

## 7. ЛЕКЦИЯ. Ценностно-ориентированный подход 2

### *Distributional RL (Распределительное обучение с подкреплением)*

*Идея Distributional (распределительного) подхода*

Задача RL такова, что в среде содержится в том числе неподконтрольная агенту стохастика: алеаторическая неопределённость (aleatoric uncertainty измеряет то, что вы не можете понять из данных.). Агент, предсказывающий, что он получит в будущем в среднем суммарную награду 6, на самом деле может получить, например, только -10 или 10, просто последний исход случится с вероятностью 0.8, а первый — 0.2.

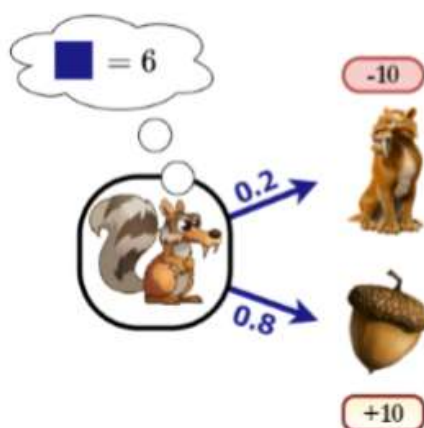


Рис. 6.1

Помимо прочего, это означает, что часто агенту приходится рисковать: например, теоретически возможна ситуация, когда агент с малой вероятностью получает гигантскую награду, и тогда оптимальный агент на практике будет постоянно получать, например, какой-то штраф, компенсирующийся редко выпадающими мегаудачами. Вся эта информация заложена в распределении награды  $R(\mathcal{T})$  как случайной величины.

В Distributional (распределительном)-подходе предлагается учить не среднее будущей награды, а всё распределение будущей награды как случайной величины. Складывается эта неопределённость как из неподконтрольной агенту стохастики — его собственных будущих выборов действий — так и неподконтрольной, переходов (и награды, если рассматривается формализм со случайной функцией награды). Среднее есть лишь одна из статистик этого распределения.

Здесь стоит заранее оговориться о противоречиях, связанных с этой идеей. Обсуждение этой темы в первую очередь мотивировано эмпирическим превосходством Distributional-подхода по сравнению с алгоритмами, учащими только среднее, однако с теоретической точки зрения ясного обоснования этого эффекта нет. Даже наоборот: мы далее встретим теоретические результаты,

показывающие эквивалентность Distributional-алгоритмов с обычными в рамках табличного сеттинга (среда, в которой происходит действие).

Одно гипотетическое объяснение преимущества distributional-подхода в нейросетевом сеттинге, когда будущая награда предсказывается сложной параметрической моделью, может быть следующая: обучая модель предсказывать не только среднее награды, но и другие величины (другие статистики), сильно связанные по смыслу со средней наградой, в модель отправляются более информативные градиенты.

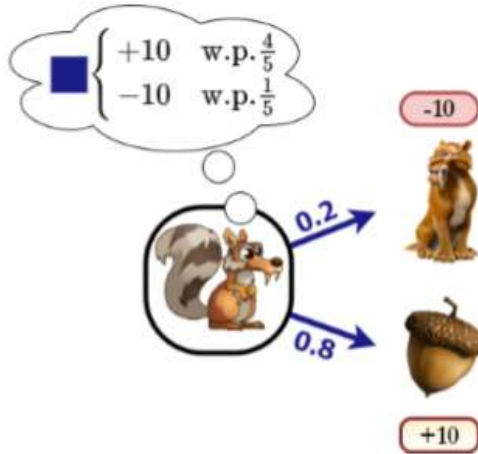


Рис. 6.2. w.p. – вероятность выигрыша

### *Z-функция*

Определение: Для данного MDP оценочной функцией в distributional форме (distributional stateaction value function – функция значения действия состояния распределения) для стратегии  $\pi$  называется случайная величина, обусловленная на пару состояние действие  $s, a$  и определяющаяся как reward-to-go (награда на вынос) для такого старта:

$$Z^\pi(s, a) \stackrel{\text{c.d.f.}}{:=} R(\mathcal{T}), \quad \mathcal{T} \sim \pi \mid s_0 = s, a_0 = a$$

Введённая так называемая Z-функция является для каждой пары  $s, a$  случайной величиной. Во-первых, это скалярная случайная величина, соответственно, она задаётся некоторым распределением на  $\mathbb{R}$ , во-вторых, как и для любой случайной величины, существенно, на что она обуславливается. Запись  $Z^\pi(s, a)$  предполагает, что мы сидим в состоянии  $s$  и выполнили действие  $a$ , после чего «бросаем кости» для сэмплирования случайной величины; нас, вообще говоря, будет интересовать её функция распределения (cumulative distribution function – кумулятивная функция распределения, c. d. f.):

$$F_{Z^\pi(s, a)}(x) := P(Z^\pi(s, a) \leq x)$$

Поскольку зачастую распределение  $Z^\pi(s, a)$  — дискретное или вообще вырожденное (принимающее с вероятностью 1 только какое-то одно значение).

Таким образом, пространство всевозможных  $Z$ -функций имеет такой вид:

$$Z^\pi(s, a) \in \mathcal{S} \times \mathcal{A} \rightarrow P(\mathbb{R}),$$

где  $P(\mathbb{R})$  — пространство скалярных случайных величин.

Надпись c.d.f. над равенством здесь и далее означает, что слева и справа стоят случайные величины. Очень важно, что случайные величины справа и слева в подобных равенствах обусловлены на одну и ту же информацию: справа, как и слева, стоит случайная величина, обусловленная на  $s, a$ . Случайная величина здесь задана процессом генерации: сначала генерируется случайная траектория  $\mathcal{T}$  при заданных  $s_0 = s, a_0 = a$  (это по определению MDP эквивалентно последовательному сэмплированию  $s_1, a_1, s_2, \dots$ ), затем от этой случайной величины считается детерминированная функция  $R(\mathcal{T})$ . Запись c.d.f. означает, что  $Z^\pi(s, a)$  имеет в точности то же распределение, что и случайная величина, генерируемая процессом, описанном справа.

По определению:

Утверждение:

$$Q^\pi(s, a) = \mathbb{E}Z^\pi(s, a)$$

Утверждение: В терминальных состояниях для всех действий  $Z^\pi(s, a)$  есть вырожденная случайная величина, с вероятностью 1 принимающая значение ноль.

Допустим, мы сидим в состоянии и выполнили действие ■. Как будет выглядеть  $Z^\pi(s, \blacksquare)$  MDP и стратегии  $\pi$  с рисунка,  $\gamma = 0.5$ ?

Нас ждёт два источника случайности: сначала среда кинет нас или в состояние В, или в состояние С, затем мы случайно будем определять своё следующее действие. Всего нас ждёт 4 возможных исхода. Для каждого мы можем посчитать его вероятность и получаемый reward-to-go. Итого  $Z^\pi(s, \blacksquare)$  — дискретное распределение с 4 исходами:

Исход $s'$	Исход $a'$	Вероятность	reward-to-go
В	■	0.3	$1 + 2\gamma$
В	■	0.3	1
С	■	0.1	$\gamma$
С	■	0.3	0

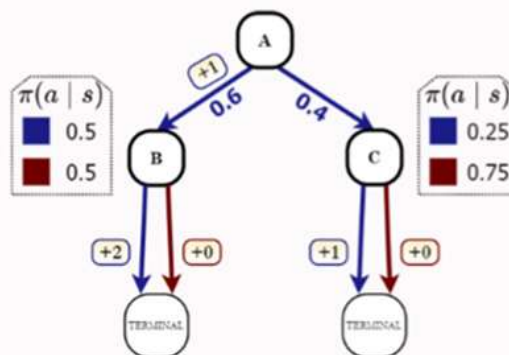


Рис. 6.3. Пример

### *Distributional (распределительная)-форма уравнения Беллмана*

Заметим, что в доказательстве уравнений Беллмана, мы ссылаемся на то, что для reward-to-go (награда на вынос) любых траекторий верно рекурсивное соотношение. После этого мы берём по траекториям мат.ожидание слева и справа, получая традиционное уравнение Беллмана. Ясно, что мы могли бы вместо среднего взять любую другую статистику от случайной величины (дисперсию, медианы, другие квантили...), а, вообще говоря, верно совпадение

левой и правой части по распределению. Иначе говоря, можно зачеркнуть символ мат.ожидания из уравнения Беллмана для получения более общего утверждения.

