Лекция 9

Обучение с подкреплением

Никита Юдин, iudin.ne@phystech.edu

Московский физико-технический институт
Физтех-школа прикладной математики и информатики

17 апреля 2024

ФПМИ

Deep Deterministic Policy Gradient (DDPG)

Схема DQN имела принципиальный недостаток: мы не могли работать с непрерывными пространствами действий в силу необходимости постоянно считать операторы максимума и аргмаксимума

$$\max_a Q_\theta(s, a)$$

как для жадного выбора действия, так и для построения таргета в задаче регрессии.

Единственная возможная архитектура модели — приём действий на вход вместе с состояниями, тогда поиск аргмаксимума проводить, в целом, можно, но дорого: инициализируем a^0 случайно, и устраиваем градиентный подъём по входу в модель:

$$a^{k+1} = a^k + \alpha \left. \nabla_a Q_\theta(s, a) \right|_{a=a^k}.$$

Итак, пусть $\pi_\omega(s)$ принимает на вход состояние s и выдаёт аргмаксимум текущей аппроксимации Q-функции, то есть будем добиваться $\pi_\omega(s) \approx \operatorname{argmax}_a Q_\theta(s, a)$. Понятно, как обучать такую сеть:

$$Q_\theta(s, \pi_\omega(s)) \to \max_\omega .$$

Весь алгоритм DQN оставляем неизменным с единственной модификацией, что на каждом батче также нужно сделать шаг оптимизации ω. При этом каждый раз, когда в схеме необходимо считать максимум или аргмаксимум Q_θ, используется $\pi_\omega(s)$.
В стандартном алгоритме DQN нам было необходимо считать $\max_{a'} Q_{\theta^-}(s', a')$, и в дефолтной версии алгоритма использовалась таргет-сеть. Технически это означает, что для таргет-сети $Q_{\theta^-}(s', a')$ нам тоже нужно знать аргмаксимум, поэтому можно хранить старую версию вспомогательной функции $\pi_{\omega^-}(s)$.

Итого мы получили, что для жадного выбора действия используется $\pi_\omega(s)$ (отсюда такое обозначение этой «вспомогательной» функции — это фактически стратегия); а таргет для перехода $\mathbb{T} \coloneqq (s, a, r, s')$ вычисляется по формуле

$$y(\mathbb{T}) \coloneqq r + \gamma Q_{\theta^-}(s', \operatorname*{argmax}_{a'} Q_{\theta^-}(s', a')) \approx r + \gamma Q_{\theta^-}(s', \pi_{\omega^-}(s)).$$

Такой алгоритм называется Deep Deterministic Policy Gradient (DDPG), и название может сбить с толку: а причём здесь policy gradient?

Теорема 1 — Deterministic Policy Gradient: В непрерывных пространствах действий в предположении дифференцируемости Q-функций по действиям:

$$\nabla_\theta J(\pi_\theta) = \frac{1}{1-\gamma}\mathbb{E}_{d_{\pi_\theta}(s)} \left(\nabla_\theta \pi_\theta(s)\right)^\top \nabla_a Q^\pi(s,a)\big|_{a=\pi_\theta(s)}. \qquad (1)$$

Доказательство.

$$\nabla_\theta V^{\pi_\theta}(s) = \{\text{VQ уравнение}\} = \nabla_\theta \mathbb{E}_{a\sim\pi_\theta(s)} Q^{\pi_\theta}(s,a) = \nabla_\theta Q^{\pi_\theta}(s,\pi_\theta(s)) = (*).$$

Заметим, что в последнем выражении при малом изменении θ поменяется не только $\pi_\theta(s)$, но и сама оценочная функция Q^{π_θ}. Считая, что якобиан функции $\mathbb{R}^n \to \mathbb{R}^m$ имеет размерность $n \times m$, и обозначая размерность действий как A, а размерность параметров θ буквой d, получаем следующие размерности матриц и векторов:

$$\nabla_\theta Q^{\pi_\theta}(s,a) \in \mathbb{R}^{d\times 1}, \quad \nabla_a Q^{\pi_\theta}(s,a) \in \mathbb{R}^{A\times 1}, \quad \nabla_\theta \pi_\theta(s) \in \mathbb{R}^{A\times d}.$$

Тогда продолжение вычисления выглядит так:

$$(*) = \left(\nabla_\theta \pi_\theta(s)\right)^\top \nabla_a Q^\pi(s,a)\big|_{a=\pi_\theta(s)} + \nabla_\theta Q^{\pi_\theta}(s,a)\big|_{a=\pi_\theta(s)},$$

где последнее слагаемое — градиент Q^{π_θ} при фиксированном a по параметрам стратегии π_θ, которую он оценивает. Отдельно это слагаемое имеет вид:

$$\nabla_\theta Q^{\pi_\theta}(s,a) = \{\text{QV уравнение }\} = \gamma \mathbb{E}_{s'} \nabla_\theta V^{\pi_\theta}(s').$$

Получаем рекурсивную формулу, аккуратно собирая которую, получим:

$$\nabla_\theta V^{\pi_\theta}(s) = \mathbb{E}_{\mathcal{T}\sim\pi|s_0=s} \sum_{t\geq 0} \gamma^t \left(\nabla_\theta \pi_\theta(s_t)\right)^\top \nabla_a Q^\pi(s_t,a)\big|_{a=\pi_\theta(s_t)}.$$

