Model-Based methods

Разворотнев Иван

11.07.2019

Tinkoff.ru

Model-Based методы

Методы основанные на выучивании динамики среды

Достоинства:

Недостатки:

- Модель среды не всегда известна, а ее изучение может быть не эффективно или сложно.

- При большом количестве примеров модельные методы проигрывают model-free

Альтернативная система обозначений

u_t – контроллер

x_t – состояние

c_t – стоимость

a_t – контроллер

s_t – состояние

r_t – вознаграждение

Л. С. Понтрягин

Р. Беллман

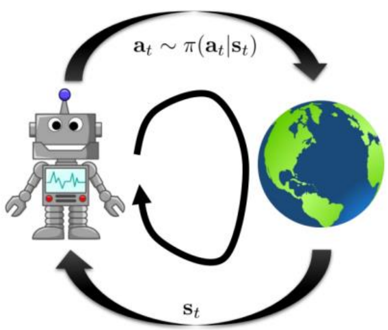
Известная динамика среды

Планирование - выбор оптимальных действий на основе динамики среды

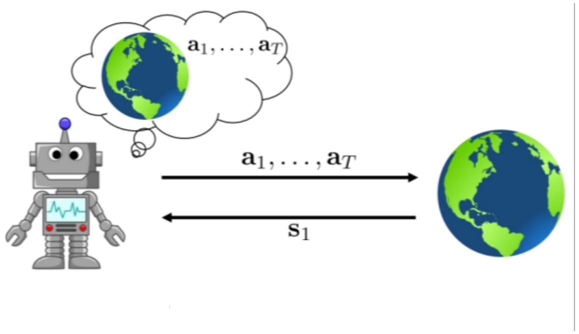
Примеры определенной динамики:
- Правила игры
- Законы физики
- Симуляции

Динамика задается:
$p(s'|s,a)$ и $p(r|s,a)$

Два сценария планирования:



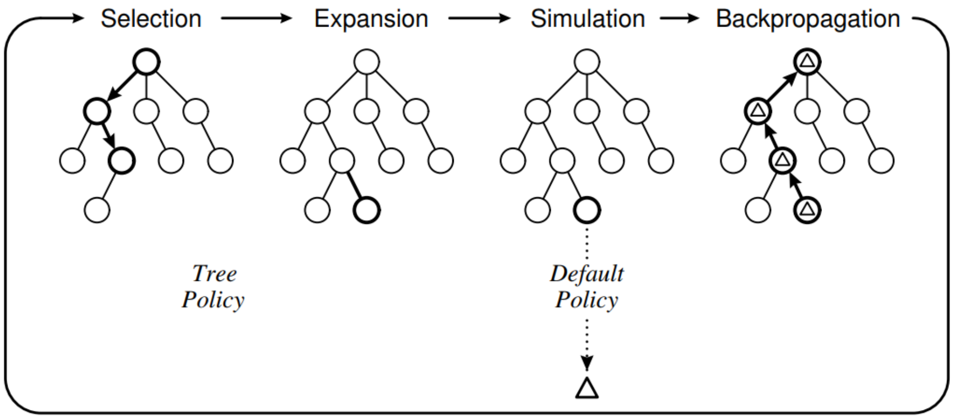
Closed-Loop – планируем действие каждый шаг



Open-Loop – планируем траекторию целиком

Monte Carlo Tree Search(MCTS)

Последовательность:



1. **Выбор**. Проходим по дереву и выбираем оптимальный путь(от вершины до листа).

2. **Расширение**. К вершинам(одной или нескольким), находящемся на этом пути добавляем потомка(делаем новое действие).

3. **Симуляция**. Из добавленного(добавленных) вершин запускаем default политику до завершения эпизода.

4. **Распространение**. Используем реворды, полученные в итоге симуляции и обновляем Q-значения.

