Лекция 12

Обучение с подкреплением

1. Введение в обучение с подкреплением

Обучение с подкреплением (ОП) — это метод машинного обучения, в котором агенты обучаются взаимодействовать с окружающей средой, чтобы достичь определенной цели, получая награды или наказания за свои действия. В отличие от обучения с учителем, где алгоритм обучается на основе размеченных данных, обучение с подкреплением основывается на принципе проб и ошибок. Агенты постепенно накапливают знания о том, какие действия ведут к успешным результатам, улучшая свои стратегии на основе опыта.

Обучение с подкреплением находит применение в различных областях, таких как робототехника, игры, финансы и здравоохранение, где агентам необходимо адаптироваться к динамическим условиям среды. Примером является игра в шахматы, где агент может научиться стратегически мыслить и планировать свои действия на несколько шагов вперед для достижения победы.

2. Основные понятия обучения с подкреплением

Для понимания обучения с подкреплением важно изучить основные концепции, которые лежат в его основе:

- **Агент**: Система, которая принимает решения на основе взаимодействия со средой.
- **Среда (Environment)**: Все, с чем агент взаимодействует и что может влиять на результаты его действий.
- **Состояние (State)**: Набор параметров, описывающих текущее положение агента в среде.
- **Действие (Action)**: Выбор, который агент может сделать в текущем состоянии.
- **Награда (Reward)**: Обратная связь от среды, указывающая на успех или неудачу действий агента.
- **Политика (Policy)**: Стратегия, которой следует агент при выборе действий в каждом состоянии.
- **Функция ценности (Value Function)**: Оценка того, насколько хорошее данное состояние для агента с учетом возможных будущих наград.
- **Функция полезности (Utility Function)**: Ожидаемая совокупная награда от текущего состояния.

Эти понятия описывают основные аспекты взаимодействия агента со средой, что позволяет формализовать задачи обучения с подкреплением.

3. Математическая формализация: Процесс принятия решений в Markov Decision Process (MDP)

Процессы принятия решений в ОП формализуются с помощью модели Марковского процесса принятия решений (Markov Decision Process, MDP). MDP представляет собой пятерку (S,A,P,R,γ)(S, A, P, R, \gamma)(S,A,P,R,γ), где:

- SSS — множество возможных состояний среды.
- AAA — множество действий, которые может совершать агент.
- P(s′∣s,a)P(s'|s, a)P(s′∣s,a) — функция перехода, определяющая вероятность перехода в состояние s′s's′ при выполнении действия aaa из состояния sss.
- R(s,a)R(s, a)R(s,a) — функция награды, которая возвращает значение награды при выполнении действия aaa в состоянии sss.
- γ\gammaγ — коэффициент дисконтирования, определяющий, насколько агент ценит будущие награды по сравнению с текущими.

Цель агента в MDP — максимизировать совокупную ожидаемую награду на протяжении всего взаимодействия со средой. Это достигается путем разработки оптимальной политики π\piπ, которая указывает, какое действие следует выбрать в каждом состоянии.

4. Основные методы обучения с подкреплением

Существует два основных подхода к обучению с подкреплением: метод ценности (Value-based methods) и метод политики (Policy-based methods). Эти подходы различаются по способу, которым агент изучает оптимальную стратегию.

4.1 Методы ценности (Value-based methods)

Методы ценности — это ключевой компонент методов обучения с подкреплением, где основной задачей агента является вычисление функции ценности для каждого состояния или комбинации состояния и действия. Функция ценности определяет, насколько выгодным является текущее состояние (или действие в этом состоянии) для достижения максимального вознаграждения в долгосрочной перспективе. Цель агента — максимизировать совокупную ценность, т.е. ожидаемое суммарное вознаграждение, при переходе от текущего состояния к конечному, следуя некоторой политике.

Важным элементом обучения агента является нахождение оптимальной политики — стратегии, определяющей, какое действие следует

предпринимать в каждом состоянии, чтобы достичь наивысшей возможной ценности. Оптимальная политика направляет агента таким образом, чтобы он принимал решения, способствующие наибольшему накоплению вознаграждений на протяжении всей задачи. Методы, такие как динамическое программирование, методы Монте-Карло и Q-обучение, позволяют агенту обучаться оптимальной политике путем оценки ценности состояний и действий на основе полученных вознаграждений.

- **Q-обучение (Q-Learning)**: Один из самых популярных алгоритмов обучения с подкреплением, в котором вычисляется Q-функция Q(s,a)Q(s, a)Q(s,a), определяющая ценность выполнения действия aaa в состоянии sss. Алгоритм Q-обучения работает по следующему обновлению:

Q(s,a)=Q(s,a)+α[R(s,a)+γmax⁡a′Q(s′,a′)−Q(s,a)]Q(s, a) = Q(s, a) + \alpha \left[R(s, a) + \gamma \max_{a'} Q(s', a') - Q(s, a) \right]Q(s,a)=Q(s,a)+α[R(s,a)+γa′max Q(s′,a′)−Q(s,a)]

где α\alphaα — коэффициент обучения, γ\gammaγ — коэффициент дисконтирования, s′s's′ — следующее состояние после выполнения действия aaa.

