Федеральное государственное автономное образовательное учреждение
высшего образования
«Московский физико-технический институт
(национальный исследовательский университет)»
Физтех-школа Прикладной Математики и Информатики
Научно-образовательный центр когнитивного моделирования

Направление подготовки / специальность: 03.04.01 Прикладные математика и физика
Направленность (профиль) подготовки: Методы и технологии искусственного интеллекта

МЕТОДЫ ГЕНЕРАЦИИ ПОДЦЕЛЕЙ В МУЛЬТИМОДАЛЬНОМ ОБУЧЕНИИ С ПОДКРЕПЛЕНИЕМ

(магистерская диссертация)

Студент:
Шиман Нуцу

_____
(подпись студента)

Научный руководитель:
Панов Александр Игоревич,
канд. физ.-мат. наук, доц.

_____
(подпись научного руководителя)

Консультант *(при наличии)*:
_____
_____
(подпись консультанта)

Москва 2024

Содержание

1 Введение **4**

2 Основные понятия и определения **7**

2.1 Обучение с подкреплением 7

2.2 Теория информации в обучении с подкреплением 8

2.3 Обучение с подкреплением на основе модели мира 9

2.4 Внутренняя мотивация 12

3 Обзор литературы **13**

3.1 Ошибка предсказания 13

3.2 Новизна состояний 14

3.3 Приращение информации 15

4 Формальная постановка задачи **16**

5 Предложенные методы и вычислительные эксперименты **18**

5.1 Среды 18

5.1.1 Pointmaze 18

5.1.2 Antmaze 18

5.1.3 Atari 19

5.1.4 Метрики 19

5.2 Поведение разногласия ансамбля 20

5.2.1 Разногласие ансамбля 20

5.2.2 Вычислительные эксперименты 21

5.2.3 Ошибка модели мира 23

5.2.4 Размер категориального состояния 23

5.3 Разногласие ансамбля для категориальной модели мира 25

5.3.1 Допущение независимости распределений 27

5.3.2 Упрощение с помощью неравенства Йенсена 28

5.3.3 Вычислительные эксперименты 29

5.4 Метод BatchDist 30

5.4.1 Вычислительные эксперименты 34

6 Заключение **36**

Аннотация

Задача исследования среды еще не является решенной в обучении с подкреплением (ОсП). Важным направлением, дающим лучшие результаты, являются методы исследования на основе модели мира. Один из лучших предложенных ранее алгоритмов берет в качестве внутреннего вознаграждения разногласие ансамбля моделей мира - дисперсию предсказаний моделей. Этот метод изначально был предложен для гауссовской модели мира, однако позже была предложена категориальная модель мира, которая на задаче ОсП показала себя лучше. В данной работе рассматривается поведение разногласия ансамбля на разных моделях мира в разнообразных средах. Оказывается, что результаты заметно хуже для категориальной модели мира. В данной работе предлагаются новые методы подсчета разногласия ансамбля, выведенные из задачи максимизации взаимной информации между параметрами модели мира и следующим состоянием среды. Также предлагается новый метод подсчета внутреннего вознаграждения, основанный на оценке среднего расстояния до ранее посещенных состояний. Несмотря на простоту метода и меньшие затраты по памяти и времени обуения, предложенный метод показывает конкурентные с другими методами результаты.

1 Введение

Методы глубокого обучения с подкреплением (ОсП) рассматриваются, как возможное решение проблемы принятия решений для решения сложных задач, и в некоторых задачах удалось достичь сильных результатов. Например, алгоритмы ОсП использовались для решения видеоигр, в некоторых из них улучшая результаты человека [4]. Также удалось обучить агентов, значительно превосходящих человека, в играх требущих сложные стратегические и тактические действия, например го и шахматы [6], и в многоагентных видеоиграх требующих координации агентов внутри команды [16]. Однако, несмотря на это, области применения алгоритмов ОсП остаются ограниченными. Это связано с некоторыми проблемами: проблема исследования среды в задачах с разряженной функцией вознаграждения, проблема переиспользования опыта для решения разных задач в одной среде, проблема неэффективности по количеству шагов в среде и многие другие.

Методы исследования среды являются важным элементом агентов ОсП. Сигнал, оценивающий перспективность состояния с точки зрения исследования, называется внутренней мотивацией. При максимизации внешнего вознаграждения этот сигнал позволяет направлять агента в состояния среды, которые еще не изучены, но могут приносить вознаграждение. Для сред, в которых вознаграждение разрежено, исследование среды является особо важным для нахождения состояний среды с ненулевым вознаграждением. В таком случае зачастую внутренняя мотивация является единственным сигналом для обучения агента. Также важным применением методов исследования является обучение с подкреплением без учителя (ОсПбУ). В такой постановке агент обучается решать определенную задачу за небольшое количество шагов в среде (фаза дообучения) только после предобучения, которое происходит в среде без вознаграждения. Таким образом информация, полученная алгоритмом исследования во время предобучения, является единственной для обучения некоторых компонент агента, например модели мира или навыков, которые будут использованы при дообучении.

Важной компонентой для эффективного исследования среды является модель мира. Модель мира обучается для предсказания динамики переходов среды, то есть для предсказания следующего состояния среды по текущему состоянию и действию. Обученную модель мира можно использовать для обучения агентов в ней, а не в оригинальной среде, т.е. вместо взаимодействия со средой траектории для обуче-

ния сэмплируются с помощью модели мира. При исследовании среды это позволяет решить проблему ретроспективности, которая заключается в том, что для того, чтобы агент узнал, что у некоторого состояния среды изменилось значение внутренней мотивации, это состояние необходимо посетить. То есть агенту приходится лишний раз посещать некоторые состояния, что замедляет процесс исследования. В свою очередь, модель мира позволяет обнаружить изменение значения внутренней мотивации без взаимодействия со средой. Также модель мира является важной компонентой для ОсПбУ, так как на фазе дообучения она позволяет обучить агента за малое количество взаимодействий со средой или вовсе без таких взаимодействий.

Один из наиболее эффективных методов исследования (plan2explore [21]) использует в качестве внутреннего вознаграждения дисперсию предсказаний ансамбля моделей мира. Чем больше дисперсия в некотором состоянии, тем менее это состояние изучено, так как на изученных состояниях предсказания всех моделей в ансамбле стремятся к истинной динамике среды. Для обучения стратегии исследования этот метод используется модель мира Dreamer [15]. Однако оригинальный метод был предложен для первой версии алгоритма Dreamer. При этом в дальнейшем были разработыны новые версии алгоритма [20, 22], которые показывают лучшие результаты. Важным различием является то, что в первой версии модель мира в качестве распределения предсказывающего вероятность следующего состояния использует гауссовское распределение, а в последующих версиях используется вектор из категориальных распределений.

В данной работе рассматривается различие поведения алгоритма plan2explore на гауссовой модели мира и на категориальной. Было обнаружено, что, несмотря на то, что некоторые методы используют plan2explore с категориальной моделью мира [23], на многих средах алгоритм с категориальной моделью ведет себя хуже, а на некоторых средах значительно хуже, чем алгоритм с гауссовой моделью. Внутреннее вознаграждение в виде дисперсии предсказаний моделей ансамбля изначально было выведено из задачи максимизации взаимной информации между следующим состоянием среды и параметрами модели мира [21]. Однако этот вывод предполагал, что модели предсказывают вероятность следующего состояния в виде гауссовского распределения. Учитывая это и результаты экспериментов, предлагаются новые методы подсчета разногласия моделей в ансамбле для категориальной модели мира, однако эти методы не показали лучших результатов по сравнению оригинальным методом.

Также в данной работе предлагается новый метод подсчета внутреннего возна-

граждения, основанный на аппроксимации среднего расстояния до ранее посещенных состояний (памяти прецедентов) в латентном пространстве модели мира. Использование этого расстояния в качестве внутреннего вознаграждения предполагает движение агента в области среды, которые наиболее отдалены от уже посещенных состояний. Несмотря на простоту и меньшее потребление памяти и времени по сравнению с другими методами, эксперименты показали, что предложенный метод конкурентноспособен с ними.

Все эксперименты в данной работе направлены на оценку эффективности работы алгоритмов исследования, поэтому, чтобы максимально избавиться от сторонних влияний на исследование, рассматривается постановка задачи ОсПбУ. Для оценки исследования используется доля посещенных состояний во время фазы предобучения. В средах, где невозможно получить это значение, используется эффективность агента полученного на фазе дообучения.

В секции 2 вводятся понятия и алгоритмы необходимые для понимания основной части работы. В секции 3 приводится обзор смежных работ. В секции 4 вводится формальная постановка задачи. Секция 5 представляет основные результаты работы и эксперименты. В разделе 5.1 приводятся детали постановки экспериментов. В разделе 5.2 приводятся результаты изучения работы алгоритма plan2explore с категориальной и гауссовской моделями мира. В разделе 5.3 выводятся новые методы подсчета разногласия ансамбля для категориальной модели мира. В разделе 5.4 предлагается новый метод подсчета внутреннего вознаграждения, основаннай на оценке среднего расстояния до памяти прецедентов.

2 Основные понятия и определения

2.1 Обучение с подкреплением

Обучение с подкреплением является одним из вариантов формализации задачи последовательного принятия решений. Агент последовательно взаимодействует со средой с помощью действий, получая в ответ наблюдения и вознаграждение. Функция вознаграждения задается таким образом, чтобы агент максимизирующий суммарное вознаграждения, решал поставленную экспериментатором задачу принятия решений. Например, в задаче навигации к некоторому участку среды, вознаграждение нулевое везде, кроме целевого участка, где оно равно единице.

Такая постановка может быть формализована с помощью частично-наблюдаемого Марковского процесса принятия решения (ЧНМППР): $(S, A, T, R, O, \Omega, \rho_0)$, где

- S - множество состояний,
- A - множество действий,
- $T(s, a, s') = Pr(s_{t+1} = s' | s_t = s, a_t = a)$ - функция перехода $(s, s' \in S, a \in A)$,
- $R(s, a) \to \mathbf{R}$ - функция вознаграждения,
- O - множество наблюдений,
- $\Omega(s, o) = Pr(o_t = o | s_t = s)$ - функция наблюдения,
- $\rho_0(s) = Pr(s_0 = s)$ - распределение начального состояния.

