یادگیری تقویتی

علیرضا محمدی

انواع یاد گیری ماشین

- یادگیری با نظارت
- یادگیری بدون نظارت
- یادگیری تقویتی
- یادگیری نیمه نظارتی

1. یادگیری نظارت‌شده SUPERVISED LEARNING

در این روش، مدل از **داده‌های برچسب‌دار** استفاده می‌کند. این بدان معناست که ورودی‌ها همراه با خروجی‌های مشخص (برچسب) به مدل داده می‌شوند.

هدف: یادگیری نگاشت میان ورودی و خروجی، به‌طوری که مدل بتواند برای داده‌های جدید پیش‌بینی دقیق انجام دهد.

کاربردها:

- طبقه‌بندی Classification : پیش‌بینی دسته‌ای که یک نمونه به آن تعلق دارد (مانند شناسایی ایمیل‌های اسپم).
- رگرسیون Regression : پیش‌بینی مقادیر عددی (مانند پیش‌بینی قیمت خانه).

مثال:

- پیش‌بینی نمره دانش‌آموزان با استفاده از داده‌هایی شامل ساعات مطالعه و نمرات قبلی.

Linear Regression, SVM,

2. یادگیری بدون نظارتUNSUPERVISED LEARNING

در این روش، داده‌ها **بدون برچسب** هستند و مدل باید ساختار یا الگوهای پنهان موجود در داده‌ها را کشف کند.

هدف: کشف الگوها و ارائه‌ی بینشی در مورد داده‌ها.

کاربردها:

- خوشه‌بندی Clustering : گروه‌بندی داده‌ها بر اساس شباهت‌ها (مانند تقسیم مشتریان به دسته‌های مشابه برای بازاریابی).
- کاهش ابعاد Dimensionality Reduction : ساده‌سازی داده‌ها با کاهش تعداد ویژگی‌ها (مانند PCA برای فشرده‌سازی داده).

مثال:

- گروه‌بندی تصاویر مشابه از مجموعه‌ای بدون برچسب.

K-Means, PCA,

3. یادگیری تقویتی REINFORCEMENT LEARNING

این روش بر اساس تعامل مدل عامل یا (Agent)با یک **محیط پویا** کار می‌کند. عامل بر اساس اقدامات خود، بازخوردی در قالب **پاداش** دریافت می‌کند و هدف آن یادگیری سیاستی است که **پاداش تجمعی بلندمدت** را به حداکثر برساند.

هدف: یافتن بهترین مجموعه اقدامات برای بهبود عملکرد در طول زمان.

کاربردها:

- بازی‌ها مانند شطرنج.
- کنترل ربات‌ها.
- مدیریت منابع در سیستم‌های پیچیده.

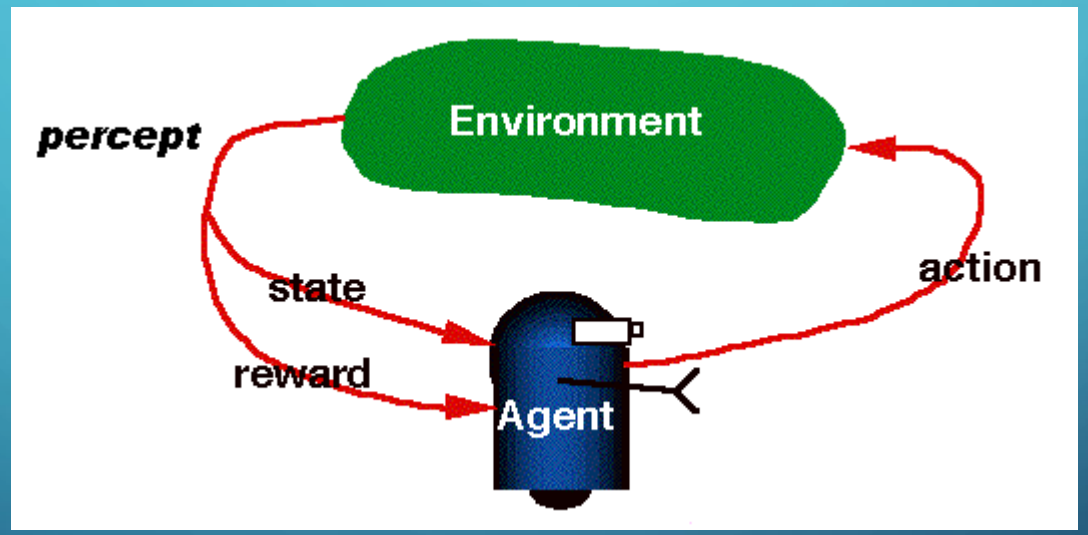
مثال:

- آموزش یک ربات برای یادگیری نحوه حرکت در محیط ناشناخته.