Осталось только применить теорему об эквивалентной форме мат.ожидания по траекториям для

$$f(s,a) = \left(\nabla_\theta \pi_\theta(s)\right)^\top \nabla_a Q^\pi(s,a)\big|_{a=\pi_\theta(s)}. \qquad \blacksquare$$

Сразу построим суррогатную функцию для такой формулы градиента:

$$\mathcal{L}_{\tilde{\pi}}(\theta) := \frac{1}{1-\gamma} \mathbb{E}_{d_{\tilde{\pi}}(s)} Q^{\tilde{\pi}}(s, \pi_\theta(s)).$$

Действительно, если мы посчитаем градиент этой функции по θ, то мы просто получим формулу chain rule для оптимизации параметров стратегии через градиент Q-функции по действиям. Иными словами, градиент по параметрам детерминированной стратегии указывает просто проводить policy improvement: выбирать те действия, для которых Q-функция больше, используя её градиент по действиям. Если мы хотим построить Actor-Critic схему, воспользовавшись такой формулой, нам придётся аппроксимировать Q-функцию $Q_\omega(s, a) \approx Q^\pi(s, a)$ и явно использовать её градиент по действиям, надеясь на то, что $\nabla_a Q_\omega(s, a) \approx \nabla_a Q^\pi(s, a)$.

Итого, будем сэмплировать батч состояний из реплей буфера и делать шаг градиентного подъёма:

$$\theta \leftarrow \theta + \alpha \mathbb{E}_s \left(\nabla_\theta \pi_\theta(s) \right)^\top \nabla_a Q^\pi(s,a)\big|_{a=\pi_\theta(s)},$$

где состояния s приходят из произвольного распределения (например, из реплей буфера). Это эквивалентно одному шагу градиентной оптимизации суррогатной функции:

$$\mathbb{E}_s Q^\pi(s, \pi_\theta(s)) \to \max_\theta . \tag{2}$$

Q-функцию $Q_\omega(s,a) \approx Q^\pi(s,a)$, необходимую для такой оптимизации, будем тоже учить в off-policy режиме с одношаговых оценок: ему для данной пары s,a требуется лишь сэмпл s', поэтому такого критика можно обучать по переходам $\mathbb{T} := (s,a,r,s')$ из буфера на таргеты

$$y(\mathbb{T}) := r + \gamma Q_{\omega^-}\left(s', \pi(s')\right).$$

Теорема

Предыдущие две схемы (вывод через DQN и через Policy Gradient) эквивалентны полностью.

Доказательство. *Методом пристального взгляда.* ∎

В рассмотренной схеме из-за использования детерминированной стратегии, как и в DQN, возникает проблема exploration-exploitation-а. В непрерывных пространствах действий вместо ε-жадной стратегии возможно добавлять к выбранным стратегией действиям шум из нормального распределения:

$$a_t := \pi(s_t) + \varepsilon_t, \qquad \varepsilon_t \sim \mathcal{N}(0, \sigma^2 I).$$

Пример

Если действия робота — это направление движения (например, поворот руля управляемой машины), а один шаг в среде это доля секунды, странно проводить исследования, рандомно «подёргиваясь» пару раз в секунду. Хочется целенаправленно смещать траекторию: если мы решили в целях исследования повернуть руль чуть правее, чем говорит наша детерминированная стратегия, следует сохранить это смещение руля вправо и в дальнейшем. Для моделирования этого шум должен быть скоррелированным: поэтому вместо независимого шума имеет смысл добавлять случайный процесс, колеблющийся вокруг нуля.

Определение

Шум Орнштейна — Уленбека *(Ornstein–Uhlenbeck noise), в начале эпизода инициализированный нулём, задаётся рекурсивно как:* $\varepsilon_{t+1} \coloneqq \alpha\varepsilon_t + \mathcal{N}(0, \sigma^2 I)$, *где* $\alpha \le 1$ *и* σ *— гиперпараметры.*

Алгоритм 1: Deep Deterministic Policy Gradient (DDPG)

Гиперпараметры: B — размер мини-батчей, β — коэф. экспоненциального сглаживания для таргет-сеток, α, σ — параметры шума, $Q_\theta(s,a)$ — нейросетка с параметрами θ, $\pi_\omega(s)$ — детерминированная стратегия с параметрами ω, SGD-оптимизаторы, K — периодичность обновления весов таргета.

Инициализировать θ, ω произвольно

Положить $\theta^- := \theta$

Положить $\omega^- := \omega$

Инициализировать шум $\varepsilon := 0$

Пронаблюдать s_0

На очередном шаге t:

1. обновить шум $\varepsilon \leftarrow \alpha\varepsilon + \hat{\varepsilon}$, где $\hat{\varepsilon} \sim \mathcal{N}(0, \sigma^2 I)$
2. выбрать $a_t := \pi_\omega(s_t) + \varepsilon$
3. пронаблюдать r_t, s_{t+1}, done_{t+1}
4. добавить пятёрку $(s_t, a_t, r_t, s_{t+1}, \mathrm{done}_{t+1})$ в реплей буфер
5. засэмплировать мини-батч размера B из буфера

6 сделать один шаг градиентного подъёма по ω:

$$\frac{1}{B}\sum_{s\in B} Q_\theta(s, \pi_\omega(s)) \to \max_\omega$$

7 для каждого перехода $\mathbb{T} := (s, a, r, s', \mathrm{done})$ посчитать таргет:

$$y(\mathbb{T}) := r + \gamma(1 - \mathrm{done}) Q_{\theta^-}\left(s', \pi_{\omega^-}\left(s'\right)\right)$$