Стратегия, действующая в среде на основании наблюдений: $\pi(a|h_t) = Pr(a_t = a|h_t)$, где $h_t = f(h_{t-1}, o_t)$, f - некоторая функция, которая кодирует все предыдущие наблюдения. Задачей ОсП является обучение стратегии максимизирующей суммарное вознаграждение: $\pi^* = argmax_\pi \mathbf{E}_{\pi,T} \sum_{t=0\ldots\infty} \gamma^t R(s_t, a_t)$, где $\gamma \in [0, 1)$ - дисконтирующий множитель, а математическое ожидание берется по эпизодам следующего вида: $s_0 \sim \rho_0(s), a_t \sim \pi(a|h_t), s_{t+1} \sim T(s_t|a_t, s_{t+1}), o_t \sim \Omega(s_t, o_t)$.

Отметим, что ЧНМППР является обобщением Марковского процесса принятия решений (МППР), соответствующему частному случаю: $O = S$, $\Omega(s) = s$. В таком случае для стратегии достаточно текущего наблюдения для принятия решения, т.е. $h_t = f(h_{t-1}, o_t) = o_t = s_t$ и $\pi(a|h_t) = \pi(a|s_t)$. Далее для МППР будут приведены

определения и теоремы позволяющие обучать стратегии, максимизирующие вознаграждение.

Определение 2.1. *Функцией полезности состояния стратегии* π *называется значение* $V^\pi(s) = \mathbf{E}_{\pi,T}[\sum_{t=0\ldots\infty} \gamma^t R(s_t, a_t)|s_0 = s]$.

Определение 2.2. *Функцией полезности действия стратегии* π *называется значение* $Q^\pi(s, a) = \mathbf{E}_{\pi,T}[\sum_{t=0\ldots\infty} \gamma^t R(s_t, a_t)|s_0 = s, a_0 = a]$.

Теорема 2.1. *Равенства Беллмана:*

$$V^\pi(s) = \mathbf{E}_{a\sim\pi} Q^\pi(s, a)$$

$$Q^\pi(s, a) = R(s, a) + \gamma \mathbf{E}_{s'\sim T} V^\pi(s')$$

Теорема 2.2. *Для любого МППР верно:*

$$\nabla_\pi \mathbf{E}_{\pi,T} \sum_{t=0\ldots\infty} \gamma^t R(s_t, a_t) = \mathbf{E}_{\pi,T}(\sum_{t=0\ldots\infty} \gamma^t R(s_t, a_t)) \nabla_\pi \log \pi(a_t|s_t) \tag{2.1}$$

Данная теорема вдохновляет многие алгоритмы ОсП [5, 9].

2.2 Теория информации в обучении с подкреплением

Теория информации дает математическую базу для оценки количества информации в случайных величинах и зависимости нескольких случайных величин. В качестве оценки количества информации в одной случайной величине используется энтропия:

$$H(x) = \mathbf{E}_{x\sim p(x)} - \log p(x) = \int_x -p(x) \log p(x)$$

В случае дискретных случайных величин интеграл упрощается в сумму по принимаемым случайной величиной значениям.

В случае наличия двух зависимых случайных величин энтропия принимает следующий вид:

$$H(x) = \int_x - \int_y p(x|y)p(y) \log \int_y p(x|y)p(y)$$

Также вводится условная энтропия:

$$H(x|y) = \int_y \int_x -p(x|y)p(y) \log p(x|y)p(y)$$

Для оценки зависимости двух случайных величин используется взаимная информация:

$$I(x;y) = \mathbf{E}_{x\sim p(x), y\sim p(y)} \log \frac{p(x,y)}{p(x)p(y)} =$$

$$= H(x) - H(x|y) = H(y) - H(y|x)$$

Описанные выше понятия широко используется в машинном обучении и в частности в обучении с подкреплением. Например, максимизация энтропии стратегии приводит к улучшению результатов ОсП [14], максимизация взаимной информации между действиями и наблюдениями может использоваться для обучения навыков [1, 11], максимизация взаимной информации между параметрами модели мира и состояниями среды позволяет получить метод исследования среды [21].

Также определим меру схожести двух распределений - расстояние Кульбака-Лейблера:

$$KL(p(x)||q(x)) = \mathbf{E}_{x\sim p(x)} \log \frac{p(x)}{q(x)}$$

2.3 Обучение с подкреплением на основе модели мира

Методы обучения с подкреплением на основе модели мира (ОсПнОММ) помимо стратегии и функций полезности обучают модель мира для аппроксимации функции перехода T среды. Данная модель может использоваться методами планирования для решения задачи, либо она может использоваться, как среда, в которой обучаются агенты ОсП. Во втором случае есть несколько преимуществ перед обучением агентов в оригиальной среде: в качестве стартового состояния может быть выбрано любое состояние из уже посещенных, взаимодействие с моделью мира быстрее, чем с оригинальной средой, и для дифференцируемых моделей мира и непрерывных действий появляется возможность оптимизировать стратегии непосредственно с помощью градиентного спуска, что стабилизирует обучение.

Для МППР с низкоразмерными состояниями можно обучать функцию перехода $p(s_{t+1}|s_t, a_t)$, однако в случае высокоразмерных состояний (например картинки)

необходимо обучать модель в латентном пространстве $p(z_{t+1}|z_t, a_t)$, где z_t - некоторый вектор, поставленный в соответствие состоянию среды с помощью кодировщика $z_t = e(s_t)$. Для сред с частичной наблюдаемостью необходима рекуррентная модель, так как наблюдения не передают всю информацию о состоянии среды: $p(z_{t+1}|h_t, z_t, a_t), z_t = e(o_t), h_t = f(h_{t-1}, z_t)$, где f - некоторая функция.

Далее описывается алгоритм ОсПнОММ Dreamer [15, 20, 22], который реализует описанный выше функционал для частично-наблюдаемых сред с наблюдениями высокой размерности:

$$\begin{aligned} \text{Рекурентная модель:} \quad & h_t = f(h_{t-1}, z_{t-1}, a_{t-1}) \\ \text{Модель представления:} \quad & z_t \sim q_\varphi(z_t|h_t, o_t) \\ \text{Предсказатель функции перехода:} \quad & z_t \sim p_\varphi(z_t|h_t) \\ \text{Предсказатель наблюдения:} \quad & \hat{o_t} \sim p_\varphi(\hat{o_t}|h_t, z_t) \end{aligned}$$

h_t и z_t являются детерминированным и стохастичным внутренними состояниями соответственно. В изначальном алгоритме Dreamer [15] состояние z_t сэмплировалось из гауссовского распределения, параметризованного средним и дисперсией, которые предсказывает q_φ:

$$z_t \sim \mathcal{N}(\mu, diag(\sigma))$$

$$(\mu, \sigma) = q_\varphi(h_t, o_t)$$

, где $diag(\sigma)$ - матрица, у которой по диагонали значения вектора σ, а остальные нули.

В дальнейших версиях алгоритма [20] было предложено применять в качестве распределения вектор из категориальных распределений:

$$z_t \sim \begin{bmatrix} cat((p_{0j})_{j=0\ldots C}) \\ \vdots \\ cat((p_{ij})_{j=0\ldots C}) \\ \vdots \\ cat((p_{Sj})_{j=0\ldots C}) \end{bmatrix}$$

$$(p_{ij})_{i,j=1...C} = q_\varphi(h_t, o_t)$$

, где $cat(p_0, p_1, ..., p_n)$ - категориальное распределение с данными вероятностями, C - количество классов, S - количество распределений, т.е. после сэмплирования z_t имеет форму матрицы с S строками и C столбцами, и в каждой строке есть одна единица и все остальные значения нулевые. Переход к такому виду внутреннего состояния модели позволили улучшить результаты работы алгоритма [20]. Для упрощения в дальнейшем такое распределение будем обозначать следующим образом:

$$z_t \sim cat(p_{ij}), i = 0...S, j = 0...C$$

Также вводится модель предсказывающая вознаграждение среды, чтобы можно было обучать агента, максимизирующего это вознаграждение:

$$\hat{r}_t \sim p_\varphi(\hat{r}_t | h_t, z_t)$$

Все описанные выше модели обучаются предсказывать поведение среды. Обучение параметров происходит за счет минимизации следующей функции ошибки:

$$L(\varphi) = \mathbf{E} \sum_t - \ln p_\varphi(o_t|h_t, z_t) - \ln p_\varphi(r_t|h_t, z_t) + \beta KL[q_\varphi(z_t|h_t, o_t)||p_\varphi(z_t|h_t)]$$

, где математическое ожидание берется по стратегии и по динамике среда. Все модели параметризованы глубокими нейронными сетями, и их параметры оптимизируется с помощью стохастического градиентного спуска по случайным выборкам из памяти прецедентов для минимизации $L(\varphi)$.

Для обучения агента, максимизирующего вознаграждение, обучаются две модели: функция полезности $v_\zeta(z_t)$ и стратегия $\pi_\psi(a_t|z_t)$. Они обучаются на траекториях длины H, полученных с помощью модели мира: $z_t \sim p(z_t|h_{t-1}, z_{t-1}, a_{t-1}), a_t \sim \pi_\psi(a_t|z_t)$. Функция полезности обучается для минимизации временной ошибки (temporal difference) с λ-отдачей [3]:

$$L(\zeta) = \mathbf{E}(v_\zeta(z_t) - V_t)^2$$

$$V_t = \hat{r_t} + \gamma \begin{cases} (1-\lambda)v_\zeta(z_{t+1}) + \lambda V_{t+1} & \text{, if } t < H \\ v_\zeta(z_H) & \text{, if } t = H \end{cases}$$

, где $\lambda \in [0, 1]$ - гиперпараметр.

