# Оглавление

[Оглавление 2](#_Toc149269341)

[Глава 1. Теория 3](#_Toc149269342)

[**Общие понятия обучения с подкреплением 3**](#_Toc149269343)

[**Алгоритмы обучения с подкреплением (общий случай) 4**](#_Toc149269344)

[**Задача о многоруком бандите. Алгоритмы решения 5**](#_Toc149269345)

[**Адаптивные стратегии в задаче о многоруком бандите 8**](#_Toc149269346)

[**Выводы 9**](#_Toc149269347)

# Глава 1. Теория

## Общие понятия обучения с подкреплением

Обучение с подкреплением (Reinforcement Learning, RL) — это область машинного обучения, где агент обучается взаимодействовать со средой для максимизации некой награды. В отличие от обучения с учителем, в RL у агента нет явных примеров правильного поведения или сведений об окружающей среде, и он определяет оптимальную стратегию взаимодействия со средой на основе полученных наград. Такой метод обучения наиболее близок к человеческому, в котором не приходится заранее собирать и анализировать большое количество информации (обучающую выборку) для построения прогнозов. Вместо этого человек выбирает как ему лучше поступать, основываясь на собственном ранее полученном опыте.

В последние годы обучение с подкреплением сыскало большую популярность в виду универсальности своего математического аппарата. Его методы нашли применение в задача следующих прикладных задачах:

* Рекомендация новостных статей пользователям;
* Показ рекламы в Интернете;
* Управление технологическими процессами;
* Управление роботами;
* Управление ценами и ассортиментом в сетях продаж;
* Игра на бирже;
* Маршрутизация в телекоммуникационных сетях;
* Маршрутизация в беспроводных сенсорных сетях;
* Стратегические игры: шахматы, го, Dota2, StarCraft2…

Основными компонентами обучения с подкреплением являются:

* Агент: сущность, принимающая решения.
* Среда: всё, что окружает агента и на что он может воздействовать.
* Действие: взаимодействие агента со средой.
* Состояние: описание среды в данный момент времени.
* Награда: сигнал, который агент получает от среды после выполнения действия в определенном состоянии.
* Стратегия (политика): правило выбора действия агентом, приводящее к максимизации премии.
* Раунд: интервал, в которой происходит взаимодействие агента со средой.

## Алгоритмы обучения с подкреплением (общий случай)

Алгоритмы обучения с подкреплением охватывают широкий спектр методов, которые агенты используют для обучения оптимальной стратегии взаимодействия с окружающей средой. Основная цель агента — максимизировать кумулятивную награду в течение времени.

Эти алгоритмы можно классифицировать в зависимости от того, используют ли они модель среды в процессе обучения:

1. **Основанные на модели**

В этих алгоритмах агент пытается узнать или аппроксимировать модель среды.

* Динамическое программирование (Dynamic Programming): это классический метод, который использует знание модели для нахождения оптимальной политики. Примеры включают в себя итерацию по политике (Policy Iteration) и итерацию по функции ценности (Value Iteration).
* Планирование (Planning): агент использует свою модель среды для имитации возможных будущих сценариев и принятия решений на основе этих сценариев. Например, Monte Carlo Tree Search (MCTS) использует случайные сэмплы для исследования пространства действий.
* Адаптивное планирование: при изменении среды агент может адаптировать свою модель и соответственно изменять свою политику. Примером такого алгоритма является метод временных разностей (Temporal-Difference, TD).

1. **Без модели**

Агент не пытается явно узнать модель среды, но вместо этого напрямую оценивает ценность состояний или действий.

* Q-learning: оффлайн метод, который оценивает функцию ценности действия, не зависящую от текущей политики агента.
* SARSA: оценивает функцию ценности действия, основываясь на текущем и следующем действиях.
* Policy Gradient Methods: вместо оценки функции ценности агент напрямую оптимизирует свою политику с использованием градиентного восхождения.
* Actor-Critic Methods: комбинируют идеи из методов, основанных на политике, и методов оценки ценности, имея две компоненты: актер, выбирающий действия, и критик, оценивающий действия.

**Вывод**

Выбор между алгоритмами, основанными на модели, и без модели зависит от специфики задачи, доступности данных, вычислительных ресурсов и других факторов. В то время как методы, основанные на модели, могут быть более эффективными в терминах количества требуемых взаимодействий со средой, поскольку они могут "думать вперед", методы без модели часто проще в реализации и могут быть более устойчивыми в сложных или непредсказуемых средах.

## Задача о многоруком бандите. Алгоритмы решения

До текущего момента обучение с подкреплением рассматривалось как задача о среде с состояниями. То есть среда могла меняться и адаптироваться, реагируя на действия агента, что оказывало влияние на дальнейший выбор действий. Однако существует частный случай обучения с подкреплением, в котором среда является стационарной. Такая задача получила название задачи о многоруком бандите (Multi-Armed Bandit).