Теорема — Уравнение Беллмана в Distributional-форме:

$$\mathcal{Z}^\pi(s, a) \stackrel{\text{c.d.f.}}{=} r(s, a) + \gamma \mathcal{Z}^\pi(s', a'), \quad s' \sim p(s' | s, a), a' \sim \pi(a' | s')$$

Немного остановимся на этом уравнении и обсудим, что тут написано. Во-первых, необходимо пояснить, что данное уравнение есть переформулировка (другая нотация) используемых определений. Reward-to-go (Награда на вынос)  $R(\mathcal{T})$  — детерминированная функция от заданной траектории  $\mathcal{T}$ ,  $Z^\pi$  — по сути тоже самое, только траектория рассматривается как случайная величина (а параметры  $s, a$  указывают на начальные условия генерации траекторий). И слева, и справа в уравнении стоят случайные величины, зависящие от  $s, a$ ; равенство означает, что они имеют одинаковые распределения. Иными словами, слева и справа записаны два процесса генерации одной и той же случайной величины. Мы можем бросить кость  $Z^\pi(s, a)$  (случайная величина слева), а можем — сначала  $s'$ , потом  $a'$ , затем  $Z^\pi(s', a')$  и выдать исход  $r(s, a) + \gamma Z^\pi(s', a')$  (случайная величина справа), и эти две процедуры порождения эквивалентны.

Пример: Уравнение Беллмана всё ещё связывает «содержимое» Z-функции через неё же саму, раскрывая дерево на один шаг. Эти уравнения теперь затруднительно выписать аналитически, поскольку теперь «компоненты» распределения  $Z(s, a)$  есть перевзвешанные на вероятности переходов и выборов действий (и подправленные по значению на дисконт (скидка) фактор и смещённые на награду за шаг)  $Z(s', a')$  для всевозможных  $s', a'$ .

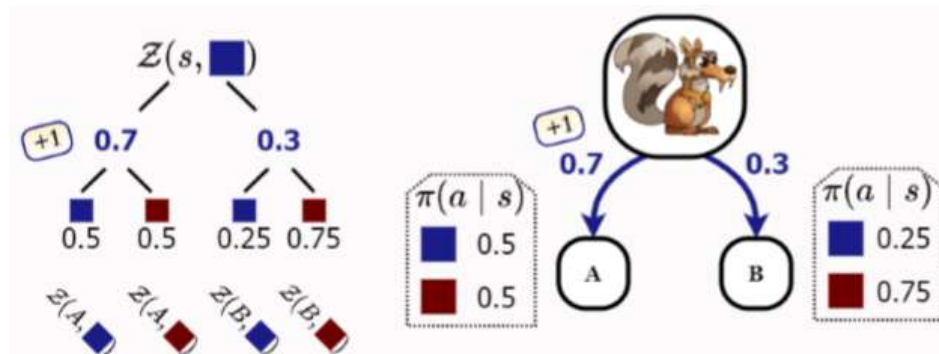


Рис. 6.4. Пример

Подобные уравнения называются recursive distributional equations (рекурсивные уравнения распределения) и рассматриваются математикой в одном из разделов теории вероятности

*Distributional Policy Evaluation (Оценка распределительной политики)*

Будем строить аналог Policy Evaluation (оценку политики) для distributional

(распределенной)-формы оценочной функции. Иными словами, мы хотим чисто теоретический алгоритм, позволяющий для данного MDP и данной стратегии  $\pi$  посчитать распределение  $Z^\pi(s, a)$ . MDP пока считаем полностью известным (распределение  $p(s'|s, a)$  считаем данным). Действуем в полной аналогии с обычными уравнениями: начнём с ввода оператора Беллмана.

Определение: Для данного MDP и стратегии  $\pi$  будем называть оператором Беллмана в distributional форме оператор  $\mathfrak{B}_D$ , действующий из пространства  $Z$ -функций в пространство  $Z$ -функций, задающий случайную величину для  $s, a$  на выходе оператора как правую часть distributional уравнения Беллмана

$$[\mathfrak{B}_D \mathcal{Z}](s, a) \stackrel{\text{c.d.f.}}{:=} r(s, a) + \gamma \mathcal{Z}(s', a'), \quad s' \sim p(s' | s, a), a' \sim \pi(a' | s')$$

По определению, истинное  $Z^\pi$  будет неподвижной точкой такого оператора:

$$Z^\pi = \mathfrak{B}_D Z^\pi$$

Нас интересует вопрос о сходимости метода простой итерации. Что это означает? Если на  $k$ -ой итерации мы храним большую табличку, где для каждой пары  $s, a$  хранится целиком и в точности всё распределение  $Z_k(s, a)$ , то на очередном шаге для всех пар  $s, a$  происходит обновление

$$\mathcal{Z}_{k+1}(s, a) \stackrel{\text{c.d.f.}}{:=} r(s, a) + \gamma \mathcal{Z}_k(s', a'),$$

где вероятности случайных величин  $s' \sim p(s'|s, a), a' \sim \pi(a'|s')$  мы знаем и потому можем полностью посчитать свёртку распределений  $Z_k(s', a')$  для всевозможных пар следующих  $s', a'$ .

Чтобы показать сходимость такой процедуры, хочется в аналогии с традиционным случаем доказать сжимаемость оператора  $\mathfrak{B}_D$ . Однако, обсуждение сжимаемости имеет смысл только при заданной метрике, а в данном случае даже для конечных пространств состояний и пространств действий пространство  $Z$ -функций бесконечномерно, поскольку бесконечномерно  $P(\mathbb{R})$ . Нам нужна метрика в таком пространстве, и, внезапно, от её выбора будет зависеть ответ на вопрос о сжимаемости.

Определение: Пусть  $\mathcal{D}$  — метрика в пространстве  $P(\mathbb{R})$ . Тогда её максимальной формой (maximal form) будем называть следующую метрику в пространстве  $Z$ -функций:

$$\mathcal{D}^{\max}(\mathcal{Z}_1, \mathcal{Z}_2) := \sup_{s \in \mathcal{S}, a \in \mathcal{A}} \mathcal{D}(\mathcal{Z}_1(s, a), \mathcal{Z}_2(s, a))$$

Теорема: Для любой метрики  $\mathcal{D}$  в пространстве  $P(\mathbb{R})$  её максимальная форма  $\mathcal{D}^{\max}$  есть метрика в пространстве  $Z$ -функций.

Соответственно, вопрос о выборе метрики в пространстве  $Z$ -функций

сводится к вопросу о метрике в пространстве скалярных случайных величин.

Определение: Для скалярной случайной величины  $X$  с функцией распределения  $F_X(x): \mathbb{R} \rightarrow [0,1]$  квантильной функцией называется

$$F_X^{-1}(\omega) := \inf\{x \in \mathbb{R} \mid F_X(x) \geq \omega\}$$

Значение этой функции  $F_X^{-1}(\omega)$  в точке  $\tau \in (0,1)$  будем называть  $\tau$ -квантилем.