Для обучения параметров стратегии π минимизируется следующий лосс:

$$L(\psi) = \mathbf{E} \sum_{t=1...H} (-\rho \ln p_\psi(a_t|z_t) sg(V_t - v_\psi(z_t)) - (1-\rho)V_t) - \eta H[a_t|z_t]$$

, где sg останавливает градиент: $\nabla_\theta sg(f(\theta)) = 0$. Первое слагаемое лосса соответствует обучению по теореме о градиенте стратегии (2.1), второе слагаемое соответствует обучению дифференцированием функции полезности и третье слагаемое отвечает за энтропию стратегии, максимизация которой улучшает обучение [14]. Для сред с непрервным пространством действий $\rho = 0$, а для сред с дискретными действиями $\rho = 1$, так как для дискретных действий V_t недифференцируемо по параметрым стратегии.

2.4 Внутренняя мотивация

Внутренней мотивацией называют заложенное разработчиком алгоритма не зависящее от внешнего вознаграждения знание о предпочитаемом поведении агента. Внутренняя мотивация наблюдается у биологических агентов (например у людей), и её примером может служить любопытство, которое в рамках ОсП означало бы намерение посетить неисследованные состояния среды. Подобные аналогии вдохновили применение внутренней мотивации в ОсП, и в частности некоторые типы мотивации у животных вдохновили алгоритмы внутренней мотивации [1].

Методы внутренней мотивации в ОсП можно разделить на два типа: направленные на исследование среды и направленные на обучение навыков. Во втором случае стратегия может зависеть от навыка c: $\pi(a_t|s_t, c)$. Обычно методы для решенияи задачи обучения навыков, реализуют алгоритм для максимизации взаимной информации между множеством навыков C и множеством состояний среды S:

$$\pi(a_t|s_t, c) = argmax I(C; S)$$

В случае применения внутренней мотивации для задачи исследования среды вводится дополнительная компонента: метод подсчета внутреннего вознаграждения. Этот метод может быть непараметрическим, то есть не применять обучаемых параметров для подсчета вознаграждения. В случае, если метод параметрический, для обучения параметров функции внутреннего вознаграждения $r_\theta(s_t)$ минимизируется некоторый лосс:

$$\theta = argmin_\theta L_i(\theta)$$

Одной из задач разработчика параметризированного метода подсчета внутреннего вознаграждения является введения такой функции ошибки L_i, что максимизация этого вознаграждения приведет к исследованию среды.

3 Обзор литературы

В данном разделе рассматриваются предложенные в литературе методы подсчета внутреннего вознаграждения для исследования среды. Такие методы можно разделить на несколько типов: основанные на ошибке предсказания, основанные на новизне состояний и основанные на приращении информации о среде. Далее рассмотрим все типы в отдельности.

3.1 Ошибка предсказания

В таких методах идея состоит в том, чтобы направлять агента к областям среды, где предсказание следующего состояния с использованием информации только о ранее посещенных состояниях затруднено. Такое внутреннее вознаграждение можно формализовать, как ошибку предсказания:

$$r_i(s_t, a_t, s_{t+1}) = ||g(s_t) - F(s_{t+1}|s_t, a_t)||$$

, где $g(s_t)$ - некоторое представление состояния, F - обучаемая функция переодов.

В работе [13] показано, что применение $g(s_t) = s_t$ не работает для высокоразмерных состояний, однако для конкурентных результатов с другими методами достаточно использовать в качестве функции g случайную функцию, формирующую случайные низкоразмерные свойства состояния s_t. Однако такой подход не может справиться с, так называемой, проблемой шумного телевизора: если среда сильно стохастична, то области стохастичности привлекают агента больше неизученных областей, что делает исследование среды затруднительным. Для решения этой проблемы в работе ICM [8] предлагается метод, который обучает функцию g с помощью предсказывания выбранного действия по смежным состояниям:

$$g(s_t) = argmin||a_t - f(s_t, s_{t+1})||$$

Такой подход позволяет не учитывать при исследовании те переходы, которые не зависят от действий агента, что позволяет решить проблему шумного телевизора.

3.2 Новизна состояний

В подходах данного типа в качестве внутреннего вознаграждения агенту подается оценка того, насколько это состояние ново для агента. В средах в дисретными состояниями это значение можно посчитать явно следующим образом:

$$r_i(s_t) = \frac{1}{N(s_t)}$$

, где $N(s_t)$ - количество раз, которое состояние s_t было посещено ранее. Очевидно, что эта формула не может быть применена для задач, в которых пространство состояний непрерывно. Для сред с непрерывными состояниями в работе [7] предлагается использовать плотность посещенных состояний для оценки новизны:

$$r_i(s_t) = \frac{1}{\hat{N}({}_t s)}$$

$$\hat{N}(s_t) = \frac{\rho(s_t)(1 - \rho'(s_t))}{\rho'(s_t) - \rho(s)}$$

, где ρ - оценка плотности посещенных состояний, а ρ' - оценка вероятности посетить s_t сразу после посещения s_t.

Так для многих сред явный подсчет плотности состояний затруднителен, некоторые методы стремятся достичь оценки новизны без оценки плотности. Например, в работе [19] показывается, что норма свойства последователей (successor features) [10] может использоваться в качестве внутреннего вознаграждения, и это вознаграждение ведет себя схожим образом с новизной состояний.

В работе [12] предложен метод RND для оценки новизны состояния предлагается использовать ошибку предсказания модели. Метод использует две нейросети: целевую $T(s_t)$ и обучаемую $N_(\theta)(s_t)$. Первая нейросеть инициализируется случайными параметрами и не меняется по ходу исследования среды, а вторая нейросеть обучается предсказвыать выход первой:

$$L(\theta) = \mathbf{E}_{s_t \sim B} ||T(s_t) - N_\theta(s_t)||$$

И в качестве внутреннего вознаграждения используется следующее значение:

$$r_i(s_t) = ||T(s_t) - N_\theta(s_t)||$$

3.3 Приращение информации

Использование приращения информации о среде в качестве внутреннего вознаграждения подразумевает, что агент стремить уменьшить неопределнность свих знаний о среде. Это направляет агента в неисследованные области, так как там можно уменьшить неопределенность сильнее, чем в уже исследованных. Также такой подход способен решать проблему шумного телевизора, так как в стохастичных областях среды невозможно уменьшить неопределнность, она там всегда будет высокой.

Несколько методов стремятся реализовать такой подход. Например, в работе предлагается использовать Байесовскую нейронную сеть [2] для аппроксимации динамики среды. Тогда уменьшение неопределенности параметров этой сети используется в качестве внутреннего вознаграждения. Однако такой подход работает только на простых средах.

Некоторые работы [18, 17] рассматривают задачу подсчета неопределнности с помощью ансамбля моделей мира. Они параллельно обучют несколько моделей мира и используют оценку разногласия предсказания моделей в качестве внутреннего вознаграждения. Эти работы отличаются лишь в способе подсчета разногласия: [17] использует дисперсию предсказаний, а [18] использует дивергенцию Йенсена-Шеннона. Такой подход позволяет исследовать среду, так как в уже неоднократно посещенных состояниях модели обучатся предсказывать похожие значения, и поэтому вознаграждение будет маленьким, а в неисследованных состояниях предсказания моделей будут сильно отличаться, а следовательно вознаграждение будет большим.

4 Формальная постановка задачи

В данной работе рассматриваются методы исследования среды, а в частности методы подсчета внутреннего вознаграждения. Для уменьшения влияния других компонент агента на метод исследования не рассматривается постановка ОсП, так как в таком случае на выбор действия агентом влияет как внутренне, так и внешнее вознаграждение. Вместо этого будет рассмотрена постановка обучения с подкреплением без учителя. В такой постановке обучение делится на два этапа: предобучение и дообучение. На этапе предобучения агент не имеет доступа к внешенму вознаграждению, его целью является как можно больше исследовать среду и обучить компоненты, которые могут быть полезны на этапе дообучения. На этапе дообучения агент получает доступ к внешенму вознаграждению, и от него требуется за минимальное количество шагов в среде обучить стратегию, максимизирующую это возграждение.

В качестве метрики для сравнения эффективности методов исследования используется доля посещенных состояний среды. Однако не во всех средах ее можно посчитать, поэтому в некоторых средах в качестве метрики используется эффективности обученной на этапе дообучения стратегии. В такой постановке можно оценить только финальный результат исследования, но было бы полезно иметь значения метрики на промежуточных шагах предобучения, как для метрики доли посещенных состояний. Чтобы этого добиться классическая постановка задчи ОсПбУ преобразуется: исследование среды и обучение стратегии, максимизирующей внешнее вознаграждение, происходят параллельно.

Среда формализуется с помощью ЧНМППР. Агент состоит из следующих компонент:

- модель мира Dreamer,
- внутреннее вознаграждение r_i,
- стратегия π_i, максимизирующая внутреннее вознаграждение (исследовательская стратегия),
- стратегия π_e, максимизирующая внешнее вознаграждение.

Со средой взаимодействует только исследовательская стратегия:

$$\tau = (o_0, a_0, o_1, a_1, ..., o_i, a_i, ...) \sim T, \pi_i$$

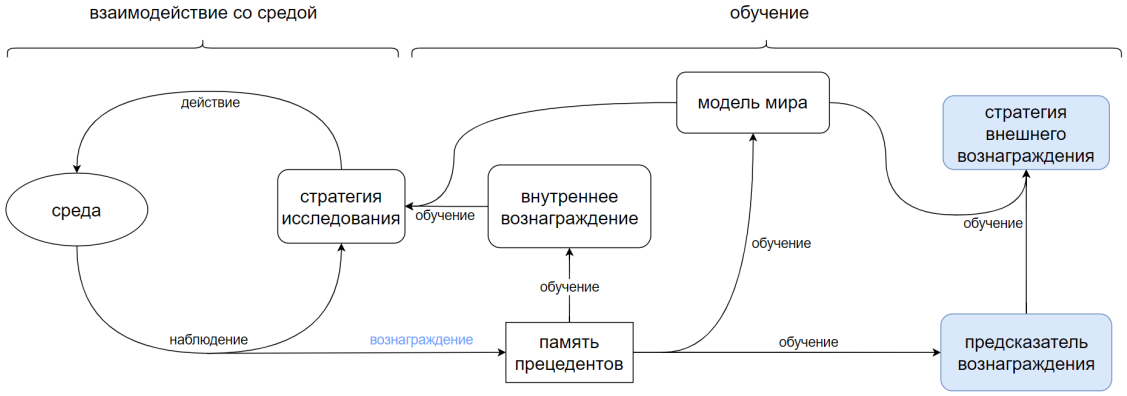

Рис. 1: Схема работы алгоритма. Голубым указаны компоненты, которые необязательны для оценки результатов исследования в средах, в которых можно посчитать долю посещенных состояний.