8 сделать один шаг градиентного спуска по θ:

$$\frac{1}{B}\sum_{\mathbb{T}} \left(Q_\theta(s, a) - y(\mathbb{T})\right)^2 \to \min_\theta$$

9 обновить таргет-сети, если $t \bmod K = 0$:

$$\theta^- \leftarrow (1 - \beta)\theta^- + \beta\theta$$

$$\omega^- \leftarrow (1 - \beta)\omega^- + \beta\omega$$

Алгоритм 2: Twin Delayed DDPG (TD3)

Гиперпараметры: B — размер мини-батчей, N — периодичность обновления весов стратегии, α, σ — параметры шума, $\hat{\sigma}, c$ — параметры шума для добавки к действиям для таргета, β — коэф. экспоненциального сглаживания для таргет-сеток, $Q_{\theta_1}(s,a), Q_{\theta_2}(s,a)$ — нейросетки с параметрами θ_1, θ_2, $\pi_\omega(s)$ — детерминированная стратегия с параметрами ω, SGD-оптимизаторы.

Инициализировать $\theta_1, \theta_2, \omega$ произвольно
Инициализировать таргет-сетки $\theta_1^- := \theta_1, \theta_2^- := \theta_2, \omega^- := \omega$
Инициализировать шум $\varepsilon_0 := 0$
Пронаблюдать s_0
На очередном шаге t:

1. посчитать шум $\varepsilon_t := \alpha\varepsilon_{t-1} + \varepsilon$, где $\varepsilon \sim \mathcal{N}(0, \sigma^2 I)$
2. выбрать $a_t := \pi_\omega(s_t) + \varepsilon_t$
3. пронаблюдать r_t, s_{t+1}, done_{t+1}
4. добавить пятёрку $(s_t, a_t, r_t, s_{t+1}, \mathrm{done}_{t+1})$ в реплей буфер
5. засэмплировать мини-батч размера B из буфера

6 для каждого перехода $\mathbb{T} := (s, a, r, s', \mathrm{done})$ посчитать таргет:

$$\varepsilon' \sim \mathrm{clip}(\mathcal{N}(0, \hat{\sigma} I), -c, c)$$

$$y(\mathbb{T}) := r + \gamma(1 - \mathrm{done}) \min_{i \in \{1,2\}} Q_{\theta_i^-}(s', \pi_{\omega^-}(s') + \varepsilon')$$

7 сделать один шаг градиентного спуска по θ_1 и θ_2:

$$\frac{1}{B} \sum_{\mathbb{T}} \left(Q_{\theta_1}(s, a) - y(\mathbb{T}) \right)^2 \to \min_{\theta_1}$$

$$\frac{1}{B} \sum_{\mathbb{T}} \left(Q_{\theta_2}(s, a) - y(\mathbb{T}) \right)^2 \to \min_{\theta_2}$$

8 **если** $t \bmod N = 0$:

- сделать один шаг градиентного подъёма по ω:

$$\frac{1}{B} \sum_{s \in B} Q_{\theta_1}(s, \pi_\omega(s)) \to \max_\omega$$

- обновить таргет-сети:

$$\theta_1^- \leftarrow (1-\beta)\theta_1^- + \beta\theta_1$$

$$\theta_2^- \leftarrow (1-\beta)\theta_2^- + \beta\theta_2$$

$$\omega^- \leftarrow (1-\beta)\omega^- + \beta\omega$$

Обучение стохастических политик

Определение

Скажем, что для параметризации $\pi_\theta(a \mid s)$ применим ***репараметризационный трюк*** *(reparameterization trick), если сэмплирование $a \sim \pi_\theta(a \mid s)$ эквивалентно сэмплированию шума из некоторого не зависящего от параметров распределения $\varepsilon \sim p(\varepsilon)$ и его дальнейшего детерминированного преобразования $a = f_\theta(s, \varepsilon)$.*

Обучение стохастических политик

Пример 1: Пусть наша стратегия параметризована нормальным распределением:

$$\pi_\theta(a \mid s) \coloneqq \mathcal{N}(\mu_\theta(s), \sigma_\theta(s)^2 I).$$

Тогда для неё применим репараметризационный трюк: сэмплирование действий эквивалентно $a \coloneqq \mu_\theta(s) + \varepsilon \odot \sigma_\theta(s)$, где $\varepsilon \sim \mathcal{N}(0, I)$, $\odot$ — поэлементное перемножение.

Пример

Семейство детерминированных стратегий $\pi_\theta(s)$ тоже можно считать таким «вырожденным» примером параметризаций, для которой можно проворачивать репараметризационный трюк: просто шум ε считаем «пустым».