Вершины – состояния
Переходы взвешены Q-значениями/среднеми ревордами/…

Tree Policy – политика выбора траектории по дереву
Default Policy – политика для rollout'a

Как выбирать на шаге 1?

UTC:

Если вершина не была полностью расширена – расширяем.
Иначе выберем потомка максимизирующего:

$$\frac{Q(v')}{N(v')} + c\sqrt{\frac{2\ln N(v)}{N(v')}}$$

Линейно-квадратичный регулятор(LQR)

Пусть среда известна и линейна

$$s_{t+1} = f(\boldsymbol{s}_t, \boldsymbol{a}_t) = \boldsymbol{F}_t \begin{bmatrix} \boldsymbol{s}_t \\ \boldsymbol{a}_t \end{bmatrix} + f_t$$

$$r(s_t, a_t) = \frac{1}{2} \begin{bmatrix} \boldsymbol{s}_t \\ \boldsymbol{a}_t \end{bmatrix}^T \boldsymbol{R}_t \begin{bmatrix} \boldsymbol{s}_t \\ \boldsymbol{a}_t \end{bmatrix} + \begin{bmatrix} \boldsymbol{s}_t \\ \boldsymbol{a}_t \end{bmatrix}^T \boldsymbol{r}_t$$

Задача:

$Q(s_t, a_t)$

$$\max_{a_1,\dots,a_T} \sum_{t=1}^{T} r(s_t, a_t), \text{при } s_t = f(s_{t-1}, a_{t-1})$$

Без условия:

$$\max_{a_1,\dots,a_T} r(s_1, a_1) + r(f(s_1, a_1), a_2) + \dots + r(f(f(\dots), \dots), a_t)$$

Идея решения:

1.Обратный проход: Начиная с конца, выражаем оптимальные действия через предыдущие состояния

2. Прямой проход: восстанавливаем последовательность оптимальных действий

LQR. Обратный проход

$$\max_{a_1,\ldots,a_T} r(s_1,a_1) + r(f(s_1,a_1),a_2) + \cdots + \underbrace{r(f(f(\ldots),\ldots),a_t)}_{\text{Найдем оптимальную } a_t \text{ при } s_t}$$

Начинаем с $t = T$

1. Выразим Q от a и s:

$$Q(\boldsymbol{s}_t, \boldsymbol{a}_t) = \frac{1}{2}\begin{bmatrix} s_t \\ a_t \end{bmatrix}^T \boldsymbol{R}_t \begin{bmatrix} s_t \\ a_t \end{bmatrix} + \begin{bmatrix} s_t \\ a_t \end{bmatrix}^T \boldsymbol{r}_t + V(s_t)$$

$$R_t = \begin{bmatrix} R_{s_t s_t} & R_{s_t a_t} \\ R_{a_t s_t} & R_{a_t a_t} \end{bmatrix} \qquad \boldsymbol{r}_t = \begin{bmatrix} \boldsymbol{r}_{s_t} \\ \boldsymbol{r}_{a_t} \end{bmatrix}$$

0 в t=T

2. Найдем a

$$\nabla_{a_t} Q(\boldsymbol{s}_t, \boldsymbol{a}_t) = \boldsymbol{R}_{a_t s_t}\boldsymbol{s}_t + \boldsymbol{R}_{a_t a_t}\boldsymbol{a}_t + \boldsymbol{r}_{a_t}^T = 0$$

$$\mathrm{a}_t = \mathbf{K}_t s_t + \boldsymbol{k}_t$$

$$\mathbf{K}_t = -R_{aa_t}^{-1} R_{as_t}$$

$$k_t = -R_{aa_t}^{-1} r_{a_t}$$

(const)

Зависит только от s_t

3. Найдем V: (Подставим a в Q)

$$V(s_t) = \frac{1}{2} s_t V_t s_t + s_t v_t$$

Зависит только от s_t

$$V_t = R_{ss_t} + R_{sa_t} K_t + K_t^T + K_t^T R_{ax_t} + K_t^T R_{aa_t} K_t$$

$$v_t = r_{s_t} + R_{sa_t} k_t + K_t^T R_{a_t} + K_t^T R_{aa_t} k_t$$

(const)

