سوال ۱:

الف) عبارت اول $\mathbb{E}_{s\sim D, a\sim\mu(a|s)}\{Q(s,a)\}$ و عبارت دوم $\mathbb{E}_{s\sim D, a\sim\hat{\pi}_\beta(a|s)}\{Q(s,a)\}$

است. مشکل اصلی در offline RL این است که Distributional Shift وجود دارد و در واقع Q-value ها overestimate می‌شوند. جمله اول که داریم Overestimate که $\mathbb{E}_{s\sim D, a\sim\mu(a|s)}\{Q(s,a)\}$ مسئول این است که عبارت‌های overestimate می‌شوند را کاهش دهیم. اما (s,a) هایی وجود دارند که داخل دیتای اصلی وجود دارند و در واقع overestimate نشده‌اند. در اینجا عبارت دوم که داریم $-\mathbb{E}_{s\sim D, a\sim\hat{\pi}_\beta(a|s)}\{Q(s,a)\}$ مسئول این است که عباراتی که قبلاً دیده‌ایم و می‌دانیم Overestimate نشده‌اند را کنترل کنیم. به این صورت اپت‌هایی که قبلاً ندیدیم را Overestimate نمی‌کنیم.

ب) ابتدا بهینه‌سازی داخل را $\max_\mu$ را حل می‌کنیم

$$\text{Objective: } \max_\mu \mathbb{E}_{s\sim D, a\sim\mu(a|s)}\{Q(s,a)\} + \mathbb{E}_{s\sim D}\{H(\mu(\cdot|s))\}$$

$$\text{s.t. } \forall s: \sum_a \mu(a|s) = 1$$

حال می‌توان این مسئله را برای هر s به صورت جداگانه حل کرد، در نهایت داریم که مسئله بهینه‌سازی ما به شکل زیر است:

$$\max_\mu \mathbb{E}\sum_a \mu(a|s)Q(s,a) - \sum_a \mu(a|s)\log(a|s)$$

$$\text{s.t. } \sum_a \mu(a|s) = 1$$

حال با استفاده از لاگرانژ داریم که:

$$L = \sum_a \mu(a|s)Q(s,a) - \sum_a \mu(a|s)\log(\mu(a|s)) + \lambda\left\{\sum_a \mu(a|s) - 1\right\}$$

$$\frac{\partial L}{\partial \mu(a|s)} = Q(s,a) - 1 - \log\mu(a|s) + \lambda = 0 \Rightarrow \mu(a|s) = e^{-1+\lambda}e^{Q(s,a)}$$

$$\sum_a \mu(a|s) = 1 \Rightarrow e^{-1+\lambda} = \frac{1}{\sum_{a'} e^{Q(s,a')}} \Rightarrow$$

$$\Rightarrow \mu(a|s) = \frac{e^{Q(s,a)}}{\sum_{a'} e^{Q(s,a')}}$$

~~$CQL = \min_Q \alpha \mathbb{E}_{s\sim D} \{ \mathbb{E} \sum_a \mu(a|s) Q(s,a)$~~

(I)

$$CQL = \min_Q \Big\{ \alpha \mathbb{E}_{s\sim D} \Big\{ \mathbb{E}_{a\sim\mu(a|s)} \{Q(s,a)\} \Big\} + \mathbb{E}_{s\sim D} \{ H(\mu(\cdot|s)) \}$$

$$- \alpha \mathbb{E}_{s\sim D} \Big\{ \mathbb{E}_{a\sim \hat{\pi}_\beta(a|s)} \{Q(s,a)\} \Big\} + \frac{1}{2} \mathbb{E}_{s,a,s'\sim D} \Big\{ (Q - \hat{B}^{\pi_k} \hat{Q}^k)^2 \Big\} \Big\} =$$