Теорема

Если для $\pi_\theta(a \mid s)$ применим репараметризационный трюк, то:

$$\nabla_\theta J(\pi_\theta) = \frac{1}{1-\gamma} \mathbb{E}_{d_{\pi_\theta}(s)} \mathbb{E}_{\varepsilon \sim p(\varepsilon)} \nabla_\theta f_\theta(s, \varepsilon) \; \nabla_a Q^\pi(s, a)\Big|_{a=f_\theta(s,\varepsilon)} . \tag{3}$$

Доказательство. Полностью полностью повторяет вывод теоремы 1. ∎

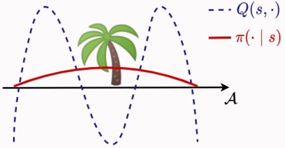
Таким образом, все идеи DDPG расширяются на этот случай. Убирая из формулы градиента частоты посещения состояний и переходя к policy iteration схеме, получаем следующий функционал:

$$\mathbb{E}_s \mathbb{E}_{a \sim \pi_\theta(a|s)} Q_\omega(s, a) \to \max_\theta, \tag{4}$$

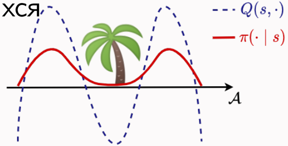
где состояния берутся из буфера. В силу репараметризационного трюка мы легко справимся со взятием градиента на практике:

$$\nabla_\theta \mathbb{E}_{a \sim \pi_\theta(a|s)} Q_\omega(s, a) = \mathbb{E}_{\varepsilon \sim p(\varepsilon)} \nabla_\theta Q_\omega(s, f_\theta(s, \varepsilon)).$$

Пример 2: Представьте, что вы хотите объехать дерево. Вы можете объехать его справа, можете слева. Критик сообщает вам высокие значения и слева, и справа, и оптимизируя (4) в классе гауссиан, можно получить стратегию, которая с наибольшей вероятностью выбирает действие «врезаться в дерево».



Пример 3: Например, можно использовать смесь гауссиан. Тогда при использовании K компонент смеси актёр для данного состояния s выдаёт следующие величины: K суммирующихся в единицу чисел $w_i(s,\theta)$, а также K векторов $\mu_i(s,\theta), \sigma_i(s,\theta)$, где $i \in 1, 2, \dots, K$. Итоговое распределение полагается



$$\pi_\theta(a \mid s) := \sum_{i=1}^{K} w_i(s,\theta)\mathcal{N}(\mu_i(s,\theta), \sigma_i(s,\theta)^2 I).$$

Утверждение

Градиент (4) *по параметрам актёра* θ *равен*

$$\nabla_\theta = \mathbb{E}_{a \sim \pi_\theta(a|s)} \nabla_\theta \log \pi_\theta(a \mid s) \left(Q_\omega(s, a)(s, a) - b(s)\right),$$

где $b(s)$ *— бэйзлайн, произвольная функция от состояний.*

Теорема (Формула замены переменной в плотности)

Пусть $\pi_\theta(a \mid s) \coloneqq g(u)$, *где* $g\colon \mathbb{R}^A \to \mathbb{R}^A$, *и* $u \sim \mu_\theta(u \mid s)$. *Тогда:*

$$\log \pi_\theta(a \mid s) = \log \mu_\theta(a \mid s) - \log|\det \nabla_u g|. \tag{5}$$

Без доказательства. ■

Определение

Энтропией распределения $\pi(a)$ *называется*

$$\mathcal{H}(\pi(a)) \coloneqq -\mathbb{E}_{\pi(a)} \log \pi(a). \tag{6}$$

Пример

Например, для функции $g(u) := \tanh(u)$ *якобиан* $\nabla_u g$ *есть диагональная матрица (поскольку преобразование поэлементное), и его определитель равен покомпонентному произведению:*

$$\det \nabla_u g(u) = \prod_{i=1}^{A} \nabla_{u_i} \tanh(u_i) =$$

$$= \{\textit{производная гиперболического тангенса}\} = \prod_{i=1}^{A} (1 - \tanh^2(u_i))$$

Заметим, что все компоненты положительные ($\tanh(u_i) \leq 1$)*, поэтому модуль из формулы* (5) *брать не нужно. Подставляя в* (5)*, получаем окончательно:*

$$\log \pi_\theta(a \mid s) = \log \mu_\theta(a \mid s) - \sum_{i=1}^{A} \log(1 - \tanh^2(u_i)).$$

Soft Actor-Critic

Определение

*Задачей **Maximum Entropy RL** является максимизация функционала*

$$J_{\text{soft}}(\pi) := \mathbb{E}_{\mathcal{T}\sim\pi} \sum_{t\geq 0} \gamma^t \left[r_t + \alpha\mathcal{H}(\pi(\cdot \mid s))\right] \to \max_{\pi}, \quad (7)$$

*где α — гиперпараметр, называемый **температурой** (temperature).*

Soft Actor-Critic

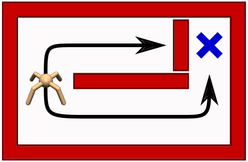
Утверждение

Задача (7) эквивалентна

$$J_{\mathrm{soft}}(\pi) := \mathbb{E}_{\mathcal{T}\sim\pi} \sum_{t\geq 0} \gamma^t \left[r_t - \alpha \log \pi(a_t \mid s_t)\right] \to \max_{\pi},$$

Доказательство. Следует из определения энтропии (6); ведь мат.ожидание по действиям присутствует в мат.ожидании по траекториям. ■

Пример 4: Представим, что у агента есть два пути, и по мере углубления награда на каждом пути одинаково растёт. Пусть первый путь заканчивается тупиком и суммарно позволяет набрать не более +100, а на втором тупик стоит чуть дальше и даёт +110. Во время обучения агент может уловить награду вдоль первого пути и учиться углубляться в него, игнорируя исследование второго пути, даже если агент умеет набирать там награду как на первом; за счёт бонуса за наиболее стохастичную стратегию агент мотивирован в течение обучения в начале эпизодов случайно выбирать между двумя путями. То есть, энтропийный бонус помогает избегать подобных «локальных максимумов» в среде.