Определение: Для  $1 \leq p \leq +\infty$  для двух случайных скалярных величин  $X, Y$  с функциями распределения  $F_X$  and  $F_Y$  соответственно расстоянием Вассерштайна (Wasserstein distance) называется

$$\mathcal{W}_p(X, Y) := \left( \int_0^1 |F_X^{-1}(\omega) - F_Y^{-1}(\omega)|^p d\omega \right)^{\frac{1}{p}}$$

$$\mathcal{W}_\infty(X, Y) := \sup_{\omega \in [0,1]} |F_X^{-1}(\omega) - F_Y^{-1}(\omega)|$$

Теорема — Эквивалентная форма  $\mathcal{W}_1$ :

$$\mathcal{W}_1(X, Y) = \int_{\mathbb{R}} |F_X(x) - F_Y(x)| dx$$

Пример: Расстояние  $\mathcal{W}_1$  между двумя распределениями неспроста имеет второе название Earth Moving Distance (расстояние перемещения земли). Аналогия такая: нам даны две кучи песка. Объём песка в кучах одинаков, но у них разные конфигурации, они «насыпаны» по-разному. Чтобы перенести каждую песчинку массы  $m$  на расстояние  $x$ , нам нужно затратить «работы» объёмом  $mx$ . Расстояние Вассерштайна  $\mathcal{W}_1$  замеряет, какое минимальное количество работы нужно совершить, чтобы перевести конфигурацию первой кучи песка во вторую кучу; объём песка в каждой куче одинаков. Для дискретных распределений, когда функции распределения (и, соответственно, квантильные функции) — «ступеньки», минимальная работа полностью соответствует площади между функциями распределений

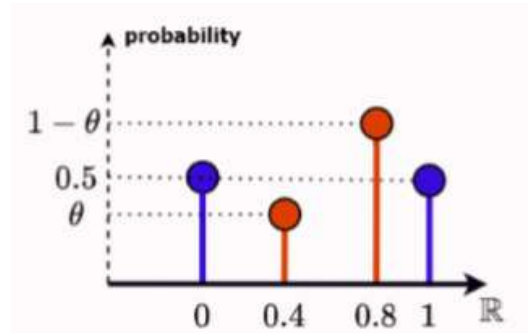


Рис. 6.5.

Посчитаем  $\mathcal{W}_1$  между двумя следующими распределениями. Первое распределение (синие на картинке) — честная монетка с исходами 0 и 1. Вторая случайная величина (красная на картинке) принимает значение 0.4 с вероятностью  $\theta < 0.5$  и 0.8 с вероятностью  $1 - \theta$ . Рассуждаем так: давайте «превратим» вторую кучу песка в первую. Посмотрим на песок объёма  $\theta$  в точке 0.4. Куда его переносить? Наверное, в точку 0, куда его тащить ближе. Перенесли; совершили работы объёмом  $0.4\theta$ . Посмотрим на песок объёма  $1 - \theta$  в точке 0.8. Его удобно тащить в точку 1, но там для получения первой конфигурации нужно только 0.5 песка. Поэтому 0.5 песка из точки 0.8 мы можем перевести в точку 1, совершив работу  $0.2 \cdot 0.5$ , а оставшийся объём  $1 - \theta - 0.5$  придётся переводить в точку 0, совершая работу  $0.8(0.5 - \theta)$ . Итого расстояние Вассерштайна равно:

$$\mathcal{W}_1 = 0.4\theta + 0.8(0.5 - \theta) + 0.1$$

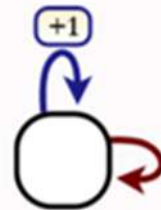
Утверждение: Максимальная форма метрики Вассерштайна  $\mathcal{W}_p^{\max}$  есть метрика в пространстве Z-функций

Теорема: По метрике  $\mathcal{W}_p^{\max}(Z_1, Z_2)$  оператор  $\mathfrak{B}_D$  является сжимающим.

**Пример** Попробуем найти  $Z^\pi$  для случайной  $\pi$  (выбирающей из двух действий всегда равновероятно) для MDP с рисунка и  $\gamma = 0.5$ .

Сначала попробуем понять, в каких границах может лежать наша награда за весь эпизод. Если, например, мы всё время выбираем  $\blacksquare$ , то получим в итоге ноль; меньше, понятнo, получить нельзя. Если же мы всё время выбираем  $\blacksquare$ , то получим в итоге  $1 + \gamma + \gamma^2 + \dots + \gamma^{T-1} = \frac{1 - \gamma^T}{1 - \gamma}$ . Значит, вероятные исходы размазаны на отрезке  $[0, 2]$ .

Попробуем посмотреть на  $Z^\pi(\blacksquare)$ . По определению, мы предполагаем, что на первом шаге выбирается действие  $\blacksquare$ , и значит, на первом шаге мы гарантированно получим +0. Тогда, проводя аналогичные рассуждения, можно заключить, что дальнейшая возможная награда лежит в отрезке  $[0, 1]$ . Но что именно это за распределение? Можно рассмотреть распределение случайной величины  $\sum_{t=0}^{T-1} \gamma^t r_t \mid a_0 = \blacksquare$  не при  $T = +\infty$ , а при меньших  $T$ . Например, при  $T = 1$  мы получим +0,



затем в качестве  $r_1$  с равными вероятностями получим +0 или  $+\frac{1}{2}$ . Получится равновероятное распределение с исходами 0,  $+\frac{1}{2}$ . При  $T = 2$  мы получим уже равновероятное распределение с исходами 0,  $+\frac{1}{4}$ ,  $+\frac{1}{2}$ ,  $+\frac{3}{4}$ . Продолжая рассуждение дальше, можно увидеть, что при  $T \rightarrow +\infty$  распределение продолжает равномерно размазывать вероятности по  $[0, 1]$ . Видимо, в пределе получится просто равномерное распределение на  $[0, 1]$ . Как можно строго доказать, что это правильный ответ?

Попробуем подставить в уравнения Беллмана ( ) в качестве  $Z^\pi(\blacksquare)$  равномерное распределение на отрезке  $[0, 1]$ , а в качестве  $Z^\pi(\blacksquare)$  равномерное распределение на отрезке  $[1, 2]$  (так как тут мы гарантированно получим +1 на первом же шаге). Что мы получим? Рассмотрим первое уравнение:

$$Z^\pi(\blacksquare) \stackrel{\text{c.d.f.}}{=} \underbrace{+1}_r + \underbrace{0.5}_{\gamma} Z^\pi(a'),$$

где  $a'$  — случайная величина, с равной вероятностью принимающая оба возможных значения.

Вот мы выбрали  $\blacksquare$ : с одной стороны левая часть уравнения говорит, что мы получим равномерное распределение на  $[1, 2]$ . С другой стороны правая часть уравнения рассматривает «одношаговое приближение»: мы точно получим +1, затем кинем кубик; с вероятностью 0.5 выберем на следующем шаге  $\blacksquare$  и получим равномерное из  $[1, 2]$ , а с вероятностью 0.5 выберем  $\blacksquare$  и получим равномерное из  $[0, 1]$ . Значит, начиная со второго шага мы получим сэмпл из равномерного на  $[0, 2]$ ; он будет дисконтирован на гамму и получится сэмпл из  $[0, 1]$ ; дальше мы его сдвинем на +1, который мы получили на первом шаге, и в итоге как раз получится равномерное из  $[1, 2]$ ! Сошлось; в левой и правой стороне уравнения получается одно и то же распределение! Аналогично проверяется, что сходится второе distributional-уравнение. Из доказанного нами свойства сжатия следует, что это решение — единственное, и, значит, является истинной  $Z^\pi$ .