LQR. Обратный проход

Идем на шаг назад
Для t-1:

4. Выразим V и Q через $s_t = f(s_{t-1}\ a_{t-1})$ (Подставим a в Q f)

$$V(s_t) = \frac{1}{2}\begin{bmatrix} s_{t-1} \\ a_{t-1} \end{bmatrix}^T \boldsymbol{F}_{t-1}^{\boldsymbol{T}} \boldsymbol{V}_t \boldsymbol{F}_{t-1} \begin{bmatrix} s_{t-1} \\ a_{t-1} \end{bmatrix} + \begin{bmatrix} s_{t-1} \\ a_{t-1} \end{bmatrix}^T \boldsymbol{F}_{t-1}^{\boldsymbol{T}} \boldsymbol{V}_t \boldsymbol{f}_{t-1} + \begin{bmatrix} s_{t-1} \\ a_{t-1} \end{bmatrix}^T \boldsymbol{F}_{t-1}^{\boldsymbol{T}} \boldsymbol{V}_t$$

Зависит только от s_{t-1}, u_{t-1}

$$Q(\boldsymbol{s}_{t-1}, \boldsymbol{a}_{t-1}) = \frac{1}{2}\begin{bmatrix} s_{t-1} \\ a_{t-1} \end{bmatrix}^T \boldsymbol{R}_{t-1} \begin{bmatrix} s_{t-1} \\ a_{t-1} \end{bmatrix} + \begin{bmatrix} s_{t-1} \\ a_{t-1} \end{bmatrix}^T \boldsymbol{r}_{t-1} + V(s_t)$$

5. Выразим Q через линейные и квадратичные коэффициенты:

$$Q(\boldsymbol{s}_{t-1}, \boldsymbol{a}_{t-1}) = \frac{1}{2}\begin{bmatrix} s_{t-1} \\ a_{t-1} \end{bmatrix}^T \boldsymbol{Q}_{t-1} \begin{bmatrix} s_{t-1} \\ a_{t-1} \end{bmatrix} + \begin{bmatrix} s_{t-1} \\ a_{t-1} \end{bmatrix}^T \boldsymbol{q}_{t-1}$$

$$\boldsymbol{Q}_{t-1} = \boldsymbol{R}_{t-1} + \boldsymbol{F}_{t-1}^{T} \boldsymbol{V}_t \boldsymbol{F}_{t-1}$$

$$\boldsymbol{q}_{t-1} = \boldsymbol{r}_{t-1} + \boldsymbol{F}_{t-1}^{T} \boldsymbol{V}_t \boldsymbol{f}_{t-1}$$

6. Найдем a

$$\mathrm{a}_{t-1} = \mathbf{K}_{t-1} s_{t-1} + \boldsymbol{k}_{t-1}$$

$$\mathbf{K}_{t-1} = Q_{aa_{t-1}}^{-1} Q_{as_{t-1}}$$

$$\boldsymbol{k}_{t-1} = Q_{aa_{t-1}}^{-1} q_{a_{t-1}}$$

Зависит только от s_{t-1}

Повторить для всех t до t=1

LQR. Прямой проход

Восстанавливаем начиная с t=1 a и s

Для всех t от 1 до T:

$$a_t = K_t s_t + k_t$$

$$s_{t+1} = f(s_t, a_t)$$

LQR. Стохастическая динамика

Переопределим динамику среды:

$$s_{t+1} = f(\boldsymbol{s}_t, \boldsymbol{a}_t) = \boldsymbol{F}_t \begin{bmatrix} \boldsymbol{s}_t \\ \boldsymbol{a}_t \end{bmatrix} + f_t$$

$$s_{t+1} \sim p(\boldsymbol{s}_{t+1} | \boldsymbol{s}_t, \boldsymbol{a}_t) = \mathcal{N}\left(\boldsymbol{F}_t \begin{bmatrix} \boldsymbol{s}_t \\ \boldsymbol{a}_t \end{bmatrix} + \boldsymbol{f}_t, \Sigma_t\right)$$