$$\rightarrow (I) = \min_Q \Big\{ \alpha \mathbb{E}_{s\sim D} \Big\{ \alpha \mathbb{E}_{a\sim\mu(a|s)} \{Q(s,a)\} + H(\mu(\cdot|s)) \Big\} =$$

$$= \min_Q \Big\{ \mathbb{E}_{s\sim D} \Big\{ \alpha \sum_a \mu(a|s) Q(s,a) - \sum_a \mu(a|s) \log(\mu(a|s)) \Big\} =$$

$$= \min_Q \Big\{ \mathbb{E}_{s\sim D} \Big\{ \alpha \sum_a \mu(a|s) \Big\{ Q(s,a) - Q(s,a) + \sum_{a'} \log \Big\{ \sum_{a'} \exp(Q(s,a')) \Big\} \Big\} \Big\} \Big\}$$

$$= \min_Q \Big\{ \mathbb{E}_{s\sim D} \Big\{ \alpha \log \Big(\sum_a \exp(Q(s,a)) \Big) \Big\} \Rightarrow$$

$$\Rightarrow CQL = \min_Q \Big\{ \alpha \mathbb{E}_{s\sim D} \Big\{ \log \sum_a \exp(Q(s,a)) - \mathbb{E}_{a\sim\hat{\pi}_\beta(a|s)} \{Q(s,a)\} \Big\}$$

$$+ \frac{1}{2} \mathbb{E}_{s,a,s'\sim D} \Big\{ (Q - \hat{B}^{\pi_k} \hat{Q}^k)^2 \Big\} \Big\}$$

(ج)

$$\mu^* = \arg\max_\mu \Big\{ \mathbb{E}_{s\sim D, a\sim\mu(a|s)} \{Q(s,a)\} + R(\mu) \Big\}$$

$$CQL = \min_Q \Big\{ \alpha \mathbb{E}_{s\sim D, a\sim\mu^*(a|s)} (Q(s,a)) + \frac{1}{2} \mathbb{E}_{s,a,s'\sim D} \Big\{ \big(Q(s,a) - \hat{B}^{\pi_k} \hat{Q}^k(s,a)\big)^2 \Big\}$$

$$+ R(\mu^*) \Big\}$$

$= \min_Q$ همانند قسمت قبل می توان برای هر s این رابطه را بازنویسی کرد

$$CQL = \min_Q \left\{ \mathbb{E}_{a\sim\mu^*(a|s)}\{Q(s,a)\} + \frac{1}{2}\mathbb{E}_{a\sim\hat{\pi}_\beta(a|s)}\left\{ \left(Q(s,a) - \hat{B}^{\hat{\pi}^k} \hat{Q}^k_{(s,a)}\right)^2 \right\} \right\}$$

A

حال می‌توان با مشتق گرفتن این عبارت نسبت به $Q(s,a)$ به دست آورد:

$$\frac{\partial CQL}{\partial Q}\frac{\partial A}{\partial Q_{(s,a)}} = 0 \rightsquigarrow \frac{\partial}{\partial Q_{(s,a)}}\left\{ \alpha \sum_a \mu^*(a|s) Q(s,a) + \frac{1}{2}\sum_a \hat{\pi}_\beta(a|s)\left(Q(s,a) - \hat{B}^{\hat{\pi}^k}\hat{Q}^k_{(s,a)}\right)^2 \right\} = 0$$

$$\Rightarrow \alpha\mu^*(a|s) + \hat{\pi}_\beta(a|s)\left(Q(s,a) - \hat{B}^{\hat{\pi}^k}\hat{Q}^k_{(s,a)}\right) = 0 \Rightarrow$$

$$\Rightarrow \hat{Q}^{k+1}_{(s,a)} = \hat{B}^{\hat{\pi}^k}\hat{Q}^k_{(s,a)} - \alpha\frac{\mu^*(a|s)}{\hat{\pi}_\beta(a|s)}$$