Q-обучение является методом безмодельного обучения, что позволяет агенту действовать в неизвестной среде. Агент накапливает опыт и обновляет свои оценки на основе наблюдений, постепенно улучшая политику.

- **SARSA (State-Action-Reward-State-Action)**: Метод, похожий на Q-обучение, но с разницей в способе выбора действия. В SARSA обновление Q-значений происходит на основе текущего действия и будущего действия, определенного политикой. Этот метод позволяет учитывать текущую политику агента и обеспечивает более стабильное обучение.

4.2 Методы политики (Policy-based methods)

Методы политики позволяют агенту непосредственно оптимизировать политику, минуя вычисление функции ценности. Эти методы полезны, когда пространство действий слишком велико или требует детальной настройки, как, например, в задачах с непрерывными действиями.

- **REINFORCE**: Один из основных алгоритмов методов политики. В основе REINFORCE лежит метод градиентного спуска, который оптимизирует вероятность выбранного действия, направляя его в

сторону увеличения ожидаемой награды. Обновление параметров политики определяется следующим образом:

θ=θ+α∇θlogπ(a|s;θ)·R\theta = \theta + \alpha \nabla_\theta \log \pi(a|s; \theta) \cdot Rθ=θ+α∇θlogπ(a|s;θ)·R

где π(a|s;θ)\pi(a|s; \theta)π(a|s;θ) — политика агента, параметризованная θ\thetaθ, а RRR — совокупная награда.

4.3 Акторно-критические методы (Actor-Critic Methods)

Акторно-критические методы сочетают методы ценности и политики, что позволяет агенту одновременно обновлять и политику, и функцию ценности. Актор отвечает за выбор действий, а критик оценивает ценность состояния, обеспечивая актера обратной связью.

5. Глубокое обучение с подкреплением (Deep Reinforcement Learning)

Глубокое обучение с подкреплением (Deep Reinforcement Learning, DRL) — это расширение ОП, в котором используются нейронные сети для представления функции ценности или политики. Основная идея DRL заключается в том, чтобы использовать нейронные сети для обработки сложных и высокоразмерных данных, таких как изображения.

5.1 Глубокое Q-обучение (Deep Q-Learning, DQN)

DQN — это метод, в котором используется глубокая нейронная сеть для аппроксимации функции Q. Архитектура DQN включает в себя две сети — основную и целевую. Основная сеть обучается на текущих данных, а целевая используется для стабилизации обучения.

DQN позволило добиться значительных успехов в решении сложных задач, таких как игра в Atari, где агент обучается игре, наблюдая за игровым экраном.

5.2 Акторно-критические методы в DRL

Акторно-критические методы, такие как A3C (Asynchronous Advantage Actor-Critic), используют нейронные сети как для актера, так и для критика. A3C обучается параллельно на нескольких потоках, что ускоряет обучение и делает его более стабильным.

6. Применение обучения с подкреплением

Обучение с подкреплением находит широкое применение в самых разных областях:

- **Робототехника**: Использование агентов для автоматизации движений роботов, адаптации к окружающей среде и выполнения сложных задач.
- **Игровая индустрия**: Агент, обученный с помощью ОП, может достигать высокого уровня мастерства в играх, таких как шахматы, го и видеоигры.
- **Автономные системы**: ОП используется для создания автономных автомобилей и дронов, способных адаптироваться к изменяющимся условиям.
- **Финансы**: ОП применяется для автоматической торговли и оптимизации инвестиционных стратегий.

7. Проблемы и вызовы в обучении с подкреплением

Несмотря на достижения, ОП сталкивается с рядом проблем:

- **Стабильность обучения**: Процесс обучения с подкреплением часто нестабилен и может привести к нежелательным результатам.
- **Размерность состояния и действия**: Сложные задачи с большим количеством возможных состояний требуют огромных вычислительных ресурсов.
- **Эксплоратация и эксплуатация**: Баланс между исследованием новых действий и использованием уже известных успешных действий остается важной проблемой.
- **Безопасность и этика**: В некоторых приложениях, таких как автономные транспортные средства, ошибки агента могут приводить к серьезным последствиям.

8. Перспективы и будущее обучение с подкреплением

Будущее ОП связано с развитием более стабильных и эффективных алгоритмов, способных справляться с задачами высокой сложности и адаптироваться к динамическим условиям.