**Постановка задачи:**

Пусть – множество всех возможных действий агента.

– неизвестное распределение премии для действия .

– стратегия агента в момент для действия, распределенная на .

Тогда игра агента со средой производится следующим образом:

1. Инициализируется стратегия ;
2. Для всех раундов ;

2.1) Агент выбирает действие согласно стратегии: ;

2.2) Среда в ответ на действие генерирует премию: ;

2.3) В зависимости от премии агент корректирует стратегию: .

Таким образом, после раундов для каждого из действий агентом может быть накоплена некоторая статистика, обозначающая среднюю премию, получаемую за действие в раундах. Метод подсчета этой статистики приведен в формуле 1.

Формула 1. Средняя премия за действие в раундах.

Если устремить число раундов к бесконечности, получится узнать среднюю оценку премии для каждого действия. Тогда для максимизации выигрыша останется лишь выбрать действие с максимальной средней премией и все время использовать его. Однако провести бесконечное число раундов невозможно, из-за чего вводится абстрактное понятие ценности действия , описанное формулой 2.

Формула 2. Ценность действия a.

С учетом введенных понятий, задача о многоруком бандите сводится к быстрому предсказыванию величины для всех действий и выбора максимального значения.

То есть требуется сыграть минимальное количество раундов , достаточное для построения оценки средней премии для каждого действия , с помощью которой удастся с как можно большей вероятностью “угадать” ценность каждого из действий и выбрать из них самое ценное.

**Выбор стратегии.**

Ключевым шагом в обучении с подкреплением является шаг 2.3) игры агента со средой. Интуитивно понятным методом выбора следующего действия является жадная стратегия (Greedy Policy). В ней предпочтения отдается действиям, уже имеющим максимальную среднюю оценку премии, а другие действия удаляются из рассмотрения.

Обновление множества действий производится по правилу, описанному в формуле 3.

Формула 3*.* Множество действий с максимальной текущей оценкой ценности.

Тогда выбор действия в момент времени может осуществляться по формуле 4.

Формула 4. Жадная стратегия. Выбор любого действия .

Однако жадная стратегия содержит существенный недостаток: из рассмотрения при первой же возможности убираются действия, не давшие статистически значимых результатов на данный момент, но которые в последствии в среднем могли показать себя лучше оставленных действий. Очевидно, что для полноты обучения на каждом шаге следует оставлять вероятность выбора действия, до текущего момента не принесшего максимальной выгоды.

Для этого был введен принцип изучения–применения (Exploration–Exploitation), который заключается переходе от равномерного выбора действия к жадному.

Простейшую реализацию этого принципа можно встретить в алгоритме эпсилон-жадной стратегии (Epsilon-Greedy Policy), приведенном в формуле 5.

Формула 5.Эпсилон-жадная стратегия.

Здесь при больших стратегия будет стремиться к равномерной (изучающей), а при малых, наоборот, к жадной (применяющей). Это позволит набрать необходимую статистику в начале обучения, а затем перейти к жадной стратегии, уменьшив параметр со временем.

Метод изучения-применения является ключевым не только в задаче о многоруком бандите, но и в задаче обучения с подкреплением в общем. Все описанные выше алгоритмы используют этот принцип в том или ином виде.

Останавливаясь подробнее на равномерно-жадных стратегиях для задачи о многоруком бандите, необходимо отметить идею, состоящую в том, что чем больше средняя оценка действия , тем больше должна быть вероятность следующего выбора этого действия (с поправкой на возможность исследования других действий). Таким образом появляется необходимость сопоставления множеству ценности действий множества вероятностей выбора.

SoftMax Policy (Распределение Гиббса) – более мягкий, нежели -жадная стратегия, вариант компромисса “изучение-применение”, включающее в себя вышеописанную идею: чем больше , тем больше вероятность выбора действия . Реализация приведена в формуле 6.

Формула 6*.* СтратегияSoftMax.

Параметром, отвечающим за баланс между изучением и применением, в данном случае, является параметр τ – температура, которая при стремлении к 0 делает стратегию более жадной, а при стремлении к бесконечности – равномерной, то есть чисто исследовательской. Для достижения наилучших результатов τ, как и в -жадной стратегии, можно уменьшать со временем.

Очевидно, что, пользуясь таким принципом, можно придумать большое количество стратегий, однако жадность алгоритма всегда будет позволять производить обучение быстрее. Хотелось бы иметь оптимальную жадную стратегию, умеющую адаптироваться от изучения к применению с минимальными потерями.

Возвращаясь к жадной стратегии, нельзя не отметить метод UCB, одну из самых популярных стратегий выбора следующего действия.