fixed-point : $\hat{Q}^{\hat{\pi}}_{(s,a)} = \hat{B}^{\hat{\pi}^\pi}\hat{Q}^\pi_{(s,a)} - \alpha\frac{\mu^*(a|s)}{\hat{\pi}_\beta(a|s)}$

$$\forall Q,s,a\in D : -\zeta_\delta(s,a) \preceq \hat{B}^{\hat{\pi}^k}\hat{Q}(s,a) - B^\pi Q(s,a) \preceq \zeta_\delta(s,a)$$

$$\forall Q : B^\pi Q = R + \gamma P^\pi Q$$: همچنین داریم

$$\hat{Q}^{\hat{\pi}}_{(s,a)} - \alpha\frac{\mu^*(a|s)}{\hat{\pi}_\beta(a|s)} \preceq \hat{B}^{\hat{\pi}}\hat{Q}_{(s,a)} + \zeta_\delta(s,a) = R + \gamma P^{\hat{\pi}}\hat{Q}_{(s,a)} + \zeta_\delta(s,a) - \alpha\frac{\mu^*(a|s)}{\hat{\pi}_\beta(a|s)}$$

$$\Rightarrow (I - \gamma P^\pi)\hat{Q}^{\hat{\pi}}_{(s,a)} \preceq R + \zeta_\delta(s,a) - \alpha\frac{\mu^*(a|s)}{\hat{\pi}_\beta(a|s)} \Rightarrow$$

$$\Rightarrow \hat{Q}^{\hat{\pi}}_{(s,a)} \preceq (I-\gamma P^\pi)^{-1}R - \alpha\left\{(I-\gamma P^\pi)^{-1}\left(\frac{\mu}{\hat{\pi}_\beta}\right)\right\}_{(s,a)} + \left\{(I-\gamma P^\pi)^{-1}\zeta_\delta\right\}_{(s,a)}$$

$$\Rightarrow \hat{Q}^{\hat{\pi}}_{(s,a)} \preceq Q^\pi_{(s,a)} - \alpha\left\{(I-\gamma P^\pi)^{-1}\left(\frac{\mu}{\hat{\pi}_\beta}\right)\right\}_{(s,a)} + \left\{(I-\gamma P^\pi)^{-1}\zeta_\delta\right\}_{(s,a)}$$

سوال 2)

الف) در Inverse Reinforcement Learning (IRL) هدف به دست آوردن تابع reward است با استفاده از دیتاهای expert. در این صورت با توجه به اینکه تابع به دست می‌آوریم در حقیقت دیتاهایی که expert آنها را دیده است، به صورت جنتلمن عمل می‌کند و همچنین در مقابل دیتاهایی که نادیده است بهتر عمل می‌کند.

در مقابل Behavarial Cloning (BC) به صورت مستقیم رفتار expert را تقلید می‌کند. با توجه به اینکه BC صرفاً تقلید می‌کند. در مقابل دیتاهای نویزی ممکن است action بدی بدهد و در زمان‌هایی که اینت دیده نشده، نمی‌تواند خوب عمل کند زیرا داده‌ای برای تقلید ندارد.

به علاوه با توجه به اینکه IRL یک تابع reward به دست می‌آورد، تقریباً می‌تواند رفتار expert را تحلیل کند. که در بعضی از شرایط این اتفاق می‌تواند منجر به این شود که IRL از یک رفتار sup-optimal یک رفتار بهتر پیدا کند.