LQR все равно применим

$$\max \sum_{t=1}^{T} E_{(s_t, a_t) \sim p(s_t, a_t)}[r(s_t, a_t)]$$

$$E[X^2] = Var[X] + (E[X])^2$$

$$a_t = K_t s_t + k_t$$

iLQR

iLQR – дополнение LQR для нелинейных моделей среды

Динамика среды задана нелинейной непрерывной функцией

Динамика среды: (Аппроксимация нелинейной системы линейно-квадратичной)

$$f(s_t, a_t) \approx f(\hat{s}_t, \hat{a}_t) + \nabla_{s_t, a_t} f(\hat{s}_t, \hat{a}_t) \begin{bmatrix} s_t - \hat{s}_t \\ a_t - \hat{a}_t \end{bmatrix}$$

$$r(s_t, a_t) \approx r(\hat{s}_t, \hat{a}_t) + \nabla_{s_t, a_t} r(\hat{s}_t, \hat{a}_t) \begin{bmatrix} s_t - \hat{s}_t \\ a_t - \hat{a}_t \end{bmatrix} + \frac{1}{2} \begin{bmatrix} s_t - \hat{s}_t \\ a_t - \hat{a}_t \end{bmatrix}^T \nabla^2_{s_t, a_t} r(\hat{s}_t, \hat{a}_t) \begin{bmatrix} s_t - \hat{s}_t \\ a_t - \hat{a}_t \end{bmatrix}$$

где $(\hat{s}_t, \hat{a}_t)^N{}_{t=1}$ - траектория

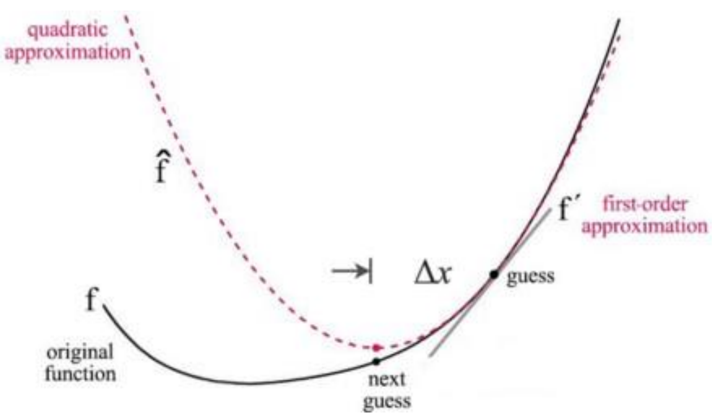
Метод Ньютона для $\min\limits_x g(x)$:

До сходимости:

$$g = \nabla_x g(\hat{x})$$

$$H = \nabla_x^2 g(\hat{x})$$

$$\hat{x} \leftarrow argmin_x \frac{1}{2}(x - \hat{x})^T H (x - \hat{x}) + g^T (x - \hat{x})$$



iLQR

Определим линейную среду:

$$\bar{f}(\delta s_t, \delta a_t) = F_t \begin{bmatrix} \delta s_t \\ \delta a_t \end{bmatrix}$$

$$\nabla_{x_t, a_t} f(\hat{s}_t, \hat{a}_t)$$

$$\bar{r}(\delta s_t, \delta a_t) = r_t \begin{bmatrix} \delta s_t \\ \delta a_t \end{bmatrix} + \frac{1}{2} \begin{bmatrix} \delta s_t \\ \delta a_t \end{bmatrix}^T C_t \begin{bmatrix} \delta s_t \\ \delta a_t \end{bmatrix}$$

$$\nabla_{x_t, a_t} r(\hat{s}_t, \hat{a}_t) \qquad \nabla^2_{x_t, a_t} r(\hat{s}_t, \hat{a}_t)$$