Утверждение: Существует единственная функция  $\mathcal{S} \times \mathcal{A} \rightarrow P(\mathbb{R})$ , являющаяся решением уравнений Беллмана в Distributional (распределительный), и метод простой итерации сходится к ней из любого начального приближения по метрике Вассерштайна.

В реальности нам с Вассерштайном обычно неудобно работать, и мы предпочитаем более удобные дивергенции, например, KL-дивергенцию (дивергенция Кульбака-Лейблера – является мерой того, насколько одно распределение вероятностей отличается от второго эталонного распределения вероятностей). Чисто теоретически мы можем задаться вопросом, как ведёт себя расстояние от  $Z_k := \mathfrak{B}_D^k Z_0$  до истинной  $Z^\pi$  в терминах KL-дивергенции, то есть стремится ли оно хотя бы к нулю? Оказывается, не просто не стремится, а вообще полное безобразие происходит: KL-дивергенция не умеет адекватно мерить расстояние между распределениями с несовпадающим доменом (disjoint support).

Теорема: Расстояние между  $\mathfrak{B}_D^k Z_0$  и истинным  $Z^\pi$  по максимальной форме KL-дивергенции может быть равно бесконечности для всех  $k$ .

Пусть  $\mathfrak{B}$  — обычный оператор Беллмана из пространства Q-функций в пространство Q-функций, а  $\mathfrak{B}_D$ , как и раньше, оператор Беллмана из пространства Z-функций в пространство Z-функций.

Теорема: Пусть инициализации  $Z_0$  и  $Q_0$  удовлетворяют  $\mathbb{E}Z_0 = Q_0$ , и рассматривается два метода простой итерации:

$$\begin{aligned} Q_k &:= \mathfrak{B}^k Q_0 \\ Z_k &:= \mathfrak{B}_D^k Z_0 \end{aligned}$$

Тогда:

$$Q_k = \mathbb{E}Z_k$$

*Distributional Value Iteration (Итерация распределения ценности)*

По аналогии с традиционным случаем, можно ввести оптимальную оценочную функцию в distributional (распределительной)-форме как Z-функцию оптимальных стратегий

$$\mathcal{Z}^*(s, a) \stackrel{\text{c.d.f.}}{:=} Z^{\pi^*}(s, a)$$

С уравнением оптимальности Беллмана для  $Z^*$  тоже внезапно есть тонкости. Пропустим вычислительные доказательства.

Определение: Введём оператор оптимальности Беллмана в distributional (распределительной)-форме  $\mathfrak{B}_D^*$ :

$$[\mathfrak{B}_D^* \mathcal{Z}](s, a) \stackrel{\text{c.d.f.}}{:=} r(s, a) + \gamma \mathcal{Z}(s', \arg\max_{a'} \mathbb{E} \mathcal{Z}(s', a')), \quad s' \sim p(s' | s, a)$$

Пусть также  $\mathfrak{B}^*$  — обычный оператор оптимальности Беллмана из

пространства Q-функций в пространство Q-функций.

Теорема: Пусть инициализации  $Z_0$  и  $Q_0$  удовлетворяют  $\mathbb{E}Z_0 = Q_0$  и рассматривается два метода простой итерации:

$$\begin{aligned} Q_k &:= (\mathfrak{B}^*)^k Q_0 \\ Z_k &:= (\mathfrak{B}_D^*)^k Z_0 \end{aligned}$$

Тогда:

$$Q_k = \mathbb{E}Z_k$$

Итак, мы показали, что в методе простой итерации с оператором  $\mathfrak{B}_D^*$  средние движутся точно также, как и  $Q^*$  в обычном подходе. Однако, хвосты распределений при этом могут вести себя довольно нестабильно.

Теорема: Оператор  $\mathfrak{B}_D^*$  может не являться непрерывным.

Утверждение: Оператор  $\mathfrak{B}_D^*$  может не являться сжимающим.

Пояснение. По определению, любой сжимающий оператор, и, значит, обязан быть непрерывным.

#### *Категориальная аппроксимация Z-функций*

Определение: Зададимся набором атомов  $r^{\min} = z_0 < z_1 < z_2 \dots < z_A = r^{\max}$ , где  $A + 1$  — число атомов. Обозначим семейство категориальных распределений  $\mathcal{C} \subset P(\mathbb{R})$  как множество дискретных распределений на домене  $\{z_0, z_1 \dots z_A\}$ : если  $Z(s, a) \in \mathcal{C}$ , то

Типично атомы образуют просто равномерную сетку, для задания которой требуется три гиперпараметра: число атомов, минимальное и максимальное значение награды. Распространённый дефолтный вариант для Atari игр — 51 атом на отрезке  $[-10, 10]$ . В честь такой параметризации (categorical with 51 atoms — категориальный с 51 атомом) иногда алгоритм Categorical DQN, к построению которого мы приближаемся, называют c51.

Итак, для каждой пары  $s, a$  мы будем хранить в табличке  $A + 1$  неотрицательное число  $p_0, p_1 \dots p_A$ , суммирующиеся в единицу, и полагать, что  $A + 1$  узлов нашей сетки  $z_0, z_1 \dots z_A$  являются единственно возможными исходами будущей награды. Такова наша аппроксимация.

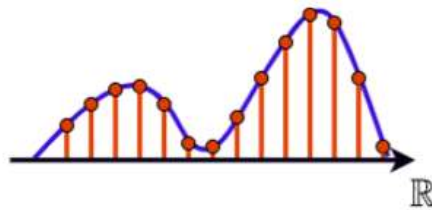


Рис. 6.6.

Возникает следующая проблема: мы, в принципе, можем посчитать распределения  $\mathfrak{B}_D^* Z$ , но что, если оно «непопадёт» в рассматриваемое семейство

аппроксимаций? То есть что, если для какой-то пары  $s, a$   $[\mathfrak{B}_D^* Z](s, a) \notin \mathcal{C}$ , то есть что, если оно не является категориальным распределением на домене  $\{z_0, z_1 \dots z_A\}$ ? Нам придётся как-то проецировать полученный результат на нашу сетку...

Утверждение: В табличном сеттинге если  $Z(s, a) \in \mathcal{C}$  для всех  $s, a$ , то  $[\mathfrak{B}_D^* Z](s, a)$  — дискретное распределение с конечным множеством исходов.

Значит, нам нужно научиться проецировать лишь дискретные распределения.

Определение: Введём оператор проекции  $\Pi$ , действующий из пространства произвольных дискретных распределений в  $\mathcal{C}$  следующим образом. Пусть  $\tilde{Z}(s, a)$  — произвольное дискретное распределение с исходами  $\tilde{z}_i$  с соответствующими вероятностями  $\tilde{p}_i$  (суммирующимися в единицу). Изначально инициализируем все  $p_i$  для результата работы оператора нулями.

Дальше перебираем исходы  $\tilde{z}_i$ ; если очередной исход меньше  $r^{min} = z_0$ , всю его вероятностную массу отправляем в  $p_0$ , то есть увеличиваем  $p_0$  на  $\tilde{z}_i$ . Аналогично поступаем если  $\tilde{z}_i > r^{max} = z_A$ . В остальных случаях найдётся два соседних атома нашей сетки, такие что  $z_j \leq \tilde{z}_i \leq z_{j+1}$ . Распределим вероятностную массу между ними обратно пропорционально расстоянию до них, а то есть:

$$p_j \leftarrow p_j + \frac{z_{j+1} - \tilde{z}_i}{z_{j+1} - z_j} \tilde{p}_i$$

$$p_{j+1} \leftarrow p_{j+1} + \frac{\tilde{z}_i - z_j}{z_{j+1} - z_j} \tilde{p}_i$$

Наш метод простой итерации теперь «подкорректированный», после каждого шага мы применяем проекцию:

$$Z_{k+1} \stackrel{\text{c.d.f.}}{:=} \Pi \mathfrak{B}_D^* Z_k,$$

где применение  $\Pi$  к  $Z$ -функции означает проецирование всех распределений  $Z(s, a)$  для всех  $s, a$ .