.

Полученные траектории сохраняются в память прецедентов B, по сэмплам из которого обучаеся модель мира. Обе стратегии обучаются максимизировать соответствующее вознаграждение в воображении, как описано в алгоритме Dreamer (раздел 2.2). Схема работы алгоритма представлена на рисунке 1.

5 Предложенные методы и вычислительные эксперименты

5.1 Среды

В данном разделе описываются среды, на которых проводились эксперименты. На рисунке 2 приведены картинки, описывающие эти среды.

5.1.1 Pointmaze

Среда представляет из себя ограниченное двумерное пространство со стенами. Агент - это точка в этом пространстве, и действиями являются направление движения агента. В случае, если выполнению действия не мешают стены, к позиции агента прибавляется вектор направления. Если на пути оказались стены, то позиция агента обрезается до ближайшей стены. Для усложнения среды, в отличие от предыдущих работ [24], мы рассматриваем среду не в виде лабиринта, а в виде набора комнат, соединенных между собой дверьми (отсутствием стены). При этом размеры дверей меньше максимального размера действия и двери расположены в случайных местах между комнатами. На рисунке 2 слева показана используемася среда: черным нарисованы стены, а голубым - двери. Агент всегда начинает эпизод в левом нижнем углу. Сложностью для исследования является невозможность обобщать расположение дверей и, следовательно, трудность обнаружения двери в новой комнате.

Помимо основной среды, которая является детерминированной, непрерывной и полностью наблюдаемой, вводятся варианты с противоположными характеристиками. Чтобы сделать среду стохастичной, к действию прибавляется случайный шум. Для получения дискретной среды действия дисретизируются по 32 равномерно разбитым направлениям движения. Для получения частично-наблюдаемой среды к наблюдаемому состоянию прибавляется случайный шум, и также в качестве наблюдения дается предыдущий размер шага, выполненного в среде. Это приводит к тому, что, чтобы знать текущую позицию агента, требуется учитывать все предыдущие наблюдения.

5.1.2 Antmaze

Среда представляет из себя небольшой лабиринт. Агент состоит из 4 ног с двумя суставами каждая - по одному у точки соединения ног и по одному в середине ног.

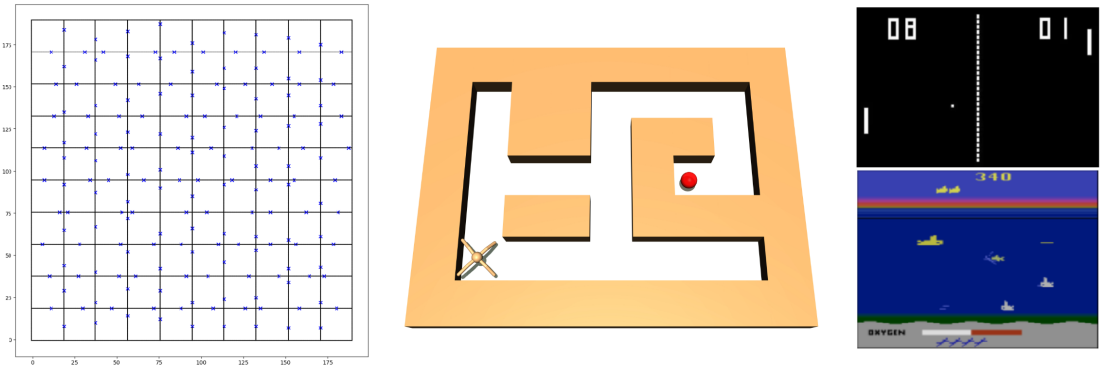

Рис. 2: Среды

Действиями являются векторы ускорения движения каждого сустава. В качестве наблюдения агент получает позицию в среде и углы поворота ног. Среда и агент представлены на рисунке 2. Агент всегда начинает эпизод в левом нижнем углу. Данная среда является детерминированной, и при рассмотрении стохастичной среды к действиям прибавляется случайный шум.

5.1.3 Atari

Описанные выше среды имеют низкорзмерные наблюдения. В качестве среды с высокоразмерными наблюдениями были выбраны среды Pong и Seaquest из набора сред Atari. В литературе для усложнения задачи рассматриваются преобразованные среды с стохастичностью: при выполнении действия с некоторой вероятностью выбирается предыдущее действие, вне зависимости от выбранного. В экспериментах для рассмотрении детерминированной среды это преобразование убирается.

5.1.4 Метрики

Для оценки качества исследовательской стратегии используются следующие метрики: для pointmaze - доля посещенных состояний среды, дискретизированных с шагом 5; для antmaze - доля посещенных состояний среды, дискретизированных с шагом 2 (без учета положения ног агента); для atari - суммарное за эпизод вознаграждение стратегии, максимизирующей внешнее вознаграждение.

5.2 Поведение разногласия ансамбля

В данном разделе будут рассмотрены результаты работы алгоритма исследования, использующего в качестве внутреннего вознаграждения ошибку ансамбля моделей мира, на средах, описанных в предыдущем разделе. Рассматриваются результаты на моделях мира с различным распределением латентного состояния: гауссовским и категориальным; и с различным целевым вектором предсказания моделей ансамбля: стохастичное состояние модели мира и детерминированное.

5.2.1 Разногласие ансамбля

Использование дисперсии ансамбля моделей мира в качестве внутреннего вознаграждения было предложено в работе [17], и метод был назван disagreement. В алгоритме plan2explore [21] этот метод был адаптирован для ОсПбУ и для обучения на основе модели мира Dreamer.

Имеются K одношаговых моделей мира $q_i, i = 0...K$ с параметрами $w_i, i = 0...K$. Каждая из моделей получает на вход состояния модели мира Dreamer h_t, z_t и действие a_t. По этому входу модели предсказывают вектор x_t, который может быть детерминированным состоянием h_t или стохастичным состоянием z_t:

$$q_i(x_{t+1}|h_t, z_t, a_t) = \mathcal{N}(\mu_i(h_t, z_t, a_t), 1)$$

, т.е. выход моделей параметризуется нормальным распределением с единичной матрицей ковариаций.

Модели ансамбля обучаются предсказывать поведение модели мира Dreamer. В случае, если $x_t = h_t$ минимизируется следующий лосс:

$$L(w_i) = \mathbf{E}(\mu_i(h_t, z_t, a_t) - f_\varphi(h_t, z_t, a_t))^2$$

Если берется $x_t = z_t$, то лосс принимает следующий вид:

$$L(w_i) = \mathbf{E}(\mu_i(h_t, z_t, a_t) - \hat{z}_{t+1})^2$$

, где $\hat{z}_{t+1} \sim p_\varphi(z_{t+1}|f(h_t, z_t, a_t))$. В обоих случая ожидание берется по распределению состояний при взаимодействии исследовательской стратегии со средой, и оно оценивается с помощью сэмплрования из памяти прецедентов.

Далее в качестве внутреннего вознаграждения для некоторого состояния берется дисперсия предсказания моделей ансамбля:

$$r_i(h_t, z_t, a_t) = Var(\{\mu_i(h_t, z_t, a_t), i = 0...K\}) =$$

$$= \frac{1}{K-1} \sum_i (\mu_i(h_t, z_t, a_t) - \mu')^2$$

$$\mu' = \frac{1}{K} \sum_i \mu_i(h_t, z_t, a_t)$$

Данный алгоритм имеет два архитектурных решения, которые требуется сделать при применении: 1) обучать на основе модели мира с гаусовским или категориальным распределением состояния z_t, 2) использовать в качестве цели предсказания ансамблей x_t детерминированное h_t или стохастичное z_t состояние модели мира. Эти два выбора могут влиять на результаты друг друга, например при выборе $x_t = z_t$ для категориальной и гауссовской модели мира модели будут предсказывать векторы с разными качествами: вектор только со значениями 0 и 1, либо вектор с непрерывными значениями. Поэтому в экспериментах рассматриваются все 4 комбинации:

- gaus stoch: $x_t = z_t$, гауссовская модель
- cat stoch: $x_t = z_t$, категориальная модель
- gaus deter: $x_t = h_t$, гауссовская модель
- cat deter: $x_t = h_t$, категориальная модель

5.2.2 Вычислительные эксперименты

Эксперименты проводятся в соответствии постановке, представленной в разделе 4, на средах, описанных в разделе 5.1. В таблице 1 представлены результаты экспериментов.

По результатам, представленным в таблице, можно сделать вывод, что использование гауссовской модели мира приводит к лучшим результатам. Также интересно, что в отличие от оригинального метода, в котором предполагается, что модели ансамбля предсказывают стохастичную часть z_t модели мира, эксперименты показали, что для гауссовской модели ошибка предсказания детерминированного состояния h_t

Таблица 1: Результаты экспериментов с разногласием ансамбля в качестве внутреннего вознаграждения. В первом столбце представлены методы: gaus/cat - гауссовская и категориальная модель мира соответственно, deter/stoch - детерминированное и стохастичное состояние в качестве цели предсказания моделей соответственно. Подписи под средами означают описанный в разделе 5.1 тип среды: classic - классическая среда, stoch - стохастичная среда, discrete - среда с дискретными действиями, pomdp - частично-наблюдаемая среда. Для сред pointmaze и antmaze в качестве результатов представлены доли посещенных состояний среды в процентах. Для сред pong и seaquest в качестве результатов представлено суммарное за эпизод вознаграждение, полученное максимизирующей внешнее вознаграждение стратегией.