Тогда, можем применить LQR:

Алгоритм iLQR

До сходимости:

$\mathrm{F_t} = \nabla_{s_t, a_t} f(\hat{s}_t, \hat{a}_t)$

$\mathrm{r_t} = \nabla_{s_t, a_t} r(\hat{s}_t, \hat{a}_t)$

$\mathrm{R_t} = \nabla^2_{s_t, a_t} r(\hat{s}_t, \hat{a}_t)$

Запустить обратный проход LQR для состояний δs_t и действий δa_t

Запустить прямой проход на реальной нелинейной динамике

$a_t = K_t(s_t - \hat{s}_t) + \alpha k_t + \hat{s}_t$

Обновить траекторию

Изучение динамики среды

В случае неизвестной динамики среды,
ее можно изучить и применить модельные методы

Наивный алгоритм выучивания модели

При данной функции награды $r = r(s_t, a_t)$ находим динамику детерминированной среды $s_{t+1} = f(s_t, a_t)$

1. run base policy $\pi_0(\mathbf{a}_t|\mathbf{s}_t)$ (e.g., random policy) to collect $\mathcal{D} = \{(\mathbf{s}, \mathbf{a}, \mathbf{s}')_i\}$
2. learn dynamics model $f(\mathbf{s}, \mathbf{a})$ to minimize $\sum_i \|f(\mathbf{s}_i, \mathbf{a}_i) - \mathbf{s}'_i\|^2$
3. plan through $f(\mathbf{s}, \mathbf{a})$ to choose actions

Часто невозможно посетить многие области в пространстве состояний следуя случайной политике

Итеративное выучивание модели

Корректируем политику согласно модели среды

1. run base policy $\pi_0(\mathbf{a}_t|\mathbf{s}_t)$ (e.g., random policy) to collect $\mathcal{D} = \{(\mathbf{s}, \mathbf{a}, \mathbf{s}')_i\}$
2. learn dynamics model $f(\mathbf{s}, \mathbf{a})$ to minimize $\sum_i \|f(\mathbf{s}_i, \mathbf{a}_i) - \mathbf{s}'_i\|^2$
3. plan through $f(\mathbf{s}, \mathbf{a})$ to choose actions
4. execute those actions and add the resulting data $\{(\mathbf{x}, \mathbf{u}, \mathbf{x}')_j\}$ to $\mathcal{D}$

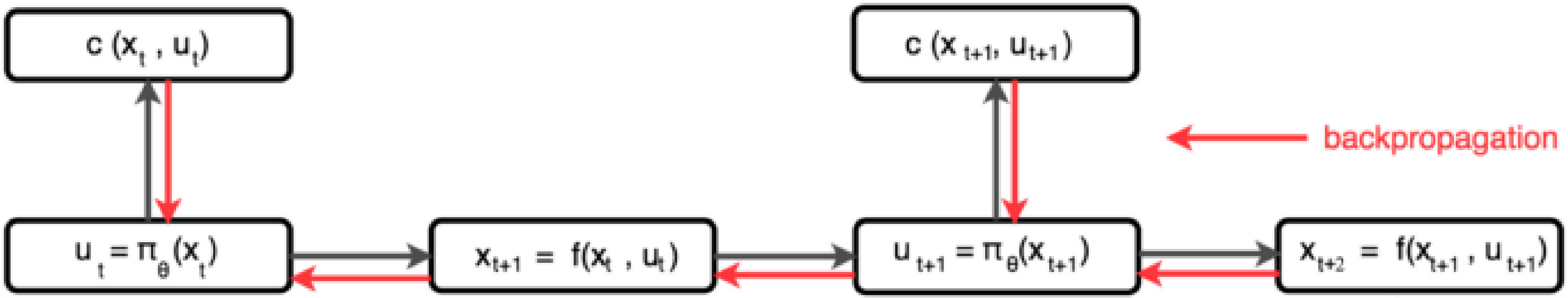
MPC(Model Predictive Control)