ب) روش MaxEnt IRL یک روش از IRL است که در آن هدف به دست آوردن یک پارامتر فانکشن است که بتواند رفتار expert را توجیه کند طبیعتاً می‌توان چندین تابع مختلف را کد کرد. در روش MaxEnt IRL فانکشن را انتخاب می‌کنیم که در آن بین همه distribution های مختلف بیشترین آنتروپی را داشته باشد. به این صورت policy که یاد گرفته می‌شود، محاسبات بایاس را دارد و بیشترین عدم قطعیت به این صورت یک روش پیدا می‌کنیم policy که generalized ترین است.

$$P(\tau|\pi) = \frac{1}{Z(\theta)} \times \exp\left\{\sum_{t=0}^{T} R(s_t, a_t)\right\}$$

$$\text{Objective}: \max_{\theta} \left\{\sum_{\tau \in D} P_\tau \log P(\tau|\theta) - \lambda \left\| \mathbb{E}_{\pi_\theta}\{\phi(s,a)\} - \mathbb{E}_{\pi_E}\{\phi(s,a)\} \right\|\right\}$$

$$H(\pi) = -\sum_{a,s} \pi(a|s) \log \pi(a|s)$$

ماکسیمم کردن آنتروپی باعث می‌شود که policy غیرقطعی (uncertain) باقی بماند که باعث می‌شود محیط را بهتر explore کنیم و جلوگیری کند برای هیچ (s,a) که دیدمات expert جلوگیری می‌کند که overfit نشوند.

D: discriminator

$$J(\pi_E, L) = \mathbb{E}_{\pi_E}\{\log(1 - D(s,a))\}$$

$$J(\pi, L) = \mathbb{E}_{\pi}\{\log(D(s,a))\} - \lambda H(\pi)$$

این تابع جلوگیری می‌کند که فاصله بین پالیسی expert با پالیسی بدست آمده را بیشتر کند.

بازی min-max به این صورت کار می‌کند که اساتا maximize کردن نسبت به D تلاش می‌کند فاصله بین policy اکسپرت با policy یادگیری شده را زیاد کند و سپس با minimize کردن نسبت به π تلاش می‌کند که بهترین جواب را در بهترین محیط پیدا کند.

د) MaxEnt IRL مسائلی از نظر objective و حساب نسبت به ویژگی‌های انتخاب شده دارد. و بعضی اوقات نیاز به تعداد زیادی sample دارد.

در روش GAIL با توجه به اینکه یک مدل adverserial است باعث می‌شود که پیچیدگی حل معادله اصلی کاهش پیدا کند. و نسبت به ویژگی‌ها خیلی منعطف‌تر است. (لذا می‌تواند یک neural net باشد)

همچنین عبارت آنتروپی داخل GAIL باعث می‌شود که exploration, overfitting بهتر شوند.

سوال (3):

الف) روش هایی که تنها از مدل محیط استفاده می کنند:

عملکرد مدل به شدت متکی به مدل است. اگر محیط بیش از اندازه ساده سازی شده باشد یا مشکل داشته باشد، خروجی مدل یک sub optimal policy می شود یا حتی policy آسیب زا. همچنین یک مسئله به نام reality gap وجود دارد که ممکن است policy ها را نتوان به دنیای واقعی تعمیم داد. همچنین این متدها نمی توانند اتفاقات غیرعادی را پیش بینی کنند و در شرایط می توانند overfit شوند.

روش هایی که تنها با محیط تعامل می کنند:

تعامل با محیط اغلب هزینه زیادی دارد و مدت زیادی صرف تعامل می شود.

همچنین در بعضی action ها ممکن است که تعامل با محیط آسیب زا باشد.

خوبی روش های ترکیبی:

این روش ها می توانند از خوبی های هر دو روش بهره ببرند. که در این صورت می توان generalization بیتری داشته باشیم.

هزینه کمتری برای بدست آوردن sample ها پیدا کنیم (از لحاظ زمان و منابع) و همچنین در جاهایی که آسیب زا است از interaction با محیط پرهیز کنند.

ب)

روش MBPO از دو روش مدل محیط و تعامل با محیط بهره می برد.