-	pointmaze				antmaze		pong		seaquest	
-	classic	stoch	discrete	pomdp	classic	stoch	classic	deter	classic	deter
gaus deter	**39**	**24**	71	27	**96**	**94**	4.7	**21**	**3500**	**4000**
gaus stoch	31	19	**78**	23	91	88	6.5	11	2100	1500
cat deter	25	12	20	**31**	91	88	**14**	19	1700	1200
cat stoch	17	**25**	54	23	77	76	8.7	19	2500	2400

показывает лучшие результаты. Метод gaus deter показывает лучший результат на большинстве сред: 7 из 10; поэтому для задачи исследования в схожих с рассмотренными средах можно его рекомендовать в качестве базового метода. Также заметим, что в средах, где gaus deter уступает другим методам, разница является небольшой, за исключением классического pong.

Для категориальной модели мира нельзя выделить, какой целевой вектор для моделей ансамбля является предпочтительным, так на разных средах тот или иной вариант показывает значительно худший результат по сравнению с другим. Также отметим, что для категориальной модели на некоторых средах оба варианта - детерминированное или стохастичное целевое состояние - показывают значительно худшие результаты по сравнению с гауссовской моделью. Это происходит несмотря на то, что некоторые алгоритмы [23] применяют plan2explore с категориальной моделью мира, не сравниваясь с гауссовской моделью мира. Учитывая это, а также то, что 1) категориальная модель мира Dreamer показала значительно лучшие результаты для задачи ОсП на многих средах [20, 22], 2) изначально метод разногласия ансамбля был предложен для гауссовской модели мира, можно сделать вывод, что для улучшения исследования среды и вообще работы агента в задаче ОсПбУ необходимо рассмотреть модификации метода подсчета разногласия ансамбля для категоримальной модели

мира. Такие методы будут предложены в следующей секции.

5.2.3 Ошибка модели мира

Для потверждения полученных результатов и сделанных выводов требуется подтвердить, что эти результаты зависят от метода исследования, а не от модели мира, применяемой для обучения стратегий. Могла возникнуть ситуация, что плохие результаты исследования для категориальной модели мира связаны с тем, что для рассмотренных сред, категориальная модель хуже предсказывает динамику среды по сравнению с гауссовской моделью. На рисунке 3 представлен график изменения функции ошибки во время обучения. Можно увидеть, что значение ошибки на протяжении почти всего обучения выше для гауссовской модели мира. Это опровергает гипотезу о том, что категориальная модель справляется с предсказанием динамики среды хуже, чем гауссовская. Все значения близки, но, предположительно, функция ошибки для гауссовской модели несколько выше, так как по причине лучшего исследования в памяти прецедентов находится больше разнообразных состояний.

5.2.4 Размер категориального состояния

Рассмотрим также влияние на исследование размера категориального состояния модели: количества классов категориальных распределений и количество S этих распределений:

$$z_t \sim cat(p_{ij}), i = 0...S, j = 0...C$$

Для этого рассмотрим результаты работы исследовательского агента на среде pointmaze для различных значений:

- k = n = 8
- k = n = 12
- k = n = 16

На рисунке 4 показана доля посещенных состояний во время обучения для метода с целевым вектором предсказания моделей ансамбля - стохастичным состоянием z_t модели мира. На рисунке видно, что при увеличении размера категориального

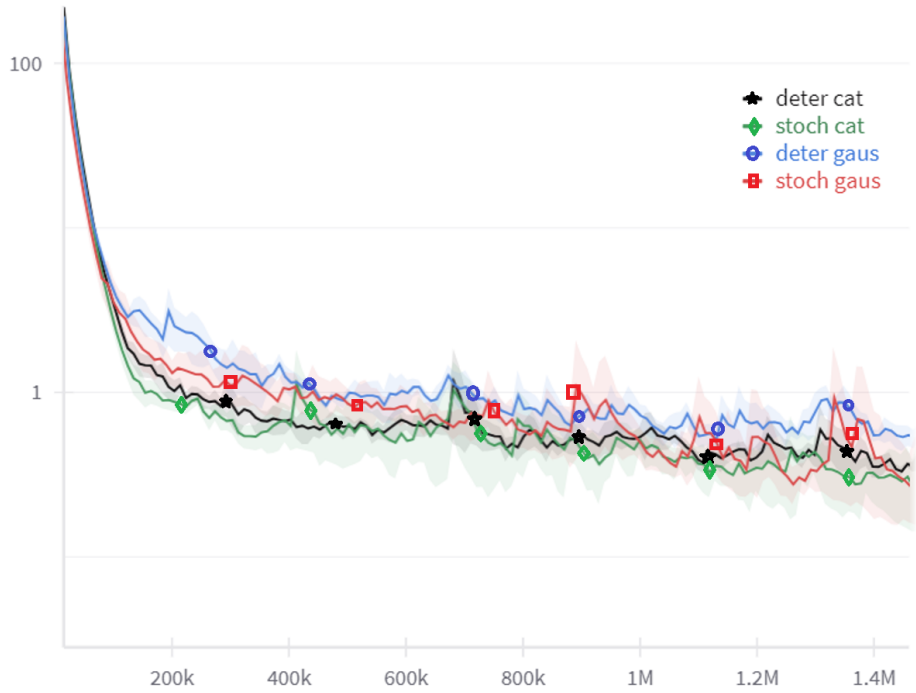

Рис. 3: Значения функции ошибки восстановления наблюдения после одношагового предсказания модели мира на среде pointmaze. Обозначения в легенде: gaus/cat - гауссовская и категориальная модель мира соответственно, deter/stoch - детерминированное и стохастичное состояние в качестве цели предсказания моделей соответственно.. По горизонтальной оси количество шагов в среде (k = 1000, M = 1000000). По вертикальной оси значения ошибки на логарифмической шкале.

состояния улучшается и качество исследования, однако для наибольшего рассматриваемого размера - 16x16 - качество исследования не превосходит метода с гауссовской моделью мира. Отметим также, что на рисунке представлены только методы, которые используют в качестве целевого значения моделей ансамбля стохастичное состояние z_t. Гауссовская модель с целевым детерминированным состоянием всё ещё значительно превосходит метод с наибольшим размером категориального состояния. При этом для размера 16x16 размерность предсказываемого вектора уже в 2.5 раза больше, чем соответствующая размерность для гауссовской модели.

Также рассмотрим влияние размера категориального состояния на исследование в случае применения детерминированного состояния h_t в качестве целевого. На рисунке 5 видно, что увелиичение размера категориального состояния почти не влияет на результаты исследования.

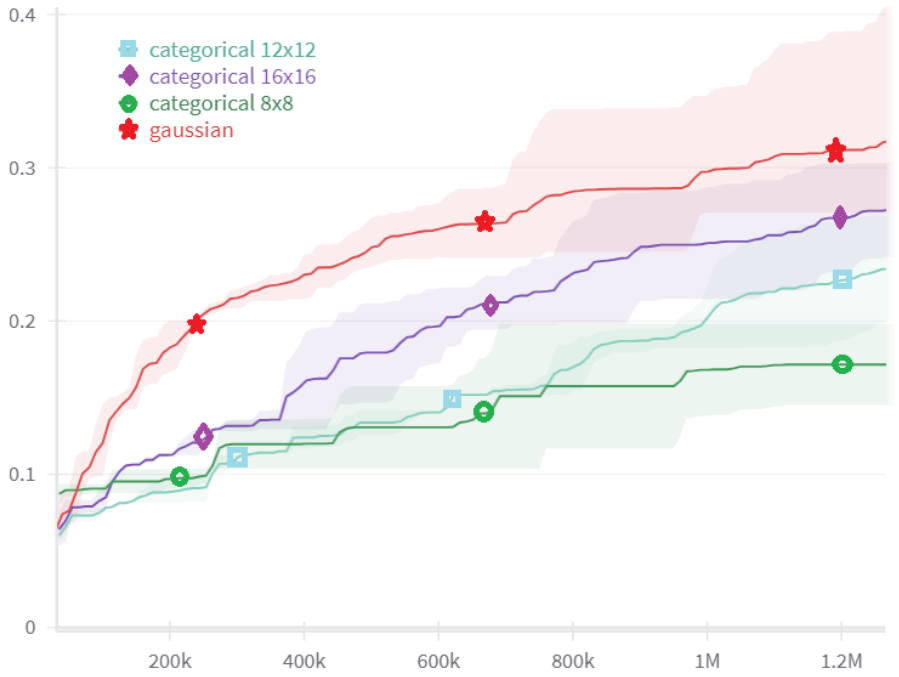

Рис. 4: Доля посещенных состояний в среде pointmaze для различных размеров категориального состояния модели мира. Целевой вектор предсказания моделей ансамбля - стохастичное состояние z_t. По горизонтальной оси количество шагов в среде (k = 1000, M = 1000000). По вертикальной оси доля посещенных состояний.

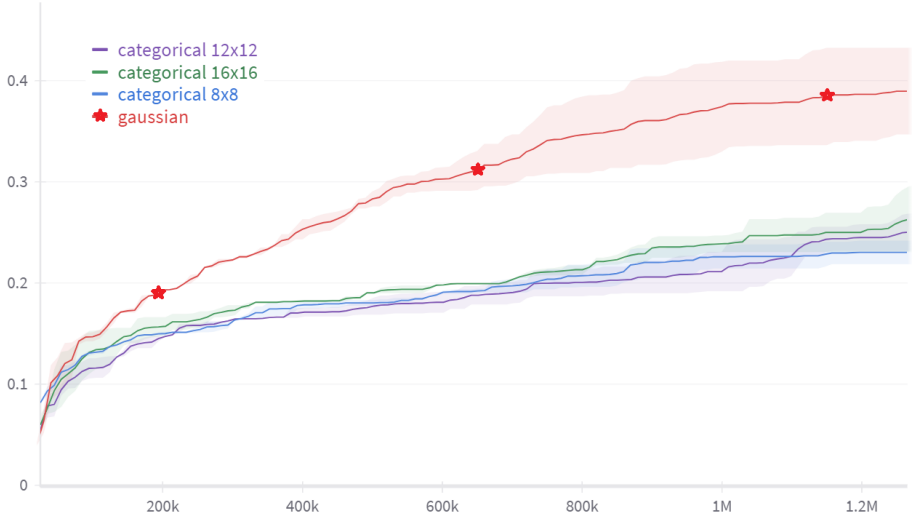

Рис. 5: Доля посещенных состояний в среде pointmaze для различных размеров категориального состояния модели мира. Целевой вектор предсказания моделей ансамбля - детериминированное состояние h_t. По горизонтальной оси количество шагов в среде (k = 1000, M = 1000000). По вертикальной оси доля посещенных состояний.