**Теорема:** Пусть  $Z(s, a)$  дискретно и выдаёт исходам вне отрезка  $[r^{min}, r^{max}]$  нулевую вероятность. Тогда оператор проекции сохраняет мат.ожидание,  $\forall Z, s, a$ :

$$\mathbb{E} \Pi Z(s, a) = \mathbb{E} Z(s, a)$$

*Categorical DQN (Категориальный)*

Попробуем составить уже полностью практический алгоритм. Во-первых, обобщим алгоритм на случай произвольных пространств состояний, моделируя  $Z_\theta \approx Z^*$  (а точнее — её распределение) при помощи нейросети с параметрами  $\theta$ .

Для каждой пары  $s, a$  такая нейросеть выдаёт  $A + 1$  неотрицательное число  $p_0(s, a, \theta), p_1(s, a, \theta) \dots p_A(s, a, \theta)$ , суммирующиеся в единицу, и мы предполагаем категориальную аппроксимацию

$$P(Z_\theta(s, a) = z_i) := p_i(s, a, \theta).$$

Как и в DQN, считаем, что у нас есть таргет-сеть с параметрами  $\theta^-$  — Z-функция  $Z_\theta$  — с предыдущего (условно,  $k$ -го) шага метода простой итерации, а мы хотим обучать  $\theta$  так, чтобы получить Z-функцию на  $k + 1$ -ом шаге: наша цель — выучить  $\mathfrak{B}_D^* Z_{\theta^-}$ .

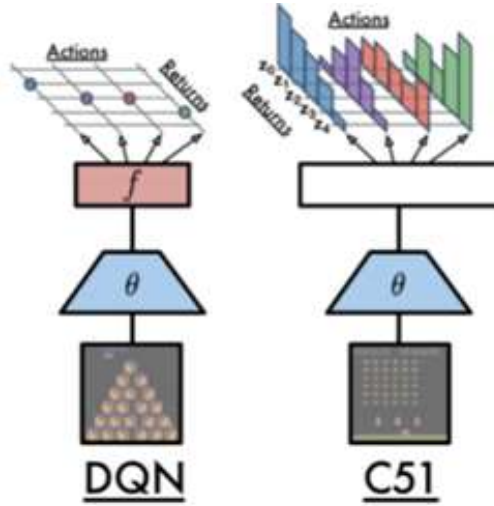


Рис. 6.7.

В model-free (без модели) режиме, без доступа к функции переходов, мы не то чтобы посчитать  $\mathfrak{B}_D^* Z_{\theta^-}$  не можем, нам даже недоступна большая часть информации о нём. Для данной пары  $s, a$  из реплей буфера мы можем получить только один сэмпл  $s' \sim p(s'|s, a)$ , и нам нужен какой-то «аналог» метода временных разностей.

Первое соображение: мы умеем сэмплировать из  $[\mathfrak{B}_D^* Z_{\theta^-}](s, a)$ . Действительно: пусть дано  $s, a$ ; берём сэмпл  $s'$  из, например, буфера; смотрим на нашу таргет сеть  $Z_{\theta^-}(s', a')$  для всех действий  $a'$ , считаем для каждого действия  $a'$  мат.ожидание (для ситуации  $Z_{\theta^-}(s', a') \in \mathcal{C}$  это, очевидно, не проблема) и выбираем «наилучшее» действие  $a' = \arg \max_{a'} \mathbb{E} Z_{\theta^-}(s', a')$ . Выбираем такое  $a'$ , и дальше у нас есть даже не сэмпл, а целая компонента искомого распределения  $[\mathfrak{B}_D^* Z_{\theta^-}](s, a)$  в виде распределения  $r(s, a) + \gamma Z_{\theta^-}(s', a')$ , которое мы и будем использовать в качестве таргета.

Итак, пусть  $T := (s, a, r, s')$  — четвёрка из буфера. Введём целевую переменную (таргет) следующим образом:

$$y(T) \stackrel{\text{c.d.f.}}{:=} r + \gamma Z_{\theta^-}(s', \arg \max_{a'} \mathbb{E} Z_{\theta^-}(s', a'))$$

где  $s'$  в формуле берётся из  $\mathbb{T}$ , то есть взято из буфера. Такой таргет является дискретным распределением  $s$ , очевидно,  $A$  атомами, но из-за того, что мы взяли лишь один сэмпл  $s'$ , он является лишь компонентой из  $[\mathfrak{B}_D^* Z_{\theta^-}](s, a)$ .

Второе соображение: допустим, для данной пары  $s, a$  мы сможем оптимизировать следующий функционал для некоторой дивергенции  $\mathcal{D}$ , используя лишь сэмплы из первого распределения:

$$\mathcal{D}([\mathfrak{B}_D^* Z_{\theta^-}](s, a) \parallel Z_{\theta}(s, a)) \rightarrow \min_{\theta}$$

Если бы мы могли моделировать произвольные  $Z$ -функции, минимум достигался бы в нуле на совпадающих распределениях, и наша цель была бы достигнута. Однако мы ограничены нашим аппроксимирующим категориальным семейством  $\mathcal{C}$  и при оптимизации такого функционала даже чисто теоретически мы получим лишь проекцию на  $\mathcal{C}$ ; здесь возникает вопрос, а не потеряем ли мы свойство сохранения мат.ожидания. Мы могли бы приближать наше распределение сразу к «хорошей» проекции:

$$\mathcal{D}([\Pi \mathfrak{B}_D^* Z_{\theta^-}](s, a) \parallel Z_{\theta}(s, a)) \rightarrow \min_{\theta}$$

Тогда мы будем учить категориальное распределение с сохранённым мат.ожиданием.

### Теорема

$$[\Pi \mathfrak{B}_D^* Z_{\theta^-}](s, a) \stackrel{\text{c.d.f.}}{=} \text{Py}(\mathbb{T}), \quad s' \sim p(s' \mid s, a)$$

Разберёмся, что здесь написано. Мы можем (теоретически) посчитать полностью одношаговую аппроксимацию  $\mathfrak{B}_D^* Z_{\theta^-}$  и спроецировать полученное распределение (случ. величина слева); а можем взять случайный  $s'$ , посмотреть на распределение  $r + \gamma Z_{\theta^-}(s, a)$  для жадного  $a'$  и спроецировать его (случ. величина справа). Утверждается эквивалентность этих процедур: мы можем проецировать лишь компоненты  $[\mathfrak{B}_D^* Z_{\theta^-}](s, a)$ . Таким образом, сэмплы из  $\text{Py}(\mathbb{T})$  при случайных  $s' \sim p(s' \mid s, a)$  есть сэмплы  $[\Pi \mathfrak{B}_D^* Z_{\theta^-}](s, a)$ .

Однако.

Теорема: Градиенты расстояния Вассерштайна до сэмплов не являются несмещёнными оценками градиента расстояния Вассерштайна до полного распределения.