Можно считать, что в данном фреймворке мы на самом деле лишь чуть-чуть модифицировали награду в среде:

$$r_{\text{soft}}(s, a) := r(s, a) - \alpha \log \pi(a \mid s). \tag{8}$$

Утверждение

Оптимальной детерминированной стратегии может не существовать.

Доказательство. В MDP, где награда всегда 0, оптимальна стратегия с максимальной энтропией. ∎

Теорема (Мягкие уравнения связи)

$$Q^\pi_{\text{soft}}(s, a) = r(s, a) + \gamma \mathbb{E}_{s'} V^\pi_{\text{soft}}(s') \tag{9}$$

$$V^\pi_{\text{soft}}(s) = \mathbb{E}_{\pi(a|s)} \left[Q^\pi_{\text{soft}}(s, a) - \log \pi(a \mid s) \right] \tag{10}$$

Доказательство. *По определению с учётом договорённости.* ∎

Теорема (Мягкие уравнения Беллмана (soft Bellman equations))

$$Q^\pi_{\text{soft}}(s, a) = r(s, a) + \gamma \mathbb{E}_{s'} \mathbb{E}_{a'} \left[Q^\pi_{\text{soft}}(s', a') - \log \pi(a' \mid s') \right] \tag{11}$$

$$V^\pi_{\text{soft}}(s) = \mathbb{E}_{a} \left[r(s, a) - \log \pi(a \mid s) + \gamma \mathbb{E}_{s'} V^\pi_{\text{soft}}(s') \right] \tag{12}$$

Теорема

Операторы, стоящие в правой части мягких уравнений Беллмана, являются сжимающими с коэффициентом γ по метрике

$$\rho(V_1, V_2) := \max_s |V_1(s) - V_2(s)|.$$

Доказательство. Покажем для мягкой V-функции. Пусть $\mathfrak{B}_{\mathrm{soft}}$ — оператор, стоящий в правой части (12), и пусть даны две V-функции V_1, V_2. Тогда:

$$|[\mathfrak{B}_{\mathrm{soft}} V_1](s) - [\mathfrak{B}_{\mathrm{soft}} V_2](s)| = \gamma |\mathbb{E}_a \mathbb{E}_{s'} \left[V_1(s') - V_2(s') \right] |,$$

поскольку энтропия стратегии π вместе с наградой за шаг одинакова для V_1 и V_2 и потому сокращается. Дальше, как и для обычных V-функций, можно просто оценить это выражение сверху $\gamma\rho(V_1, V_2)$, заканчивая доказательство. ∎

Утверждение

Метод простой итерации сходится к единственному решению мягких уравнений Беллмана из любого начального приближения.

Итак, мы уже можем сразу построить процедуру обучения критика. Рассмотрим обучение $Q_\omega(s, a) \approx Q^\pi_{\text{soft}}(s, a)$ с одношаговых оценок в off-policy режиме, то есть будем просто решать мягкое уравнение Беллмана (11). Тогда для заданного перехода $\mathbb{T} = (s, a, r, s')$ целевая переменная строится как

$$y(\mathbb{T}) := r + \gamma \mathbb{E}_{a' \sim \pi(a' | s')} \left[Q_\omega(s', a') - \log \pi(a' \mid s') \right].$$

В таком варианте таргеты для критика (Q-функция с параметрами ω) и для вспомогательной V-функции выглядят следующим образом: для заданного перехода $\mathbb{T} = (s, a, r, s')$, взятого из буфера, генерируем a_π из текущей версии стратегии π и запоминаем вероятность $\pi(a_\pi \mid s)$, после чего вычисляем несмещённые оценки правых частей уравнений связи (10) и (9):

$$y_Q(\mathbb{T}) := r + \gamma V_\psi(s'),$$

$$y_V(\mathbb{T}) := Q_\omega(s, a_\pi) - \log \pi(a_\pi \mid s).$$

Теорема 2 — Soft Policy Improvement: Пусть стратегии π_1 и π_2 таковы, что для всех состояний s выполняется:

$$\mathbb{E}_{\pi_2(a|s)} Q^{\pi_1}_{\text{soft}}(s,a) + \mathcal{H}(\pi_2(\cdot \mid s)) \geq V^{\pi_1}_{\text{soft}}(s),$$

тогда $\pi_2 \succeq \pi_1$ с учётом энтропийного бонуса; если хотя бы для одного s неравенство выполнено строго, то $\pi_2 \succ \pi_1$.

Доказательство. Полностью аналогично доказательству в обычном случае. Покажем, что $V^{\pi_2}_{\text{soft}}(s) \ge V^{\pi_1}_{\text{soft}}(s)$ для любого s:

$$\begin{aligned}
V^{\pi_1}_{\text{soft}}(s) \le \{\text{по построению } \pi_2\} &\le \mathbb{E}_{\pi_2(a|s)} Q^{\pi_1}_{\text{soft}}(s,a) + \mathcal{H}(\pi_2(\cdot \mid s)) = \\
= \{\text{связь QV (9)}\} &= \mathbb{E}_{\pi_2(a|s)} \left[r + \mathcal{H}(\pi_2(\cdot \mid s)) + \gamma \mathbb{E}_{s'} V^{\pi_1}_{\text{soft}}(s') \right] \le \\
\le \{\text{применяем это же неравенство рекурсивно}\} &= \mathbb{E}_{\pi_2(a|s)} [r + \mathcal{H}(\pi_2(\cdot \mid s)) + \\
&+ \mathbb{E}_{s'} \mathbb{E}_{\pi_2(a'|s')} \left[\gamma r' + \gamma \mathcal{H}(\pi_2(\cdot \mid s')) + \gamma^2 \mathbb{E}_{s''} V^{\pi_1}_{\text{soft}}(s'') \right]] \le \\
\le \{\text{раскручиваем цепочку далее}\} &\le \ldots \le \mathbb{E}_{\mathcal{T} \sim \pi_2 | s_0 = s} \sum_{t \ge 0} \gamma^t r_t + \gamma^t \mathcal{H}(\pi_2(\cdot \mid s_t)) = \\
= \{\text{по определению мягкой V-функции}\} &= V^{\pi_2}_{\text{soft}}(s)
\end{aligned}$$

Если для какого-то s неравенство из условия теоремы было выполнено строго, то для него первое неравенство в этой цепочке рассуждений выполняется строго, и, значит, $V^{\pi_2}_{\text{soft}}(s) > V^{\pi_1}_{\text{soft}}(s)$. ∎

Таким образом, аналог общей схемы Generalized Policy Iteration в задаче Maximum Entropy RL выглядит так:

- **Soft Policy Evaluation** заключается в обучении аппроксимации мягкой оценочной функции $Q_\omega \approx Q^\pi_{\text{soft}}$ для текущей стратегии π;
- **Soft Policy Improvement** заключается в оптимизации следующего функционала (для разных состояний s):

$$\mathbb{E}_{\pi(a|s)} Q_\omega(s, a) + \mathcal{H}(\pi(\cdot \mid s)) \to \max_\pi . \tag{13}$$

Теорема 3: Задача (13) эквивалентна задаче

$$\mathrm{KL}(\pi_\theta(\cdot \mid s) \parallel \exp Q_\omega(s, \cdot)) \to \min_\theta,$$

где $\exp Q_\omega(s, \cdot)$ — ненормированное распределение над действиями.

Доказательство. Обозначим нормировочную константу распределения $\exp Q_\omega(s, \cdot)$ как $Z_\omega(s) := \int\limits_{\mathcal{A}} \exp Q\omega(s, a) \,\mathrm{d}a$. Тогда:

$$\mathrm{KL}(\pi_\theta(\cdot \mid s) \parallel \exp Q_\omega(s, \cdot)) = \mathbb{E}_{a \sim \pi_\theta(a|s)} \log \pi_\theta(a \mid s) - Q_\omega(s, a) + \log Z_\omega(s).$$

Осталось заметить, что при домножении на минус получим (13): первое слагаемое есть энтропия, а третье слагаемое не зависит от оптимизируемых параметров θ:

$$\mathbb{E}_{a \sim \pi_\theta(a|s)} \log Z_\omega(s) = \log Z_\omega(s) = \mathrm{const}(\theta). \qquad \blacksquare$$

Теорема (Вид жадной стратегии)

Максимальное значение (13) достигается на стратегии

$$\pi(a \mid s) \propto \exp Q_\omega(s, a). \tag{14}$$

Доказательство. Следует из теоремы 3, поскольку минимум KL-дивергенции достигается в нуле на совпадающих распределениях. ∎

Пример 5: Пусть наша стратегия параметризована гауссианой (см. пример 1). Для неё можно проводить репараметризационный трюк и также можно аналитически посчитать энтропию:

$$\mathcal{H}(\mathcal{N}(\mu, \sigma^2 I)) = \sum_{i=1}^{A} \log \sigma_i .$$

Итого, формула (13) в таком случае получается следующей:

$$\mathbb{E}_{\varepsilon \sim \mathcal{N}(0, I)} Q_\omega(s, \mu_\theta(s) + \sigma_\theta(s) \odot \varepsilon) + \sum_{i=1}^{A} \log \sigma_i(s, \theta) \to \max_\theta .$$

Пример 6: Пусть наша стратегия параметризована смесью гауссиан (см. пример 3). Тогда для неё не применим репараметризационный трюк, и сложно аналитически посчитать энтропию. Тогда придётся применять REINFORCE, и формула градиента (13) получается следующей:

$$\nabla_\theta = \mathbb{E}_{a \sim \pi_\theta(a|s)} \nabla_\theta \log \pi_\theta(a \mid s) \left[Q_\omega(s, a) - \log \pi_\theta(a \mid s) - b(s) \right],$$

где $b(s)$ — бэйзлайн, произвольная функция от состояний. Имеет смысл делать её близкой к среднему значению $Q_\omega(s, a) - \log \pi_\theta(a \mid s)$, то есть хорошо выбирать $b(s) := V_\omega(s) = \mathbb{E}_a \left[Q_\omega(s, a) - \log \pi_\theta(a \mid s) \right]$.

Алгоритм 3: Soft Actor-Critic (SAC)

Гиперпараметры: B — размер мини-батчей, β — параметр экспоненциального сглаживания таргет-сети, α — температура, $\pi_\theta(a \mid s) := \mathcal{N}(\mu_\theta(s), \sigma_\theta(s)^2 I)$ — гауссова стратегия с параметрами θ, $Q_{\omega_1}(s,a), Q_{\omega_2}(s,a)$ — две нейросетки с параметрами ω_1 и ω_2, $V_\psi(s)$ — нейросетка с параметрами ψ, SGD-оптимизаторы.