5.3 Разногласие ансамбля для категориальной модели мира

Учитывая результаты из прошлого раздела, которые показывают, что исследование среды с помощью категориальной модели мира уступает гауссовской, в дан-

ном разделе будут рассмотрены варианты преобразования подсчета разногласия ансамбля моделей, предсказывающих категориальное состояние z_t. Изначально метод подсчета разногласия был получен из задачи максимизации взаимной информации между параметрами модели мира и следующим состоянием [21]. Далее описывается этот вывод.

Пусть параметры w модели мира принадлежат некоторому распределению $p(w)$. Это распределение представляется в виде смеси точечных параметров моделей ансамбля w_i:

$$p(w) \triangleq \frac{1}{K} \sum_i \delta(w - w_i)$$

, где δ - дельта-функция Дирака.

Задача максимизации взаимной информации между параметрами модели и следующим состоянием может быть записана следующим обазом:

$$a_t^* = argmax_{a_t} I(z_{t+1}; w | z_t, a_t)$$

$$I(z_{t+1}; w | z_t, a_t) = H(z_{t+1} | z_t, a_t) - H(z_{t+1} | w, z_t, a_t) \tag{5.1}$$

, где H - энтропия.

Далее авторы оригинальной работы упрощают задачу, ограничивая распределение выхода модели до гауссовского распределения:

$$p(z_{t+1} | w_i, z_t, a_t) \triangleq \mathcal{N}(\mu(w_i, z_t, a_t), \Sigma)$$

, где Σ - единичная матрица ковариаций.

С таким упрощенем условная энтропия в равенстве 5.1 становится константной:

$$H(z_{t+1} | w, z_t, a_t) = \sum_i p(w_i) H(z_{t+1} | w_i) =$$

$$= \sum_i \frac{1}{k} H(\mathcal{N}(\mu(w_i, z_t, a_t), \Sigma)) =$$

$$= \frac{1}{k} \sum_i \ln \sigma_i + const = const$$

При этом можно заметить, что энтропия $H(z_{t+1}|z_t, a_t)$, хоть и не может быть посчитана в явном виде, может быть максимизирована, если максимизировать расстояние между средними предсказаний моделей ансамбля $\mu(w_i, z_t, a_t)$. И одним из вариантов подсчета расстояния между средними является подсчет дисперсии. Так и был выведен подсчет разногласия ансамбля в виде дисперсии.

Однако для категориальной модели мира:

$$p(z_{t+1}|w_i, z_t, a_t) \triangleq \begin{bmatrix} cat((p_{0j})_{j=0...C}) \\ \vdots \\ cat((p_{ij})_{j=0...C}) \\ \vdots \\ cat((p_{Sj})_{j=0...C}) \end{bmatrix}$$

такое упрощение взаимной информации не может быть проведено.

Далее будут рассмотрены варианты подсчета взаимной информации для категориальной модели мира.

5.3.1 Допущение независимости распределений

Рассмотрим сначала одно категориальное распределение $z^k \sim cat((p_i^k), i = 0...C, k = 0...K)$, где K - количество моделей в ансамбле. Для такого распределения можно в явном виде посчитать взаимную информацию (для упрощения выкладок зависимость от предыдущих состояния и действия, а также времени, опускается):

$$H(z|w) = \sum_k \frac{1}{K} H(z|w_k) =$$

$$= \sum_k \sum_j -\frac{1}{K} p(z_j^k|w_k) \ln p(z_j^k|w_k) = \sum_k \sum_j -\frac{1}{K} p_j^k \ln p_j^k \tag{5.2}$$

, где $\sum_j$ - сумма по всем возможным состояниям, в данном случае по C векторам.

Другую энтропию можно посчитать следующим образом:

$$H(z) = \sum_j \sum_k (-\frac{1}{K} p(z_j^k|w_k)) \ln \sum_k \frac{1}{K} (p(z_j^k|w_k)) = \sum_j \sum_k (-\frac{1}{K} p_j^k \ln \sum_k \frac{1}{K} p_j^k) \tag{5.3}$$

Вернемся к начальной задаче: z сэмплируется из вектора нескольких категориальных распределений, обозначим p_{ij}^k вероятность для j-го класса i-го распределения

k-ой модели ансамбля. Введем следующее упрощение: будем считать, что различные распределения внутри одного вектора независимы. Тогда взаимную информацию можно посчитать отдельно для каждого распределения и потом просуммировать:

$$I(z;w) = H(z) - H(z|w) =$$

$$= \sum_k (H(z_i) - H(z_i|w))$$

Значения энтропий могут быть посчитаны с помощью формул 5.2 и 5.3.

Далее представлен код, реализующий подсчет разногласия ансамбля в соответствии с выдкладками (размер тензора P - (S, C, K))::

```
H_zw = (-P * log(P)).mean(-1).sum(-1)
mean = P.mean(-1)
H_z = (-mean * log(mean)).sum(-1)
d = H_z.sum(-1) - H_zw.sum(-1)
```

5.3.2 Упрощение с помощью неравенства Йенсена

Если считать все распределения зависимыми, тогда подсчет $H(z|w)$ примет следующий вид:

$$H(z|w) = \sum_k H(z|w_k) =$$

$$= \sum_k \sum_z -\frac{1}{K} p(z|w_k) \ln p(z|w_k) = \sum_k \sum_{j_1,\dots,j_C} -\frac{1}{K} \prod_i p^k_{ij_i} \ln \prod_i p^k_{ij_i} \tag{5.4}$$

, где $\sum_z$ - суммирование по всем возможным состояниям, которых C^S. Число состояний являтеся слишком большим, чтобы явно по ним итерировать, однако используя следующие равенста, формулу для подсчета энтропии можно упростить:

$$\ln \prod_i p^k_{ij_i} = \sum_i \ln p^k_{ij_i} \tag{5.5}$$

$$\sum_{j_1,\dots,j_C} -\frac{1}{K} \prod_i p^k_{ij_i} \ln p^k_{ij_i} = p^k_{ij_i} \ln p^k_{ij_i} \tag{5.6}$$

Тогда получаем:

$$H(z|w) = \sum_k \sum_i \sum_j -\frac{1}{d} p_{ij}^k \ln p_{ij}^k$$

Заметим, что полученное значение равно значению, полученному при допущении независимости распределений между собой.

Далее рассмотрим энтропию $H(z)$:

$$H(z) = \sum_z - \sum_k \frac{1}{K} p(z|w_i) \ln \sum_k \frac{1}{K} p(z|w_k)$$

Данную энтропию невозможно разложить, как $H(z|w)$, так как в данной формуле под логарифмом находится сумма величин. Чтобы преодолеть эту помеху можно воспользоваться неравенством Йенсена:

$$\ln \sum_k \frac{1}{K} x_k \geqslant \sum_k \frac{1}{K} \ln x_k$$

Получаем:

$$H(z) \leqslant \sum_z - \sum_k \frac{1}{K} p(z|w_k) \sum_k \frac{1}{K} \ln p(z|w_k)$$

Далее, применяя свойсива 5.5 и 5.6 также как и для $H(z|w)$, получаем:

$$H(z) = \sum_j \sum_i (-\frac{1}{d} p_{ij}^k) \ln \sum_i \frac{1}{K} (p_{ij}^k)$$

Далее представлен код, реализующий подсчет разногласия ансамбля в соответствии с выдкладками (размер тензора P - (S, C, K)):

```
H_zw = (-P * log(P)).mean(-1).sum(-1)
mean = P.mean(-1)
H_z = (-mean * log(P).mean(-1)).sum(-1)
d = H_z.sum(-1) - H_zw.sum(-1)
```

5.3.3 Вычислительные эксперименты

Рассмотрим как влияет на результаты исследования замена оригинального метода подсчета разногласия ансамбля на методы, предложенные в прошлом разделе. Все алгоритмы испльзуют категориальную модель мира, используют в качестве целевого вектора предсказания ансамбля стохастичное состояние z_t.

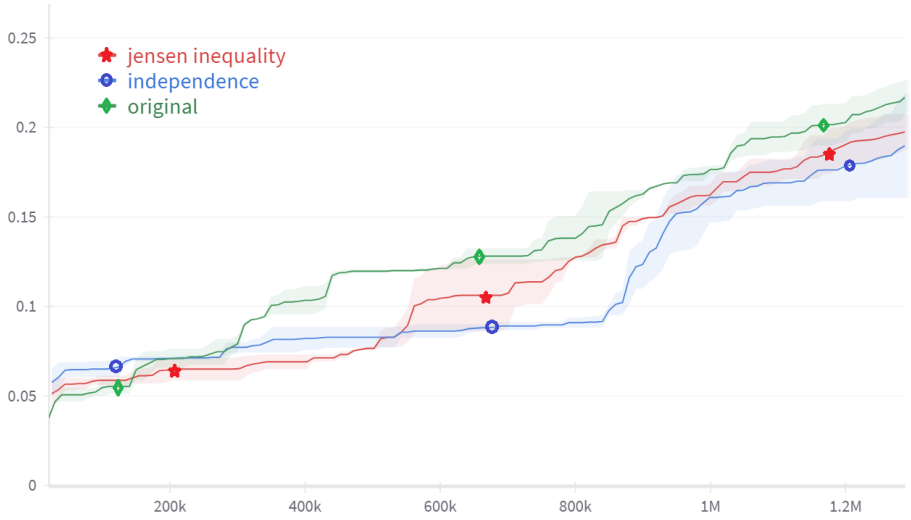

Рис. 6: Доля посещенных состояний различными вариантами подсчета разногласия модели на среде стохастичный pointmaze. original - дисперсия предсказаний моделей ансамбля, independence - предложенный в разделе 5.3.1 метод, jensen inequality - предложенный в разделе 5.3.2 метод. По горизонтальной оси количесвто шагов в среде (k = 1000, M = 1000000). По вертикальной оси доля посещенных состояний.