UCB Method (Upper Confidence Bound) – метод выбора следующего действия, использующий верхнюю оценку ценности действия, которая, в свою очередь, зависит как от размера средней премии , так и от частоты использования этого действия. Чтобы набирать статистику для каждого действия, считается, что чем чаще выбирается действие, тем с меньшей вероятностью оно должно быть выбрано снова. Реализация метода приведена в формуле 7.

Формула 7*.* Метод UCB выбора действий с максимальной верней оценкой ценности.

Параметр , стоящий в знаменателе отвечает за частоту выбора действия . Параметр , как и упомянутые ранее τ и , является обязательным условием сохранение компромисса изучения-применения и так же может со временем уменьшаться от большего к меньшему, приводя стратегию от исследовательской к жадной.

Очевидно, что коррекция стратегии будет осуществлена согласно формуле 4.

## Адаптивные стратегии в задаче о многоруком бандите

Главной проблемой всех описанных выше алгоритмов является долгое время пересчета функции средней ценности действия, которая по своей сути является средним арифметическим. Это ведет к квадратичному возрастанию времени обучения, что особенно плохо для задач с большим числом возможных действий.

Исправить не достаток помогает экспоненциальное скользящее среднее, дающее, например, общую рекуррентную формулу вычисления для корректировки стратегии (формула 8).

Формула 8. Рекуррентное обновление средней ценности

действия в задаче о многоруком бандите.

Экспоненциальное скользящее среднее может найти применение во многих вариациях задачи о многоруких бандитах. Например, если результаты SoftMax не показывают должного роста обучения, но задача подразумевает мягкий компромисс изучения-применения, можно попробовать “сгладить” подаваемые стратегии величины как разности средней премии с текущей. Этот метод получил название сравнения с подкреплением (Reinforcement Comparison). В результате обучения наибольшее предпочтение отдается действию, получающему в среднем максимальную премию. Реализация такого подхода приведена в формуле 9.

средняя премия по **всем** действиям;

предпочтения действий;

SoftMax-стратегия агента.

Формула 9. Метод сравнения с подкреплением.

Однако применение экспоненциального скользящего среднего имеет еще одно важное свойство. С его помощью можно усреднять не только числовой временной ряд, но и временной ряд, состоящий из дискретных распределений. Таким образом можно оптимизировать жадную стратегию (формула 4), которая, например, применяется при использовании метода UCB. Такая реализация получила название метод преследования (Pursuit). Его реализация приведена в формуле 10.

Формула 10. Метод преследования жадной стратегии.

## Выводы

В этой главе были приведены общие сведения о таком методе машинного обучения как обучение с подкреплением. Была предложена классификация задач обучения с подкреплением на общие (имеющие динамические среды) и на задачи о многоруком бандите (имеющие статичные среды). В классе задач с динамическими средами были выделены самые известные алгоритмы стратегий, которые были сгруппированы по принципу необходимости модели среды в их реализации. Было дано подробное описание самым популярным эпсилон-жадным стратегиям агента в задаче о многоруком бандите, а также его адаптивным стратегиям.

К сожалению, к задаче обучения с подкреплением нет универсального подхода, и все приведенные алгоритмы невозможно сравнивать без условий конкретной задачи и опытов, проделанных в испытательной среде. Это связано с серьезными различиями в строении алгоритмов стратегий, которые подразумевают сведения о возможных действиях агента и о состояниях среды. Однако можно выделить принцип, по которым можно отбирать необходимые стратегии и классифицировать задачу обучения с подкреплением.

Если среда является динамической, то есть распределение премий для каждого действия агента может меняться, в зависимости от предшествующих действий, то следует определить, как программе будут подаваться сведения о среде. Если можно легко составить модель среды, отвечающую каждому действию агента, то оптимальными методами будут динамическое программирование, планирование и адаптивное планирование. Когда среду тяжело предсказать, следует использовать алгоритмы, не использующие модель, такие как Q-learning, SARSA, Policy Gradient Method, Actor-Critic Methods и так далее.

Если среда статична, то следует воспринимать задачу как задачу о многоруком бандите. В зависимости от числа действий, распределения премий и продолжительности раундов следует выбирать между жадными, например UCB, и мягкими, например SoftMax, стратегиями. При большом числе действий первый вариант предпочтительней, так как отсекает больше ненужных действий. Так же во внимание следует принять адаптивные стратегии, использующие экспоненциальное скользящее среднее для сглаживания результатов обработки, хотя такие стратегии редко конкурируют с эпсилон-жадными, так как имеют совершенно другой подход и используются в задачах, где можно легко подобрать коэффициенты сглаживания (в задачах, где хорошо известны вероятностные распределения премий).

В виду того, что сложность алгоритмов обучения в динамических средах зачастую на порядок выше, чем в задачах о многоруком бандите, всегда следует оценивать возможность сведения среды к статичной.