Инициализировать $\theta, \omega_1, \omega_2, \psi$ произвольно

Инициализировать таргет-сеть $\psi^- := \psi$

Пронаблюдать s_0

На очередном шаге t**:**

1. выбрать $a_t \sim \pi_\theta(a_t \mid s_t)$
2. пронаблюдать r_t, s_{t+1}, done_{t+1}
3. добавить пятёрку $(s_t, a_t, r_t, s_{t+1}, \mathrm{done}_{t+1})$ в реплей буфер
4. засэмплировать мини-батч размера B из буфера
5. для каждого s из батча засэмплировать шума $\varepsilon(s) \sim \mathcal{N}(0, I)$ и посчитать $\mu(s, \theta)$, $\sigma(s, \theta)$ стратегии π_θ

6 посчитать оценку градиента по параметрам стратегии:

$$\nabla_\theta := \frac{1}{B}\sum_{s \in B} \nabla_\theta \left[\alpha \sum_{i=1}^{A} \log \sigma_i(s,\theta) + \min_{i=1,2} Q_{\omega_i}(s, \mu_\theta(s) + \sigma_\theta(s) \odot \varepsilon(s))\right]$$

7 делаем шаг градиентного подъёма по θ, используя ∇_θ

8 для каждого перехода $\mathbb{T} := (s, a, r, s', \mathrm{done})$ засэмплировать $a_\pi \sim \pi_\theta(a_\pi \mid s)$ и сохранить вероятности $\pi_\theta(a_\pi \mid s)$

9 посчитать таргеты:

$$y_V(\mathbb{T}) := \min_{i=1,2} Q_{\omega_i}(s, a_\pi) - \alpha \log \pi_\theta(a_\pi \mid s)$$

$$y_Q(\mathbb{T}) := r + \gamma V_{\psi^-}(s')$$

10 посчитать лоссы:

$$\text{Loss}_V(\psi) := \frac{1}{B} \sum_{\mathbb{T}} \left(V_\psi(s') - y_V(\mathbb{T}) \right)^2$$

$$\text{Loss}_{Q1}(\omega_1) := \frac{1}{B} \sum_{\mathbb{T}} \left(Q_{\omega_1}(s, a) - y_Q(\mathbb{T}) \right)^2$$

$$\text{Loss}_{Q2}(\omega_2) := \frac{1}{B} \sum_{\mathbb{T}} \left(Q_{\omega_2}(s, a) - y_Q(\mathbb{T}) \right)^2$$

11 делаем шаг градиентного спуска по ψ, ω_1 и ω_2, используя $\nabla_\psi \text{Loss}_V(\psi)$, $\nabla_\omega \text{Loss}_{Q1}(\omega_1)$ и $\nabla_{\omega_2} \text{Loss}_{Q2}(\omega_2)$ соответственно

12 обновляем таргет-сеть: $\psi^- \leftarrow (1 - \beta)\psi^- + \beta\psi$

Сформулируем критерий оптимальности Беллмана в задаче Maximum Entropy RL. По аналогии с обычным случаем, введём оптимальные оценочные функции:

$$V^*_{\text{soft}}(s) \coloneqq \max_\pi V^\pi_{\text{soft}}(s)$$

$$Q^*_{\text{soft}}(s, a) \coloneqq \max_\pi Q^\pi_{\text{soft}}(s, a)$$

Теорема (Вид оптимальной стратегии для Maximum Entropy RL)

Оптимальной является единственная стратегия

$$\pi(a \mid s) \propto \exp Q^*_{\text{soft}}(s, a)$$

Доказательство. Откажемся от стационарности и будем рассматривать задачу поиска оптимальной стратегии $\pi_t(a \mid s_0)$ в предположении, что в будущем мы сможем набрать максимально возможную награду $Q^*_{\text{soft}}(s, a, t)$:

$$\mathbb{E}_{\pi_t(a|s)} \left[Q^*_{\text{soft}}(s, a, t) - \log \pi_t(a \mid s) \right] \to \max_{\pi_t(a|s)}$$

Аналогично теореме 3 про soft policy improvement, можно заметить, что с точностью до константы, не зависящей от π_t, оптимизируемое выражение есть:

$$-\operatorname{KL}\left(\pi_t(a \mid s) \parallel \frac{\exp Q^*_{\text{soft}}(s, a, t)}{Z(s, t)} \right) \to \max_{\pi_t(a|s)},$$

где $Z(s, t)$ — нормировочная константа $\exp Q^*_{\text{soft}}(s, a, t)$. Понятно, что оптимум достигается в нуле на $\pi_t(a \mid s)$, совпадающей с этим распределением.

Дальнейшее рассуждение строится как раньше: Q^*_{soft} от времени не зависит по определению, поэтому оптимальная стратегия получается стационарной, следовательно

$$\pi(a \mid s) \propto \exp Q^*_{\mathrm{soft}}(s, a)$$

Заметим, что в силу однозначного определения $Q^*_{\mathrm{soft}}(s, a)$, такая стратегия в принципе единственна в отличие от обычной задачи RL.