На рисунках 7 и 6 изображены результаты обучения на средах pointmaze и стохастичном pointmaze соответственно. Можно увидеть, что на детерминированной среде, предлженные методы показывают значительно худшие результаты по сравнению с оригинальным методом, что также значительно хуже, чем другие конфигурации подсчета разногласия ансамбля. На стохастичной среде отставание не так явно, но предложенные методы всё еще уступют в финальном значении доли покрытых состояний и в скорости исследования: на некоторых этапах обучения отставание значительно. В обоих средах метод, выведенный с помощью неравенства Йенсена, лучше метода, допускающего независимость распределений.

Из экспериментов можно сделать вывод, что предложенные методы сохраняют способность максимизировать взаимную информацию из изначальной задачи, что позволяет исследовать среду. Однако предложенные методы уступают способу подсчета разногласия в виде дисперсии.

5.4 Метод BatchDist

В данном разделе описывается предложенный метод подсчета внутреннего вознаграждения, основанный на аппроксимации среднего расстояния до состояний в памяти прецедентов. Так или иначе все методы исследования направлены на то,

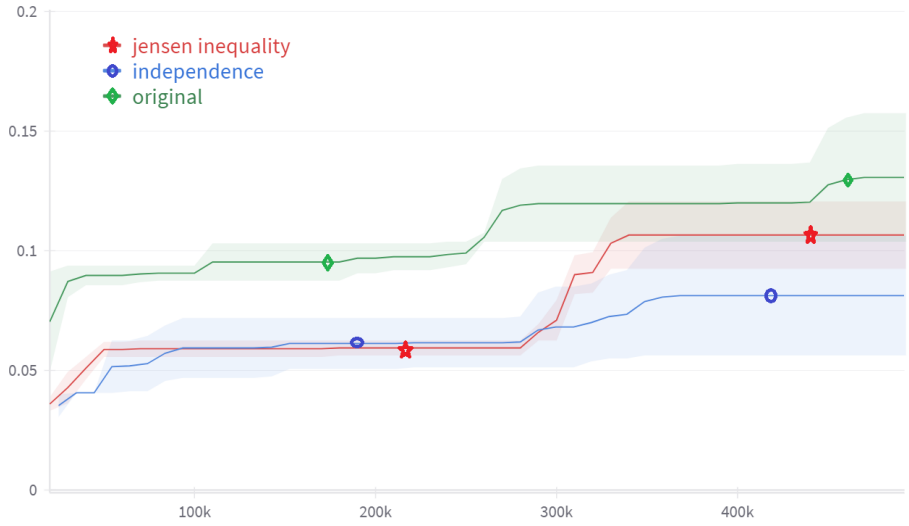

Рис. 7: Доля посещенных состояний различными вариантами подсчета разногласия модели на среде pointmaze. original - дисперсия предсказаний моделей ансамбля, independence - предложенный в разделе 5.3.1 метод, jensen inequality - предложенный в разделе 5.3.2 метод. По горизонтальной оси количесвто шагов в среде (k = 1000, M = 1000000). По вертикальной оси доля посещенных состояний.

чтобы посещать новые состояния. При этом часто посещенные состояния должны иметь низкое вознаграждение, так как, если агент их посетит, то ничего нового о среде он не узнает. На рисунке 8 показано в какие состояния среды стремится попасть агент исследования в внутренним вознаграждением disagreement. На левом лабиринте показаны уже посещенные агентом состояния, а на правом - области в которые агент стремится. Чтобы получить эти области, на текущем этапе обучения запускался агент из случайных состояний среды, и рассматривались конечные состояния траекторий. Можно увидеть, что чаще всего агент стремися в состояния на границе исследованного: состояния уже посещенные, но не успевшие оказаться в памяти прецедентов слишком много раз.

Вдохновляясь этим наблюдением, предлагается ввести метод подсчета внутреннего вознаграждения оценивающий отдаленность текущего состояний от состояний в памяти прецедентов. Для этого рассмотрим в качестве целевого значения среднее расстояние от текущего состояния до состояний из памяти прецедентов:

$$d(s) = \mathbf{E}_{s' \sim B} dist(s, s')$$

, где B - память прецедентов, а $dist$ - некоторая функция расстояния. Важными архитектурными решениями является выбор пространства, описывающего состоя-

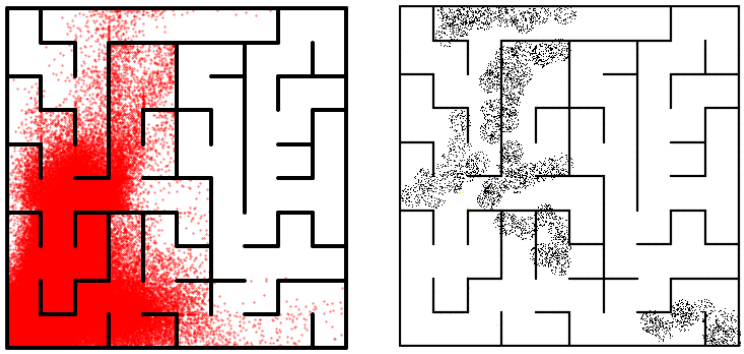

Рис. 8: Области среды, в которые стремится disagreement на определенном этапе обучения. Слева: красные точки - все ранее посещенные состояния. Справа: черные области - области в которых находятся финальные состояния траекторий стратегии исследования при сэмплировании начального состояния в случайных координатах среды.

ния, и выбор функции расстояния. В данной работе мы рассматриваем в качестве представления состояний детерминированное состояние модели мира, а в качестве функции расстояния евклидово расстояние:

$$d(h) = \mathbf{E}_{h' \sim B} ||h - h'||$$

Несмотря на простоту выбраных пространства и функции расстояния, эксперименты показывают, что этого достаточно для конкурентных результатов. Для улучшения качества работы в дальнейшем потребуется рассмотреть более продвинутые варианты.

Так как подсчет в явном виде занимает большое количество вычислений, а оценка с помощью сэмплов из памяти прецедентов имеет высокий шум и может требовать большое количество сэмплов для избавления от него, предлагается использовать глубокую нейронную сеть для аппроксимации $d(h)$. Сеть $f_\theta(h)$ обучается минимизировать следующую функцию потерь:

$$L(b, \theta) = \sum_{i=1...n} \left(\left(\sum_{j=1...n} \frac{1}{n-1} (||h_i - h_j||) \right) - f_{theta}(h_i) \right)^2$$

, где b - батч из n состояний, который сэмплируется из памяти прецедентов. По батчу состояний считаются расстояния между всеми состояниями, и среднее полученных значений считается целевым. На рисунке 9 представлена схема обучения.

В памяти прецедентов находится много состояний около стартового состояния, что умненьшает вероятность получить состояния на границе изученного при случайном сэмплировании. Поэтому предлагается при обучении f_θ сэмплировать половину

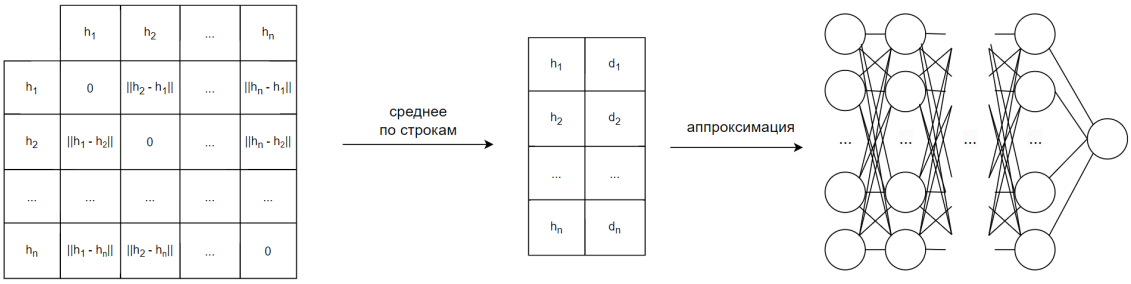

Рис. 9: Схема работы предложенного метода BatchDist. По батчу состояний h_i считаются расстояния между всеми состояниями (слева), полученные значения усредняются для каждого состояния (в середине), и далее используются для обучения глубокой нейронной сети (справа).

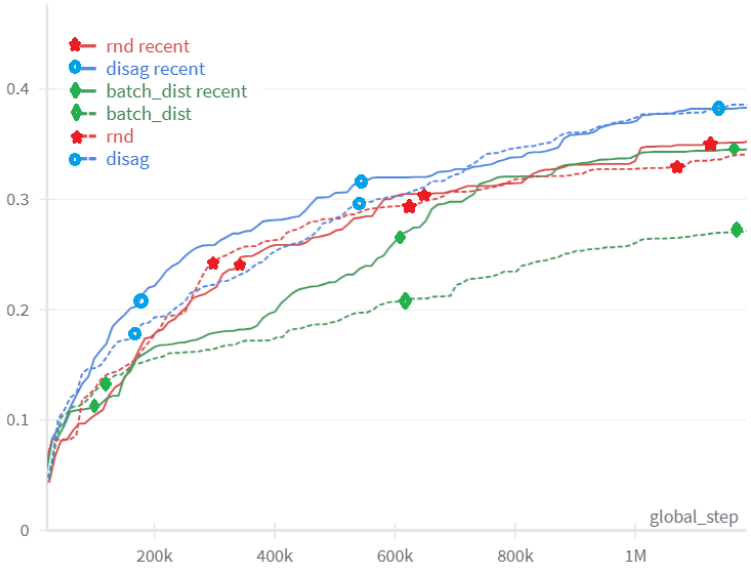

Рис. 10: Доля посещенных состояний в среде pointmaze для алгоритмов disagreement, RND и BatchDist с и без сэмплирования недавних состояний. Приписка recent в легенде означает, что половина буфера при лобучении этого алгоритма сэмплирована из последних 100000 шагов. По горизонтальной оси количесвто шагов в среде (k = 1000, M = 1000000). По вертикальной оси доля посещенных состояний.