Таким образом, псевдометрикой, которую можно оптимизировать по сэмплам, является наша любимая KL-дивергенция. Мы понимаем, что, с одной стороны, теория подсказывает нам, что в пространстве  $Z$ -функций KL-дивергенция потенциально не приближает нас к истинной оптимальной  $Z$ -функции, но зато мы сможем оптимизировать её в model-free режиме

Итак, рассмотрим в качестве  $\mathcal{D}$  KL-дивергенцию (значит, будет важен порядок аргументов). Для неё вылезает ещё одна проблема: домен сравниваемых распределений должен совпадать, иначе KL-дивергенция по определению бесконечность и не оптимизируется. К счастью, мы уже решили, что мы будем в качестве целевого распределения использовать  $\Pi y(\mathbb{T})$ , которое имеет тот же домен — сетку  $z_0 < z_1 < \dots < z_A$ .

Теорема: Градиент KL-дивергенции до целевой переменной  $\Pi y(\mathbb{T})$  есть несмещённая оценка градиента.

$$\nabla_{\theta} \text{KL}([\Pi \mathfrak{B}_D^* \mathcal{Z}_{\theta-}](s, a) \parallel \mathcal{Z}_{\theta}(s, a)) = \mathbb{E}_{s'} \nabla_{\theta} \text{KL}(\Pi y(\mathbb{T}) \parallel \mathcal{Z}_{\theta}(s, a))$$

Итак, градиент KL-дивергенции — мат.ожидание по целевому распределению, и значит, мы можем вместо мат.ожидания по  $[\Pi \mathfrak{B}_D^* \mathcal{Z}_{\theta-}](s, a)$  использовать Монте-Карло оценку по сэмплам. При этом поскольку у нас есть даже не просто сэмплы, а целая компонента  $\Pi y(\mathbb{T})$  целевого распределения, то по ней интеграл мы можем взять просто целиком (он состоит всего из  $A$  слагаемых, как видно, поскольку  $\Pi y(\mathbb{T}) \in \mathcal{C}$ )

Получается следующее: для данного перехода мы в качестве функции потерь возьмём  $\text{KL}(\Pi y(\mathbb{T}) \parallel \mathcal{Z}_{\theta}(s, a))$ , где таргет  $y(\mathbb{T})$ . Раз мы используем категориальную аппроксимацию, и  $\Pi y(\mathbb{T})$  — категориальное распределение на той же сетке, то эта KL-дивергенция считается явно и (с точностью до константы, не зависящей от  $\theta$ ) равна

$$-\sum_{t=0}^A P(\Pi y(\mathbb{T}) = z_t) \log p_t((s_t, a_t, \theta))$$

Как видно из этой формулы, мы по сути начинаем решать задачу классификации, где у нас есть для данного входа  $s, a$  сразу целая компонента «целевого» распределения. Минимизация KL-дивергенции, хоть и является стандартной функцией потерь в таких ситуациях, сейчас имеет для нас побочный эффект: мы отчасти потеряли «физический смысл» наших «классов». KL-дивергенция смотрит на каждый узел  $z_i$  нашей сетки отдельно и сравнивает вероятность, которую мы выдаём сейчас, с вероятностью  $z_i$  в таргете. Она не учитывает, находится ли разница в вероятностной массе на соседнем узле, например,  $z_{i+1}$  (в «соседнем» исходе) или на противоположном конце сетки в условном  $z_0$ ; в обоих случаях KL-дивергенция будет выдавать одно и то же значение. Адекватные метрики в пространстве распределений, например, Вассерштайн, продифференцировали бы эти случаи. Причём заметим, что мы, вообще говоря, могли бы посчитать того же Вассерштайна между  $y(\mathbb{T})$  и  $\mathcal{Z}_{\theta}(s, a)$ , но градиенты такой функции потерь не были бы несмещёнными

оценками градиента для минимизации и такой алгоритм был бы некорректен.

Тем не менее, мы получили первый полноценный Distributional (Распределительный) алгоритм. Соберём c51, он же Categorical DQN, целиком.

### Алгоритм: Categorical DQN (c51)

**Гиперпараметры:**  $B$  — размер мини-батчей,  $V_{\max}, V_{\min}$ ,  $A$  — параметры категориальной аппроксимации,  $K$  — периодичность обновления таргет-сети,  $\varepsilon(t)$  — стратегия исследования,  $p_i(s, a, \theta)$  — нейросетка с параметрами  $\theta$ , SGD-оптимизатор

Предпочитать узлы сетки  $z_i := V_{\min} + \frac{i}{A}(V_{\max} - V_{\min})$

Инициализировать  $\theta$  произвольно

Положить  $\theta^- := \theta$

Пронаблюдать  $s_0$

На очередном шаге  $t$ :

1. выбрать  $a_t$  случайно с вероятностью  $\varepsilon(t)$ , иначе  $a_t := \operatorname{argmax}_{a_t} \sum_{i=0}^A z_i p_i(s_t, a_t, \theta)$
2. пронаблюдать  $r_t, s_{t+1}, \text{done}_{t+1}$
3. добавить пятёрку  $(s_t, a_t, r_t, s_{t+1}, \text{done}_{t+1})$  в реплей буфер

4. засэмплировать мини-батч размера  $B$  из буфера

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

$$\mathbf{P}(y(\mathbb{T}) = r + \gamma(1 - \text{done})z_i) := p_i\left(s', \operatorname{argmax}_{a'} \sum_{i=0}^A z_i p_i(s', a', \theta^-), \theta^-\right)$$

6. спроецировать таргет на сетку  $\{z_0, z_1 \dots z_A\}$ :  $y(\mathbb{T}) \leftarrow \Pi y(\mathbb{T})$

7. посчитать лосс:

$$\text{Loss}(\theta) := -\frac{1}{B} \sum_{\mathbb{T}} \sum_{i=0}^A \mathbf{P}(y(\mathbb{T}) = z_i) \log p_i(s_t, a_t, \theta)$$

8. сделать шаг градиентного спуска по  $\theta$ , используя  $\nabla_{\theta} \text{Loss}(\theta)$

9. если  $t \bmod K = 0$ :  $\theta^- \leftarrow \theta$

### *Квантильная аппроксимация Z-функций*

В c51 мы воспользовались тем, что KL-дивергенция — это мат.ожидание по одному из сравниваемых распределений. Только это позволило нам несмещённо оценивать градиенты, используя лишь один сэмпл  $s'$ . Их нельзя так просто «оптимизировать по сэмплам» — и к тому же у нас есть сложности с доменом распределения, нам необходим оператор проекции и аккуратный подбор неудобных гиперпараметров  $V_{\max}, V_{\min}$ , которые критично подобрать более-менее правильно.

Оказывается, задача будет решена, если мы выберем другую аппроксимацию распределений в  $P(\mathbb{R})$ . Если раньше мы зафиксировали домен (узлы сетки) и подбирали вероятности, то теперь мы зафиксируем вероятности и будем подбирать узлы сетки. На первый взгляд это может показаться странно (как можно отказываться от предсказания вероятностей?), однако на самом деле это весьма гибкое семейство распределений с интересными свойствами. И так:

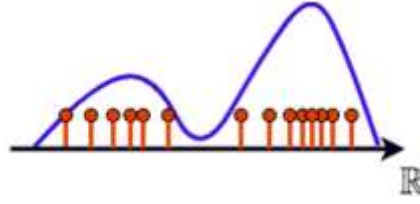


Рис. 6.8.

Определение: Обозначим семейство квантильных распределений  $Q \subset P(\mathbb{R})$  с  $A$  атомами как множество равномерных дискретных распределений с  $A$  произвольными исходами: если  $Z(s, a) \in Q$ , то для некоторых  $A$  чисел  $z_0, z_1 \dots z_{A-1}$ :

$$P(Z(s, a) = z_i) = \frac{1}{A}$$

Сразу хорошо то, что нам понадобится всего один гиперпараметр — число атомов  $A$  — и не понадобится подбирать верхнюю-нижнюю границу ручками. Также заметим, что вырожденное распределение принадлежит  $Q$ : просто все  $z_i$  в этом случае совпадают.

