Large ϵ leads to high regret across all environments due to over-exploration.

Following table is the comparison of my implementation and EmpiricalMeans and UCB algorithm (the ϵ here I set 0.01):

Algorithm	easy but widely adopted problem	An other problem	very hard problem
My implementation	**223.65**	**90.288**	**417.38**
EmpiricalMeans	307.81	155.31	435.77
UCB	327.14	231.21	490.1

Table2. Comparison of Mean regrets on 3 different algorithms.

We can see the above table, my Epsilon Greedy algorithm outperform at all MAB problem. I also compare to the UCB algorithm (as the given sample code), my Epsilon Greedy algorithm also outperform at all MAB problem. It means that the Epsilon-Greedy algorithm with appropriate ϵ can outperforms EmpiricalMeans and UCB in most environments. But I also observe that for hard environment, differences between the algorithms narrow, all algorithms struggle in this environment.

(b) Use a diminishing exploration rate in the Epsilon Greedy algorithm, i.e., ε(t) = $t^{-\alpha}$ with α > 0? Then, what are the mean regrets under α = 0.1, 0.5, 1.0, and 2.0? Can you point out some interesting things from your experimental results?

Following table is the experiment result of my implementation under different α and different three MAB problem:

epsilon	easy but widely adopted problem	An other problem	very hard problem
0.1	1775.3	1500.6	307.86
0.5	**161.02**	115.51	**218.17**
1.0	262.9	**75.89**	657.08
2.0	376.65	141.78	823.91

Table3. Mean regrets of my Epsilon Greedy algorithm in different alpha.

As the above table shows that the choice of α in the diminishing exploration rate (ε(t) = $t^{-\alpha}$) significantly affects performance across the three MAB problems. A small α (0.1) diminishes exploration too slowly, leading to high regret in simpler environments but decent performance in complex ones like the hard environment. Conversely, large α values (1.0, 2.0) reduce exploration too quickly, resulting in better performance in simpler problems but high regret in harder ones due to insufficient sampling of suboptimal arms. Interestingly, α=0.5 gets the best result in all environment. I think it's because it strikes the balance, achieving the lowest regrets overall by adapting exploration to problem complexity effectively. This shows that the optimal α depends on the specific environment, with appropreate values performing well across diverse settings.