Идея:
Выполняем первое из запланированных действий, после чего меняем план

1. run base policy $\pi_0(\mathbf{a}_t|\mathbf{s}_t)$ (e.g., random policy) to collect $\mathcal{D} = \{(\mathbf{s}, \mathbf{a}, \mathbf{s}')_i\}$
2. learn dynamics model $f(\mathbf{s}, \mathbf{a})$ to minimize $\sum_i \|f(\mathbf{s}_i, \mathbf{a}_i) - \mathbf{s}'_i\|^2$
3. plan through $f(\mathbf{s}, \mathbf{a})$ to choose actions
4. execute the first planned action, observe resulting state $\mathbf{s}'$ (MPC)
5. append $(\mathbf{s}, \mathbf{a}, \mathbf{s}')$ to dataset $\mathcal{D}$

Когда модель среды не точна,
план действия из спрогнозированного состояния
может быть некорректным

Обратное распространение ошибки по политике



1. run base policy $\pi_0(\mathbf{a}_t|\mathbf{s}_t)$ (e.g., random policy) to collect $\mathcal{D} = \{(\mathbf{s}, \mathbf{a}, \mathbf{s}')_i\}$
2. learn dynamics model $f(\mathbf{s}, \mathbf{a})$ to minimize $\sum_i \|f(\mathbf{s}_i, \mathbf{a}_i) - \mathbf{s}'_i\|^2$
3. backpropagate through $f(\mathbf{s}, \mathbf{a})$ into the policy to optimize $\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)$
4. run $\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)$, appending the visited tuples $(\mathbf{s}, \mathbf{a}, \mathbf{s}')$ to $\mathcal{D}$