این روش بدین صورت کار می کند که ابتدا با استفاده از تعامل با محیط یک سری داده ایجاد می کند و سپس با یادگیری مدل تعدادی داده جدید ایجاد می کند. dataset تجمیعی از این دو مدل ایجاد می کند. بدین صورت مدل می تواند سناریوهای مختلف بیتری را ببیند

انتخاب طول roll-out یک انتخاب حیاتی است. زیرا

طول کم roll-out به معنی این است که error مدل را minimize می کند اما diversity کمتری دارد.

طول زیاد roll-out مشکل diversity را رفع می کند اما به خاطر تجمع خطاهای باعث می شود که خطای بیتری داشته باشد.

در نتیجه باید یک انتخاب درست بین برای طول roll-out مشخص کرد که trade-off خوبی بین خطا و diversity داشته باشیم.

۸

به صورت پویا

ج) همانطور که در قسمت های قبل توضیح دادیم، روش MBPO کاملاً مبتنی بر interaction با محیط است. که به صورت پیوسته در صورت نیاز با مدل offline خود interact کند و مدل خود را update کند. در حالتی که داده ها offline باشند، MBPO نمی تواند با محیط interact کند، و خطاهای خود را تصحیح کند. و مشکلات OOD را حل کند.

د) یک روش برای این میزان تردید می تواند Ensemble Methods باشد. در این روش چندین مدل مختلف بر روی offline data آموزش می بینیم به این صورت هر مدل جداگانه برای next state تصمیم گیری می کند. حال بین آن مدل های مختلف واریانس را حساب می کنیم و Threshold برای متوقف کردن یک واریانس تعیین می کنیم

ه)

$$Q_{new}(s,a) = \min_i Q_{old}(s,a) + \gamma \mathbb{E}_{a' \sim \pi} [Q_{target}(s',a')]$$

COMBO از ایده های دو روش CQL و Dyna استفاده می کند

CQL : از روش CQL ایده ای را استفاده می کند که با قرار دادن یک پنالتی جلوی overestimation برای Q-value ها را می گیرد و محافظه کارانه عمل می کند

Dyna : از این ایده استفاده می کند که با استفاده از مدل آموزش دیده همیشه داده های جدیدی ایجاد کند و از آنها استفاده کند.

COMBO با استفاده از این ایده ها تلاش می کند که با داده های جدید exploration انجام دهد و Q-value ها را به صورت محافظه کارانه آپدیت کند. به این صورت به هدف هر دو روش Dyna و CQL رسیده است.

۹) هر دو روش COMBO و MoRel برای offline RL هستند.

COMBO

Strengths:

(۱) از یک روش مبتنی بر مدل و محافظه‌کارانه استفاده می‌کند که یک بالانس بین exploration و safety برقرار کند.

(۲) sample-efficient: استفاده از یک مدل برای ایجاد داده‌ها باعث می‌شود که efficiency بهتر شود.

(۳) Robustness: محافظه‌کارانه بودن این روش باعث می‌شود که به policy هایی قابل اطمینان‌تری برسیم

Weaknesses:

(۱) Model dependency: وابسته به کیفیت مدل است.

(۲) Coservatism: بعضی اوقات خیلی محافظه‌کارانه عمل می‌کند

(۳) Complexity: ایجاد تعادل بین exploration و Conservatism می‌تواند هزینه زیادی داشته باشد.

MORel:

Strengths

(۱) Explicit Uncertainty: عدم قطعیت را به صورت منظم دنبال می‌کند.

(۲) Efficiency: استفاده از آنسامبل‌ها efficient می‌کند

(۳) Safety: با تغییر MDP مطمئن می‌شود که policy ایمن‌شده امن است.

Weaknesses:

(۱) Model dependency:

(۲) Computational Intensity: استفاده از آنسامبل زمان زیادی می‌برد.

(۳) Conservatism

در کل هر دو روش برای offline RL مناسب هستند. COMBO یک تعادل بین محافظه‌کاری و exploration برقرار می‌کند در صورتی که MORel کاملاً از شرایط غیرقطعی دوری می‌کند. هر دوی این روش‌ها از Model استفاده می‌کنند که این اتفاق باعث وابستگی بالای آن‌ها به دقت مدل می‌باشد که مشکل‌زا است.