$\mathfrak{B}_D^* Z_{\theta^-}$  — тем не менее, может снова выпадать из такого семейства представлений, и нам всё равно понадобится какая-то проекция. Но на этот раз мы сможем сделать куда более естественную проекцию. На очередном шаге для заданной пары  $s, a$  мы будем оптимизировать расстояние Вассерштайна  $\mathcal{W}_1$  между правой частью уравнения Беллмана и тем, что мы выдаём:

$$\mathcal{W}_1([\mathfrak{B}_D^* Z_{\theta^-}](s, a) \parallel \mathcal{Z}) \rightarrow \min_{\mathcal{Z} \in Q}$$

Поскольку мы ограничены семейством квантильных распределений, то лучшее, что мы можем сделать, это спроецировать шаг метода простой итерации в него.

Введём следующее обозначение («середины отрезков сетки»):

$$\tau_i := \frac{\frac{i}{A} + \frac{i+1}{A}}{2}$$

Теорема: Пусть  $F$  — функция распределения  $[\mathfrak{B}_D^* Z_{\theta^-}](s, a)$ . Тогда решение  $Z \in Q$  задачи имеет домен  $z_0, z_1 \dots z_{A-1}$ :

$$z_i = F^{-1}(\tau_i)$$

Quantile Regression DQN (Квантильная регрессия)

Мы получили, что нам достаточно уметь искать лишь  $A$  определённых квантилей распределения  $[\mathfrak{B}_D^* Z_{\theta^-}](s, a)$  для вычисления аппроксимации правой части уравнения Беллмана. Можем ли мы это сделать, используя только сэмплы? Конечно.

Квантильная регрессия (quantile regression) — способ получить  $\tau$ -ый

квантиль некоторого распределения, из которого доступна только лишь выборка. В частном случае, мы получим известный факт о том, что для получения медианы ( $\frac{1}{2}$ -го квантиля) нужно минимизировать MAE между константным прогнозом и сэмплами из распределения.

Определение: Для заданного  $\tau \in (0,1)$  квантильной функцией потерь (quantile loss) называется:

$$Loss_{\tau}(c, X) = \begin{cases} \tau(c - X) & c \geq X \\ (1 - \tau)(X - c) & c < X \end{cases}$$

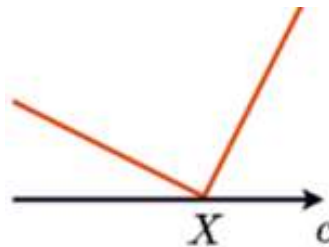


Рис. 6.9.

Теорема — Квантильная регрессия: Решением задачи

$$\mathbb{E}_X Loss_{\tau}(c, X) \rightarrow \min_{c \in \mathbb{R}}$$

будет  $\tau$ -ый квантиль распределения случайной величины  $X$ .

Итак, соберём всё вместе. У нас есть нейросеть  $z_i(s, a, \theta)$  с параметрами  $\theta$ , которая для данного состояния-действия производит  $A$  произвольных вещественных чисел, которые мы интерпретируем как  $A$  равновероятных возможных исходов случайной величины  $Z_{\theta}(s, a)$ .

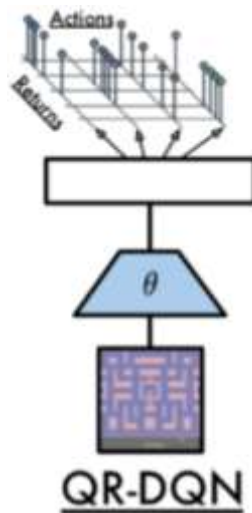


Рис. 6.10.

Обозначим за  $\theta^-$  веса таргет-сети, как обычно. Для очередного перехода  $T: = (s, a, r, s')$  из буфера мы хотим провести оптимизацию

$$\mathcal{W}_1([\mathfrak{B}_D^* \mathcal{Z}_{\theta^-}](s, a) \parallel \mathcal{Z}_{\theta}(s, a)) \rightarrow \min_{\theta}$$

и мы поняли, что это эквивалентно поиску квантилей распределения  $[\mathfrak{B}_D^* Z_{\theta^-}](s, a)$ , поэтому для оптимизации  $i$ -го выхода нейросетки будем оптимизировать квантильный лосс (по  $i$  просто просуммируем — хотим учить все  $A$  интересующих нас квантилей):

$$\sum_{i=0}^{A-1} \mathbb{E}_{x \sim [\mathfrak{B}_D^* Z_{\theta^-}](s, a)} \text{Loss}_{\tau_i}(z_i(s, a, \theta), x) \rightarrow \min_{\theta}$$

Опять же заметим, что  $\mathbb{E}_{x \sim [\mathfrak{B}_D^* Z_{\theta^-}](s, a)}$  распадается в сэмплирование  $s'$  и интегрирование по возможным исходам  $Z_{\theta^-}(s', \pi^*(s'))$ , где  $\pi^*(s')$  выбирает действие жадно. Мат.ожидание по  $Z_{\theta^-}(s', a')$  при данных  $s', a'$  есть просто усреднение по  $A$  равновероятным исходам, поэтому его мы посчитаем явно. Итого:

$$\underbrace{\sum_{i=0}^{A-1}}_{\text{учим } A \text{ квантилей}} \underbrace{\mathbb{E}_{s'}}_{\substack{\text{вероятности} \\ \text{сэмплов}}} \underbrace{\frac{1}{A}}_{\text{сэмпл}} \sum_{j=0}^{A-1} \text{Loss}_{\tau_i}(\underbrace{z_i(s, a, \theta)}_{\text{прогноз}}, \underbrace{r + \gamma z_j(s', a', \theta^-)}_{\text{сэмпл}}) \rightarrow \min_{\theta}$$

Занося внешнюю сумму под мат.ожидание по  $s'$ , получаем функцию потерь, градиент которой можно оценивать по Монте-Карло, используя лишь сэмплы  $s'$  из функции переходов.

### Алгоритм : Quantile Regression DQN (QR-DQN - Квантильная регрессия)

**Гиперпараметры:**  $B$  — размер мини-батчей,  $A$  — число атомов,  $K$  — периодичность обновления таргет-сети,  $\varepsilon(t)$  — стратегия исследования,  $z_i(s, a, \theta)$  — нейросетка с параметрами  $\theta$ , SGD-оптимизатор

Предусчитать середины отрезков квантильной сетки  $\tau_i := \frac{i + \frac{i+1}{A}}{2}$

Инициализировать  $\theta$  произвольно

Положить  $\theta^- := \theta$

Пронаблюдать  $s_0$

**На очередном шаге  $t$ :**

1. выбрать  $a_t$  случайно с вероятностью  $\varepsilon(t)$ , иначе  $a_t := \underset{a_t}{\operatorname{argmax}} \sum_{i=0}^{A-1} z_i(s_t, a_t, \theta)$
2. пронаблюдать  $r_t, s_{t+1}, \text{done}_{t+1}$

3. добавить пятёрку  $(s_t, a_t, r_t, s_{t+1}, \text{done}_{t+1})$  в реплей буфер

4. засэмплировать мини-батч размера  $B$  из буфера

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

$$y(\mathbb{T})_j := r + (1 - \text{done}) \gamma z_j \left( s', \underset{a'}{\operatorname{argmax}} \sum_i z_i(s', a', \theta^-), \theta^- \right)$$

6. посчитать лосс:

$$\text{Loss}(\theta) := \frac{1}{BA} \sum_{\mathbb{T}} \sum_{i=0}^{A-1} \sum_{j=0}^{A-1} (\tau_i - \mathbb{I}[z_i(s, a, \theta) < y(\mathbb{T})_j]) (z_i(s, a, \theta) - y(\mathbb{T})_j)$$