$f(x_i, u_i)$ – коррелируют с $f(x_{i-1}, u_{i-1}), f(x_{i-2}, u_{i-2})$

Взрыв/затухание градиентов

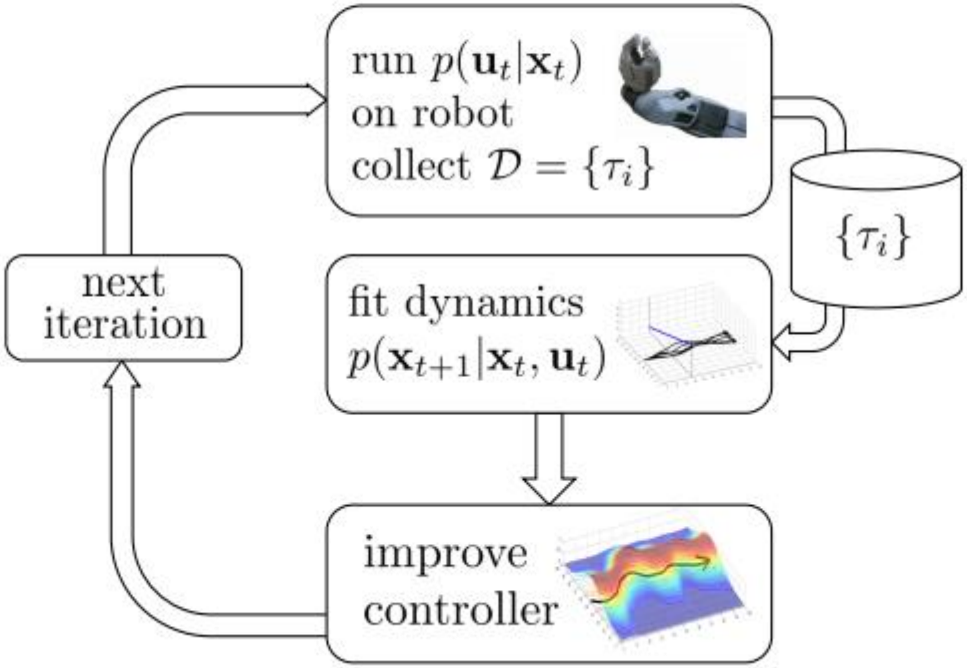
Локальные модели. Линейные модели

Локальная модель – модель определенной окрестности среды

$$p(\mathbf{x}_{t+1}|\mathbf{x}_t, \mathbf{u}_t) = \mathcal{N}(f(\mathbf{x}_t, \mathbf{u}_t), \Sigma)$$

$$f(\mathbf{x}_t, \mathbf{u}_t) \approx \mathbf{A}_t \mathbf{x}_t + \mathbf{B}_t \mathbf{u}_t$$

$$\mathbf{A}_t = \frac{df}{d\mathbf{x}_t} \qquad \mathbf{B}_t = \frac{df}{d\mathbf{u}_t}$$



Обучаем модель среды на траекториях текущей политики

Локальные модели. Политика

Нужно выбрать функцию политики, находящую решения в окрестности траектории, содержащие также “exploration”

$$p(a_t|s_t) = \mathcal{N}(\underbrace{K_t(s_t - \hat{s}_t) + k_t + \hat{a}_t}_{\text{Как в iLQR действуем в окрестности траектории}}, \Sigma_t)$$

Шум для исследования

Эвристика для ковариационной матрицы:

$$\Sigma_t = Q_{a_t a_t}^{-1}$$

“Вклад выбранных действий в вознаграждения”

(Чем меньше влияют действия – тем больше можно рисковать)

Новая цель для максимизации:

$$\max \sum_{t=1}^{T} E_{(s_t,a_t)\sim p(s_t,a_t)}[r(s_t,a_t) + \mathcal{H}(p(s_t,a_t))]$$

“Максимальный суммарный реворд при максимальной энтропии”

KL-дивергенция. Доверительный регион

Доверительный регион:

$$\max_{\pi} \sum_{t=1}^{T} E_{\pi}[r(s_t, a_t)] \ s.t. D_{KL}(\pi(\tau)||\bar{\pi}(\tau)) \ \leq \ \epsilon$$

Контроллер:

$$\pi\ (a_t|s_t) = \mathcal{N}(K_t(s_t - \widehat{s_t}) + k_t + \widehat{a_t}, \Sigma) \qquad \Sigma = Q^{-1}_{a_t, a_t}$$

Ф-ция вознаграждения зависящая от контроллера

KL-дивергенция:

$$D_{KL}(\pi(\tau)||\bar{\pi}(\tau)) = \sum_{t=1}^{T} E_{p(s_t, a_t)}[-\log \bar{\pi}(a_t|s_t) - \mathcal{H}(\pi(a_t|s_t))]$$

Дуальный градиентный спуск

Решаем задачу оптимизации с ограничениями:

$$\min_{\mathbf{x}} f(\mathbf{x}) \text{ s.t. } C(\mathbf{x}) = 0$$

$$\mathcal{L}(\mathbf{x}, \lambda) = f(\mathbf{x}) + \lambda C(\mathbf{x})$$

$$g(\lambda) = \inf_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda)$$

$$\lambda \leftarrow \arg\max_{\lambda} g(\lambda)$$

Для максимизации, посчитаем градиент:

$$\min_{\mathbf{x}} f(\mathbf{x}) \text{ s.t. } C(\mathbf{x}) = 0 \qquad \mathcal{L}(\mathbf{x}, \lambda) = f(\mathbf{x}) + \lambda C(\mathbf{x})$$

$$g(\lambda) = \inf_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda)$$

$$g(\lambda) = \mathcal{L}(\mathbf{x}^\star(\lambda), \lambda)$$

$$\frac{dg}{d\lambda} = \cancel{\frac{d\mathcal{L}}{d\mathbf{x}^\star}} \frac{d\mathbf{x}^\star}{d\lambda} + \frac{d\mathcal{L}}{d\lambda}$$