سوال (۲):

الف) طبق نامساوی Hoeffding داریم که:

$S_n = \sum_i X_i \; , \; a_i \le X_i \le b_i$

$\Rightarrow P\left(E(\bar{X}_n) - \bar{X}_n \ge t\right) \le \exp\left\{-\frac{2t^2}{\sum_i (b_i - a_i)^2}\right\}$ $\quad X_i \to$ i.i.d.

حال اگر داشته باشیم: $a_i = -1, b_i = 1$ و همچنین $\bar{X} = \frac{\sum X_i}{n}$

$\Rightarrow P\{|E(X) - \bar{X}| \ge \varepsilon\} \le \exp\left\{-\frac{n\varepsilon^2}{2}\right\}$

فرض کنید که مقدار $\varepsilon = \sqrt{\frac{2\log(1/\delta)}{n}}$ داریم که

$P\left\{|\bar{X} - E(X)| \ge \sqrt{\frac{2\log(1/\delta)}{n}}\right\} \le \delta \Rightarrow P\left(\mu \ge \hat{\mu} + \sqrt{\frac{2\log(1/\delta)}{n}}\right) \le \delta$

در نتیجه داریم که با احتمال $1-\delta$ $\leftarrow$ $\mu \le \hat{\mu} + \sqrt{\frac{2\log(1/\delta)}{n}}$ $\leftarrow$ w.h.p

در نتیجه با احتمال بالا داریم که $\hat{\mu} + \sqrt{\frac{2\log(1/\delta)}{n}}$ یک حد بالا برای μ است.

ب) بدون کاستن از مسئله فرض می‌کنیم که بهترین arm حتما arm اول باشد و $\mu^* = \mu_1$ باشد.

قبل از ادامه دادن اثبات notation را به صورت زیر تعریف می‌کنیم:

$(X_{ti})_{t \in [n], i \in [k]} \to$ i.i.d variables $\to$ پاداش مشاهده شده در tامین بار کشیدن arm i

$\hat{\mu}_{is} = \frac{1}{s}\sum_{u=1}^{s} X_{ui}$ $\qquad R_n = \sum_{i=1}^{k} \Delta_i E\{T_i(n)\}$

این اثبات به این صورت است که یک bound برای $E(T_i(n))$ $\forall i$ suboptimal پیدا می‌کنیم. رویداد G_i را به این صورت تعریف می‌کنیم:

$G_i = \left\{\mu_1 < \min_{t \in [n]} UCB_1(t, \delta)\right\} \cap \left\{\hat{\mu}_{i u_i} + \sqrt{\frac{2}{u_i}\log\left(\frac{1}{\delta}\right)} < \mu_1\right\}$

که $u_i \in [n]$ یک ثابت است که در ادامه مشخص می‌کنیم.

حال ② چیز اثبات می‌کنیم:

① : اگر G_i اتفاق بیوفتد، arm i حداکثر u_i بار انتخاب شده

② G_i^C (Complement) به احتمال کم اتفاق می‌افتد.

بدیهی است که $T_i(n) \le n$ (IV)

$$\mathbb{E}(T_i(n)) = \mathbb{E}\{\mathbb{I}\{G_i\} T_i(n)\} + \mathbb{E}\{\mathbb{I}\{G_i^c\} T_i(n)\} \le u_i + n\mathbb{P}(G_i^c)$$

برای اثبات قضیه ① از برهان خلف استفاده می کنیم.