батча равномерно из всех ранее посещенных состояний, а другую половину - из состояний, посещенных за последние k шагов (в экспериментах $k = 100000$).

В качестве внутреннего вознаграждения берется выход сети f:

$$r_i(h_t) = f_\theta(h_t)$$

5.4.1 Вычислительные эксперименты

В первую очередь необходимо проверить, что формирование батча для обучения с помощью недавно посещенных состояний дает улучшения только для алгоритма BatchDist, а не является общим улучшением для методов исследования. На рисунке 10 представлены графики исследования среды для алгоритмов disagreement, rnd и BatchDist с и без recent buffer. По графику видно, что recent buffer дает значительное улучшение только алгоритму BatchDist.

В таблице 2 представлены результаты экспериментов. По результатам видно, что BatchDist и RND проваливаются в дискретном лабиринте, а на других средах BatchDist лишь немного утупает методу disagreement, и не уступает методу RND, превосходя его в стохастичном pointmaze.

Интересно, что лучший результат BatchDist показал на частично-наблюдаемой среде. Это может происходить из-за того, что модель мира Dreamer изначально разработана для частично-наблюдаемых сред, и это может влиять на свойства пространства состояний модели h_t, что в свою очередь влияет на качество оценки расстояния в алгоритме BatchDist.

В таблице 3 представлены расчеты эффективности алгоритмов по используемой памяти и по времени обучения. Видно, что по памяти BatchDist значительно меньше, чем disagreement, и примерно в 2 меньше, чем RND. По времени обучения BatchDist также значительно меньше, чем disagreement, и немногим больше RND.

Таблица 2: Результаты экспериментов с методом BatchDist в качестве внутреннего вознаграждения. В первом столбце представлены методы: disag best - лучший результат среди всех конфигураций метода disagreement, disag gaus deter - метод disagreement с гауссовской моделью мира и детерминированным состоянием h_t в качестве целевого для моделей ансамбля. Подписи под средами означают описанный в разделе 5.1 тип среды: classic - классическая среда, stoch - стохастичная среда, discrete - среда с дискретными действиями, pomdp - частично-наблюдаемая среда. В качестве результатов представлены доли посещенных состояний среды в процентах.

-	pointmaze				antmaze
-	classic	stoch	discrete	pomdp	classic
disag best	**39**	**25**	**78**	31	**96**
disag gaus deter	**39**	**24**	71	27	**96**
rnd	37	12	29	31	92
BatchDist (our)	36	20	27	**34**	92

Таблица 3: Таблица с эффективностью методо исследования. Во втором столбце указан размер памяти, необходимый для хранения нейросейтей, используемых методами. В третьем столбце указано время обучения компонент методов исследования на одном батче.

-	память	время обучения
disagreement	2.5 GB	114 мс
RND	630 MB	19 мс
BatchDist	316 MB	23 мс

6 Заключение

Проблема исследования среды в обучении с подкреплением еще не решена полностью. Наиболее перспективным направлением, показывающим лучшие результаты, является исследование с использование обученной модели среды. В данной работе рассматривается поведение одного из наиболее эффективных методов исследования с помощью старой и новой, более эффективной, версией модели мира. Также предлягается новый метод подсчета внутреннего вознаграждения, основанный на аппроксимации среднего расстояния до ранее посещенных состояний.

Основные результаты работы:

- Было изучено поведения алгоритма plan2explore с различными архитектурными решениями на разнообразном наборе сред
- Было обнаружено, что исследование среды с помощью категориальной модели мира, показавшей эффективность в других задачах, значительно уступает более старой версии модели мира.
- Были выведены новые методы подсчета разногласия ансамбля моделей мира, однако без улучшения результатов исследования
- Был предложен новый более простой и легковесный метод подсчета внутреннего вознаграждения, показавший конкурентноспособность с другими методами

Возмыжным направлением далнейшей работы является более подробное расмотрение подсчета разногласия ансамбля для категориальной модели мира. Возможно некоторые матиматические методы позволят разложить взаимную информацию точным образом, а не оценочным, как это делают предложенные в данной работе методы.

Также перспективным направлением является рассмотрение более продвинутых пространства состояний и функции расстояния для предложенного метода BatchDist. Такое развитие метода должно улучшить результаты, так как даже простые архитектурные решения привели к конкурентным с другими методами результатами.

Список литературы

1. *Klyubin A. S.*, *Polani D.*, *Nehaniv C. L.* All Else Being Equal Be Empowered // Advances in Artificial Life / под ред. M. S. Capcarrère, A. A. Freitas, P. J. Bentley, C. G. Johnson, J. Timmis. — Berlin, Heidelberg : Springer Berlin Heidelberg, 2005. — С. 744—753. — ISBN 978-3-540-31816-3.
2. *Graves A.* Practical variational inference for neural networks // Advances in neural information processing systems. — 2011. — Т. 24.
3. High-dimensional continuous control using generalized advantage estimation / J. Schulman, P. Moritz, S. Levine, M. Jordan, P. Abbeel // arXiv preprint arXiv:1506.02438. — 2015.
4. Human-level control through deep reinforcement learning / V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski [и др.] // nature. — 2015. — Т. 518, № 7540. — С. 529—533.
5. Asynchronous methods for deep reinforcement learning / V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, K. Kavukcuoglu // International conference on machine learning. — PMLR. 2016. — С. 1928—1937.
6. Mastering the game of Go with deep neural networks and tree search / D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot [и др.] // nature. — 2016. — Т. 529, № 7587. — С. 484—489.
7. Unifying count-based exploration and intrinsic motivation / M. Bellemare, S. Srinivasan, G. Ostrovski, T. Schaul, D. Saxton, R. Munos // Advances in neural information processing systems. — 2016. — Т. 29.
8. Curiosity-driven exploration by self-supervised prediction / D. Pathak, P. Agrawal, A. A. Efros, T. Darrell // International conference on machine learning. — PMLR. 2017. — С. 2778—2787.
9. Proximal policy optimization algorithms / J. Schulman, F. Wolski, P. Dhariwal, A. Radford, O. Klimov // arXiv preprint arXiv:1707.06347. — 2017.
10. Successor features for transfer in reinforcement learning / A. Barreto, W. Dabney, R. Munos, J. J. Hunt, T. Schaul, H. P. van Hasselt, D. Silver // Advances in neural information processing systems. — 2017. — Т. 30.

11. Diversity is all you need: Learning skills without a reward function / B. Eysenbach, A. Gupta, J. Ibarz, S. Levine // arXiv preprint arXiv:1802.06070. — 2018.

12. Exploration by random network distillation / Y. Burda, H. Edwards, A. Storkey, O. Klimov // arXiv preprint arXiv:1810.12894. — 2018.

13. Large-scale study of curiosity-driven learning / Y. Burda, H. Edwards, D. Pathak, A. Storkey, T. Darrell, A. A. Efros // arXiv preprint arXiv:1808.04355. — 2018.

14. Soft actor-critic algorithms and applications / T. Haarnoja, A. Zhou, K. Hartikainen, G. Tucker, S. Ha, J. Tan, V. Kumar, H. Zhu, A. Gupta, P. Abbeel [и др.] // arXiv preprint arXiv:1812.05905. — 2018.

15. Dream to control: Learning behaviors by latent imagination / D. Hafner, T. Lillicrap, J. Ba, M. Norouzi // arXiv preprint arXiv:1912.01603. — 2019.

16. Grandmaster level in StarCraft II using multi-agent reinforcement learning / O. Vinyals, I. Babuschkin, W. M. Czarnecki, M. Mathieu, A. Dudzik, J. Chung, D. H. Choi, R. Powell, T. Ewalds, P. Georgiev [и др.] // Nature. — 2019. — Т. 575, № 7782. — С. 350—354.

17. *Pathak D.*, *Gandhi D.*, *Gupta A.* Self-supervised exploration via disagreement // International conference on machine learning. — PMLR. 2019. — С. 5062—5071.

18. *Shyam P.*, *Jaśkowski W.*, *Gomez F.* Model-based active exploration // International conference on machine learning. — PMLR. 2019. — С. 5779—5788.

19. *Machado M. C.*, *Bellemare M. G.*, *Bowling M.* Count-based exploration with the successor representation // Proceedings of the AAAI Conference on Artificial Intelligence. Т. 34. — 2020. — С. 5125—5133.

20. Mastering atari with discrete world models / D. Hafner, T. Lillicrap, M. Norouzi, J. Ba // arXiv preprint arXiv:2010.02193. — 2020.

21. Planning to explore via self-supervised world models / R. Sekar, O. Rybkin, K. Daniilidis, P. Abbeel, D. Hafner, D. Pathak // International Conference on Machine Learning. — PMLR. 2020. — С. 8583—8592.

22. Mastering diverse domains through world models / D. Hafner, J. Pasukonis, J. Ba, T. Lillicrap // arXiv preprint arXiv:2301.04104. — 2023.

23. Mastering the unsupervised reinforcement learning benchmark from pixels / S. Rajeswar, P. Mazzaglia, T. Verbelen, A. Piché, B. Dhoedt, A. Courville, A. Lacoste // International Conference on Machine Learning. — PMLR. 2023. — C. 28598—28617.

24. Planning Goals for Exploration / E. S. Hu, R. Chang, O. Rybkin, D. Jayaraman // arXiv preprint arXiv:2303.13002. — 2023.