Если $x^* = argmin_x \mathcal{L}(x,\lambda)$, то $\frac{\mathcal{L}(x,\lambda)}{dx^*} = 0$

Получим такой алгоритм:

$$\min_{\mathbf{x}} f(\mathbf{x}) \text{ s.t. } C(\mathbf{x}) = 0 \qquad \mathcal{L}(\mathbf{x}, \lambda) = f(\mathbf{x}) + \lambda C(\mathbf{x})$$

$$g(\lambda) = \mathcal{L}(\mathbf{x}^\star(\lambda), \lambda)$$

$$\mathbf{x}^\star = \arg\min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda)$$

$$\frac{dg}{d\lambda} = \frac{d\mathcal{L}}{d\lambda}(\mathbf{x}^\star, \lambda)$$

1. Find $\mathbf{x}^\star \leftarrow \arg\min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda)$
2. Compute $\frac{dg}{d\lambda} = \frac{d\mathcal{L}}{d\lambda}(\mathbf{x}^\star, \lambda)$
3. $\lambda \leftarrow \lambda + \alpha \frac{dg}{d\lambda}$

Для нашей задачи:

$$\mathcal{L}(p,\lambda) = \sum_{t=1}^{T} E_{p(s_t,a_t)}\big[r(s_t,a_t) - \overbrace{\lambda \log \bar{\pi}(a_t|s_t) - \lambda \mathcal{H}(\pi(a_t|s_t))\big] - \lambda\epsilon}^{C(x)=(D_{KL}(\pi(\tau)||\bar{\pi}(\tau)) - \epsilon)}$$

1. Найти $p^* \leftarrow argmin_p \mathcal{L}(p\ \lambda)$ } Сложно минимизировать
2. Рассчитать $\frac{dg}{d\lambda} = \frac{d\mathcal{L}}{d\lambda}(p^*, \lambda)$
3. $\lambda \leftarrow \lambda + \alpha \frac{dg}{d\lambda}$

Дуальный градиентный спуск

Проблема:

1. Найти $p^* \leftarrow argmin_p \, \mathcal{L}(p \, \lambda)$

$$\min \sum_{t=1}^{T} E_{p(s_t,a_t)}[r(s_t,a_t) - \lambda \log \bar{\pi}(a_t|s_t) - \lambda \mathcal{H}(\pi(a_t|s_t))] - \lambda \epsilon$$

Гауссовский LQR оптимизирует:

$$\max \sum_{t=1}^{T} E_{(s_t,a_t)\sim p(s_t,a_t)}[r(s_t,a_t) + \mathcal{H}(p(s_t,a_t))]$$

$$\pi\,(a_t|s_t) = \mathcal{N}(K_t(s_t - \hat{s}_t) + k_t + \hat{a}_t, \Sigma) \qquad \Sigma = Q^{-1}_{a_t,a_t}$$

Новая функция потерь для LQR:

$$\tilde{r}(s_t,a_t) = \frac{1}{\lambda} r(s_t,a_t) - \log \bar{\pi}(a_t|s_t)$$

Получим алгоритм:

1. $\tilde{r}(s_t,a_t) = \frac{1}{\lambda} r(s_t,a_t) - \log \bar{\pi}(a_t|s_t)$
2. Используя LQR найти $\pi^*\,(a_t|s_t)$ используя $\tilde{r}(s_t,a_t)$
3. $\lambda \leftarrow \lambda + \alpha(D_{KL}(\pi(\tau)\,\Big|\Big|\,\bar{\pi}(\tau)) - \epsilon)$

Материалы

- RL. Беркли. Планирование
- RL. Беркли. Изучение
- Обзор MCTS методов

Конец

Спасибо за внимание