فرض کنید که $T_i(n) > u_i$ باشد. پس i: arm پیش از بار n ام بازی شده

پس یک دور مانند $t \in [n]$ وجود دارد که $T_i(t-1) = u_i$ و با استفاده از $A_t = i$

از تعریف G_i داریم که:

$$UCB_i(t-1,\delta) = \hat{\mu}_i(t-1) + \sqrt{\frac{2\log(1/\delta)}{T_i(t-1)}} \Rightarrow UCB_i(t-1,\delta) = \hat{\mu}_{i u_i} + \sqrt{\frac{2\log(1/\delta)}{u_i}} \quad \{M_1\} \quad UCB_1(t-1,\delta)$$

تعریف UCB_i

در نتیجه داریم که $A_t = \operatorname{argmax}_j UCB_j(t-1,\delta) \neq i$ که تناقض است ~~$A_t = \operatorname{argmax}$~~

اثبات قضیه ②:

برای اثبات قضیه ② طبق تعریف داریم

$$G_i^c = \{M_1 \ge \min_{t\in[n]} UCB_1(t,\delta)\} \cup \{\hat{\mu}_{i u_i} + \sqrt{\frac{2\log(1/\delta)}{u_i}} \ge M_1\} \quad (III)$$

طبق تعریف $UCB_1(t,\delta)$ داریم که

$$\{M_1 \ge \min_{t\in[n]} UCB_1(t,\delta)\} \subset \{M_1 \ge \min_{s\in[n]} \hat{\mu}_{1s} + \sqrt{\frac{2\log(1/\delta)}{s}}\} =$$

$$= \bigcup_{s\in[n]} \{M_1 \ge \hat{\mu}_{1s} + \sqrt{\frac{2\log(1/\delta)}{s}}\}$$

حال با توجه به Union bound داریم که

$$\mathbb{P}(M_1 \ge \min_{t\in[n]} UCB_1(t,\delta)) \le \mathbb{P}\left(\bigcup_{s\in[n]} \{M_1 \ge \hat{\mu}_{1s} + \sqrt{\frac{2\log(1/\delta)}{s}}\}\right)$$

$$\le \sum_{s=1}^{n} \mathbb{P}\left(M_1 \ge \hat{\mu}_{1s} + \sqrt{\frac{2\log(1/\delta)}{s}}\right) \le n\delta$$

Hoeffding (II)

حال که (2) قضیه را ثابت کردیم، فرض کنید که u_i به صورتی انتخاب شده که

$$D_i - \sqrt{\frac{2\log(1/\delta)}{u_i}} \geq cD_i \rightarrow c_i \in (0,1)$$

$$\mu_1 = \mu_i + D_i \Rightarrow \mathbb{P}\left(\hat{\mu}_{iu_i} + \sqrt{\frac{2\log(1/\delta)}{u_i}} \geq \mu_1\right) =$$

$$= \mathbb{P}\left(\hat{\mu}_{iu_i} - \mu_i \geq D_i - \sqrt{\frac{2\log(1/\delta)}{u_i}}\right) \leq \mathbb{P}(\hat{\mu}_{iu_i} - \mu_i \geq cD_i) \leq$$

(I) $\exp\left(-\frac{u_i c^2 D_i^2}{2}\right)$

(I), (II), (III) $\Rightarrow \mathbb{P}(G_i^c) \leq n\delta + \exp\left\{-\frac{u_i c^2 D_i^2}{2}\right\}$

(IV) $\Rightarrow \mathbb{E}(T_i(n)) \leq u_i + n\left(n\delta + \exp\left\{-\frac{u_i c^2 D_i^2}{2}\right\}\right)$

حال کوچکترین u_i را انتخاب می‌کنیم به صورتی که $D_i - \sqrt{\frac{2\log(1/\delta)}{u_i}} \geq cD_i$ باشد

$$\hookrightarrow u_i = \left\lceil \frac{2\log(1/\delta)}{(1-c)^2 D_i^2} \right\rceil$$