■

Теорема

$$V^*_{\mathrm{soft}}(s) = \log \int\limits_{\mathcal{A}} \exp Q^*_{\mathrm{soft}}(s, a)\, \mathrm{d}a \qquad (15)$$

Доказательство. Мы знаем, что оптимальная стратегия имеет вид $\pi^*(a \mid s) = \frac{\exp Q^*_{\mathrm{soft}}(s,a)}{Z(s)}$, где

$$Z(s) := \int\limits_{\mathcal{A}} \exp Q^*_{\mathrm{soft}}(s,a)\,\mathrm{d}a$$

является нормировочной константой. Посчитаем энтропию такого распределения:

$$-\mathbb{E}_{\pi^*(a|s)} \log \pi^*(a \mid s) = \int\limits_{\mathcal{A}} \frac{\exp Q^*_{\mathrm{soft}}(s,a)}{Z(s)} \left(\log Z(s) - Q^*_{\mathrm{soft}}(s,a)\right)\mathrm{d}a =$$
$$= \log Z(s) - \mathbb{E}_{\pi^*(a|s)} Q^*_{\mathrm{soft}}(s,a).$$

Подставим оптимальную стратегию в мягкое VQ уравнение (10), которое справедливо в том числе и для оптимальной стратегии:

$$
\begin{aligned}
V^*_{\text{soft}}(s) &= \mathbb{E}_{\pi^*(a|s)}\left[Q^*_{\text{soft}}(s, a) - \log \pi^*(a \mid s)\right] = \\
&= \mathbb{E}_{\pi^*(a|s)} Q^*_{\text{soft}}(s, a) - \mathbb{E}_{\pi^*(a|s)} Q^*_{\text{soft}}(s, a) + \log Z(s) = \\
&= \log Z(s)
\end{aligned}
$$

Вспоминая определение $Z(s)$, получаем доказываемое.

$\blacksquare$

Утверждение

$$Q^*_{\mathrm{soft}}(s, a) = r(s, a) + \gamma \mathbb{E}_{s'} V^*_{\mathrm{soft}}(s')$$

Утверждение (Мягкое уравнение оптимальности Беллмана (soft Bellman optimality equations))

$$Q^*_{\mathrm{soft}}(s, a) = r(s, a) + \gamma \mathbb{E}_{s'} \log \int_{\mathcal{A}} \exp Q^*_{\mathrm{soft}}(s', a') \,\mathrm{d}a' \tag{16}$$

Теорема

Оператор, стоящий в правой части уравнения (16), является сжимающим с коэффициентом γ, и, следовательно, метод простой итерации решения этой системы уравнений сходится из любого начального приближения к единственной неподвижной точке.

Доказательство. Пусть даны две Q-функции, Q_1, Q_2, и пусть

$$\rho(Q_1, Q_2) := \max_{s,a} |Q_1(s,a) - Q_2(s,a)| < \varepsilon.$$

Тогда:

$$\log \int_{\mathcal{A}} \exp Q_1(s,a)\,\mathrm{d}a \leq \log \int_{\mathcal{A}} \exp(Q_2(s,a) + \varepsilon)\,\mathrm{d}a = \varepsilon + \log \int_{\mathcal{A}} \exp Q_2(s,a)\,\mathrm{d}a.$$

Аналогично можно показать, что

$$\log \int_{\mathcal{A}} \exp Q_1(s,a)\,\mathrm{d}a \geq -\varepsilon + \log \int_{\mathcal{A}} \exp Q_2(s,a)\,\mathrm{d}a$$

Пусть $\mathfrak{B}_{\mathrm{soft}}$ — оператор, стоящий в правой части (16). Тогда:

$$|[\mathfrak{B}_{\mathrm{soft}} Q_1](s,a) - [\mathfrak{B}_{\mathrm{soft}} Q_2](s,a)| =$$

$$= \gamma |\mathbb{E}_{s'} \left(\log \int_{\mathcal{A}} \exp Q_1(s',a')\,\mathrm{d}a' - \log \int_{\mathcal{A}} \exp Q_2(s',a')\,\mathrm{d}a' \right) | \leq \gamma\varepsilon.$$

Таким образом, $\rho(\mathfrak{B}_{\text{soft}} Q_1, \mathfrak{B}_{\text{soft}} Q_2)$ уменьшилось по крайней мере в γ раз по сравнению с $\rho(Q_1, Q_2)$.

■

Утверждение

Если $\pi(a \mid s) \propto \exp Q^{\pi}_{\text{soft}}(s, a)$, то она оптимальна.

Доказательство. Q-функция такой стратегии удовлетворяет мягкому уравнению оптимальности Беллмана (16) и в силу единственности его решения совпадает с $Q^{*}_{\text{soft}}(s, a)$. ■

Теперь можно построить аналог DQN для задачи Maximum Entropy RL, называемым **Soft Q-learning**. В нём нет отдельной модели актёра, и текущая стратегия просто полагается жадной по отношению к текущему критику. Это также можно интерпретировать как моделирование **Soft Value Iteration** — решение мягкого уравнения оптимальности (16). Тогда таргет для перехода $\mathbb{T} := (s, a, r, s')$ строится как

$$y(\mathbb{T}) := r + \gamma \log \int_{\mathcal{A}} \exp Q_{\theta^-}(s', a') \, \mathrm{d}a',$$

где θ^- — параметры таргет-сети.

Теорема (Soft Policy Gradient)

$$\nabla_\theta J_{\text{soft}}(\theta) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{\pi_\theta}(s)}\mathbb{E}_{a\sim\pi_\theta(a|s)}\nabla_\theta \log \pi_\theta(a \mid s) Q^{\pi}_{\text{soft}}(s,a) +$$
$$+ \alpha \nabla_\theta \mathcal{H}(\pi_\theta(\cdot \mid s)).$$

Доказательство. *Аналогично доказательству в обычном случае.* ∎