7. сделать шаг градиентного спуска по  $\theta$ , используя  $\nabla_{\theta} \text{Loss}(\theta)$

8. если  $t \bmod K = 0$ :  $\theta^- \leftarrow \theta$

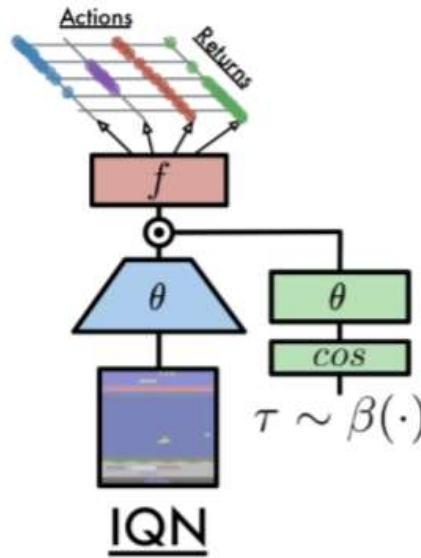
### *Implicit Quantile Networks (Неявные квантильные сети)*

BQR-DQN мы фиксировали «равномерную сетку» на оси квантилей: говорили, что наше аппроксимирующее распределение есть равномерное на домене из  $A$  атомов. Идея: давайте будем уметь в нашей нейросети выдавать произвольные квантили, каким-то образом задавая  $\tau \in (0,1)$  дополнительно на вход. Тогда наша модель  $z(s, a, \tau, \theta)$  будет неявно (implicit) задавать, вообще говоря, произвольное распределение на  $\mathbb{R}$ . По сути, мы моделируем квантильную функцию «целиком»; очень удобно:

$$F_{Z_{\theta}(s,a)}^{-1}(\tau) := z(s, a, \tau, \theta)$$

Поймём, как тогда считать мат.ожидание (или Q-функцию) в такой модели.

Теорема: Пусть  $F$  — функция распределения случайной величины  $X$ . Тогда, если  $\tau \sim U[0,1]$ , случайная величина  $F^{-1}(\tau)$  имеет то же распределение, что и  $X$ .



Итак, мы можем аппроксимировать жадную стратегию примерно так:

$$\pi^*(s) := \arg \max_a \sum_{i=0}^N z(s, a, \tau_i, \theta), \tau_i \sim U[0,1]$$

В качестве функции потерь предлагается использовать тот же квантильный лосс, что и в QR-DQN, только если в QR-DQN нам были нужны определённые  $A$  квантилей, то теперь предлагается засэмплировать  $N'$  каких-то квантилей и посчитать лосс для них. Для подсчёта лосса нам было нужно брать мат.ожидание по  $Z_{\theta}(s', a')$ , для чего в формуле мы пользовались тем, что это распределение в нашей модели равномерно. Теперь же этот интеграл мы заменяем на МонтеКарло оценку с произвольным числом сэмплов  $N''$ , а для сэмплирования

опять же используем рассмотренную теорему:

$$Loss(\mathbb{T}, \theta) := \sum_{i=0}^{N'} \frac{1}{N''} \sum_{j=0}^{N''} Loss_{\tau_i}(z(s, a, \tau_i, \theta), r + \gamma z(s', \pi^*(s'), \tau_j, \theta^-))$$

где  $\tau_i, \tau_j \sim U[0,1]$ .

*Rainbow DQN (Падыга)*

И так были рассмотрены весьма разные улучшения DQN, нацеленные на решения очень разных проблем. Хорошо видно, что все эти модификации «ортогональны» и могут включаться выключаться, так сказать, независимо в алгоритм. Distributional (Распределительный)-подход, вообще говоря, не решает какую-то проблему внутри DQN, но может рассматриваться как ещё одна модификация базового алгоритма DQN.

Rainbow DQN совмещает 6 модификаций алгоритма DQN:

- Double DQN (Двойной)
- Dueling DQN (Дуэльные)
- Noisy DQN (Шумные)
- Prioritized Experience Replay (Повтор приоритетного опыта)
- Multi-step DQN (Многошаговый)
- Distributional RL (Распределительный)

Понятно, что можно использовать любой алгоритм.

Алгоритм: Rainbow DQN

**Гиперпараметры:**  $B$  — размер мини-батчей,  $V_{\max}, V_{\min}, A$  — параметры категориальной аппроксимации,  $K$  — периодичность обновления таргет-сети,  $N$  — количество шагов в оценке,  $\alpha$  — степень приоритизации,  $\beta(t)$  — параметр importance sampling коррекции для приоритизированного реплея,  $p_i(s, a, \theta, \varepsilon)$  — нейросетка с параметрами  $\theta$ , SGD-оптимизатор

Предпочитать узлы сетки  $z_i := V_{\min} + \frac{i}{A}(V_{\max} - V_{\min})$

Инициализировать  $\theta$  произвольно

Положить  $\theta^- := \theta$

Пронаблюдать  $s_0$

**На очередном шаге  $t$ :**

1. выбрать  $a_t := \operatorname{argmax}_{a_t} \sum_{i=0}^A z_i p_i(s_t, a_t, \theta, \varepsilon)$ ,  $\varepsilon \sim \mathcal{N}(0, I)$
2. пронаблюдать  $r_t, s_{t+1}, \text{done}_{t+1}$
3. построить  $N$ -шаговый переход  $\mathbb{T} := (s, a, \sum_{n=0}^N \gamma^n r^{(n)}, s^{(N)}, \text{done})$ , используя последние  $N$  наблюдений, и добавить его в реплей буфер с максимальным приоритетом
4. засэмплировать мини-батч размера  $B$  из буфера, используя вероятности  $P(\mathbb{T}) \propto \rho(\mathbb{T})^\alpha$
5. посчитать веса для каждого перехода:

$$w(\mathbb{T}) := \frac{1}{\rho(\mathbb{T})^{\beta(t)}}$$

6. для каждого перехода  $\mathbb{T} := (s, a, \bar{r}, \bar{s}, \text{done})$  посчитать таргет:

$$\varepsilon_1, \varepsilon_2 \sim \mathcal{N}(0, I)$$

$$\mathbf{P}(y(\mathbb{T}) = \bar{r} + \gamma^N(1 - \text{done})z_i) := p_i\left(\bar{s}, \underset{\bar{a}}{\operatorname{argmax}} \sum_{i=0}^A z_i p_i(\bar{s}, \bar{a}, \theta, \varepsilon_1), \theta^-, \varepsilon_2\right)$$

7. спроецировать таргет на сетку  $\{z_0, z_1 \dots z_A\}$ :  $y(\mathbb{T}) \leftarrow \Pi y(\mathbb{T})$

8. посчитать для каждого перехода лосс:

$$L(\mathbb{T}, \theta) := - \sum_{i=0}^A \mathbf{P}(y(\mathbb{T}) = z_i) \log p_i(s_t, a_t, \theta, \varepsilon) \quad \varepsilon \sim \mathcal{N}(0, I)$$

9. обновить приоритеты всех переходов из буфера:  $\rho(\mathbb{T}) \leftarrow L(\mathbb{T}, \theta)$

10. посчитать суммарный лосс:

$$\text{Loss}(\theta) := \frac{1}{B} \sum_{\mathbb{T}} w(\mathbb{T}) L(\mathbb{T}, \theta)$$

11. сделать шаг градиентного спуска по  $\theta$ , используя  $\nabla_{\theta} \text{Loss}(\theta)$

12. если  $t \bmod K = 0$ :  $\theta^- \leftarrow \theta$