حال با توجه به اینکه $T_i(n) \leq n$ و با فرض $\delta = \frac{1}{n^2}$

$$\mathbb{E}(T_i(n)) \leq u_i + 1 + n^{1-\frac{2c^2}{(1-c)^2}} = \left\lceil \frac{2\log(1/\delta)}{(1-c)^2 D_i^2} \right\rceil + 1 + n^{1-\frac{2c^2}{(1-c)^2}}$$

حال تنها کافی است $c \in (0,1)$ را انتخاب کنیم

اگر c خیلی به (1) نزدیک باشد مقدار u_i به سمت بی‌نهایت می‌رود و همچنین اگر c خیلی به صفر باشد توان n باید کمتر از صفر باشد. به صورت دلخواه $c = 1/2$ قرار می‌دهیم

$$\Rightarrow \mathbb{E}(T_i(n)) \leq 3 + \frac{16\log(n)}{D_i^2}$$

2) برای اثبات این قسمت، ابتدا اثبات می‌کنیم که

$$R_n = \sum_{i=1}^{k} \Delta_i \mathbb{E}\{T_i(n)\}$$

Proof

$$R_n = n\mu^* - \mathbb{E}\{S_n\} = \sum_{a \in A} \sum_{i=1}^{k} \sum_{t=1}^{n} \mathbb{E}\left[(\mu^* - X_t) \mathbb{I}\{A_t = i\}\right]$$

$$\left(\rightarrow S_n = \sum_t X_t = \sum_t \sum_{i=1}^{k} X_t \mathbb{I}\{A_t = i\}\right.$$

حال می‌توان نوشت که،

$$\mathbb{E}\{(\mu^* - X_t)\mathbb{I}\{A_t = i\}\} = \mathbb{I}\{A_t = i\}\{\mathbb{E}\{\mu^* - X_t | A_t\}$$

$$= \mathbb{I}\{A_t = i\}(\mu^* - \mu_{A_t}) = \mathbb{I}\{A_t = i\}(\mu^* - \mu_i) =$$

$$= \mathbb{I}\{A_t = i\}\Delta_i \Rightarrow$$

$$\Rightarrow \mathbb{E}\{(\mu^* - X_t)\mathbb{I}\{A_t = i\}\} = \mathbb{E}\{\mathbb{I}\{A_t = i\}\Delta_i\}$$

$$R_n = \sum_i \Delta_i \sum_t \mathbb{E}\{\mathbb{I}\{A_t = i\}\} = \sum_{i=1}^{k} \Delta_i \mathbb{E}(T_i(n))$$

حال با جایگذاری این عبارت در چیزی که قسمت ب) اثبات کردیم داریم که

$$R_n \leq 3\sum_{i=1}^{k} \Delta_i + \sum_{i:\Delta_i > 0} \frac{16 \log(n)}{\Delta_i}$$

۵) داریم که

$$R_n = \sum_{i=1}^{k} \Delta_i \, \mathbb{E}\{T_i(n)\}$$

فرض کنیم $\Delta = \sqrt{\frac{16k\log(n)}{n}}$ می‌باشد.

$$R_n = \sum_{i:\Delta_i < \Delta} \Delta_i \mathbb{E}(T_i(n)) + \sum_{i:\Delta_i \geq \Delta} \Delta_i \mathbb{E}(T_i(n))$$

$$\sum_{i:\Delta_i < \Delta} T_i(n) \leq n$$

$$\leq 3 + 16\frac{\log(n)}{\Delta_i^2}$$

$$\Rightarrow R_n \leq n\Delta + \sum_{i:\Delta_i \geq \Delta} \left\{ 3\Delta_i + \frac{16\log(n)}{\Delta_i} \right\}$$

$$\leq n\Delta + \frac{16k\log(n)}{\Delta} + 3\sum_i \Delta_i$$

حال با جایگذاری Δ داریم

$$R_n \leq 8\sqrt{nk\log(n)} + 3\sum_i \Delta_i$$

اثبات این سوال با استفاده از کتاب

Bandit Algorithms

Tor Lattimore and Csaba Szepesvári