Q-Learning, PPO, DQN

ویژگی	یادگیری با نظارت	یادگیری بدون نظارت	یادگیری تقویتی
داده‌ها	برچسب‌دار	بدون برچسب	تعامل با محیط
هدف	پیش‌بینی دقیق	کشف الگوها	بیشینه‌سازی پاداش
کاربردها	پیش‌بینی و دسته‌بندی	خوشه‌بندی و اکتشاف	تصمیم‌گیری و کنترل

اجزا اصلی یادگیری تقویتی



اجزا اصلی یادگیری تقویتی

1. عامل (ساده،هدفگرا، سودمندی، یادگیرنده)

عامل، موجودیت یا برنامه‌ای است که تصمیم می‌گیرد چه کاری انجام دهد. هدف اصلی عامل، یادگیری از طریق تعامل با محیط است تا پاداش بیشتری به دست آورد.
مثال: یک ربات که در اتاق حرکت می‌کند و اشیای خاصی را جمع‌آوری می‌کند.

2. محیط (environment)

محیط شامل همه چیزهایی است که عامل با آن در تعامل است، از قوانین گرفته تا شرایط اطراف. وقتی عامل کاری انجام می‌دهد، محیط به آن بازخورد می‌دهد، معمولاً به شکل پاداش یا تغییر وضعیت.
مثال: تخته شطرنج برای یک عامل بازی‌کننده شطرنج.

3. حالت (state)

حالت، مشاهده ایی از وضعیت محیط است که عامل در هر لحظه در آن قرار دارد.
مثال: موقعیت فعلی مهره‌های شطرنج روی تخته.

4. اقدام (action)

اقدام، کاری است که عامل می‌تواند در یک حالت خاص انجام دهد. همه اقدامات ممکن، فضایی به نام "فضای اقدام" تشکیل می‌دهند.
مثال: حرکت مهره سرباز در شطرنج.

5.پاداش (reward)

پاداش، یک عدد است که عامل پس از انجام یک اقدام در یک حالت دریافت می‌کند. این عدد به عامل می‌گوید که آیا عملکردش خوب بوده یا نه.
هدف عامل: جمع‌آوری بیشترین پاداش در طول زمان.
مثال‌ها: +10 برای رسیدن به هدف, -1 برای برخورد با مانع.

6.سیاست (policy)

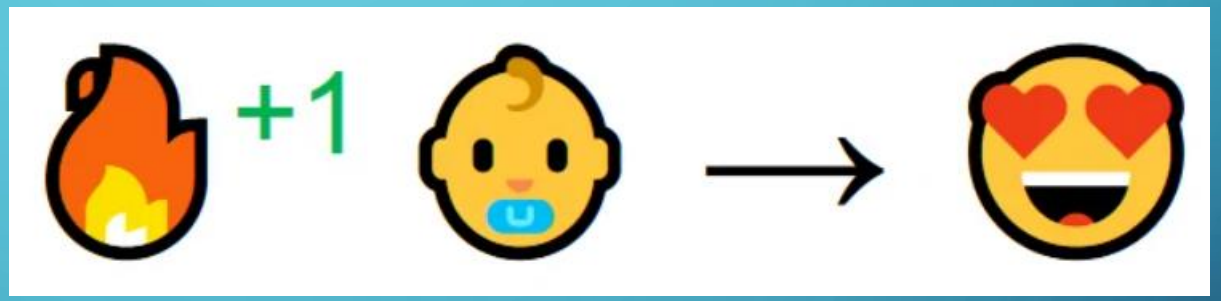
سیاست، یک برنامه یا راهنما است که مشخص می‌کند عامل در هر حالت چه کاری انجام دهد.
انواع سیاست:

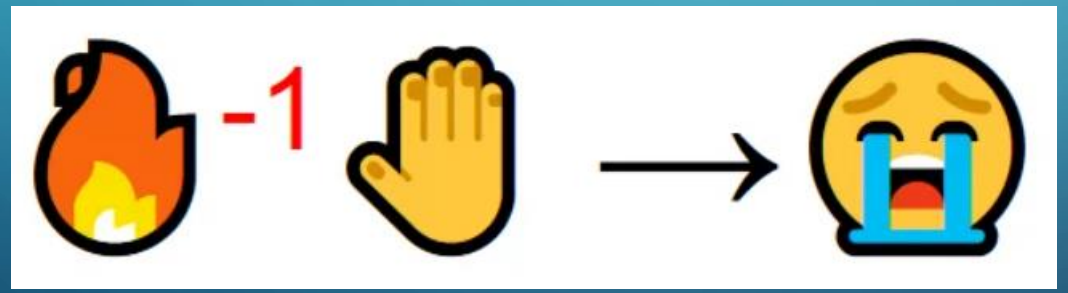
- قطعی: برای هر حالت، دقیقاً یک اقدام مشخص می‌شود.
- احتمالی :احتمال انجام هر اقدام تعریف می‌شود.
مثال: اگر در گوشه زمین باشی، به سمت مرکز حرکت کن.

7.تابع ارزش (value function)

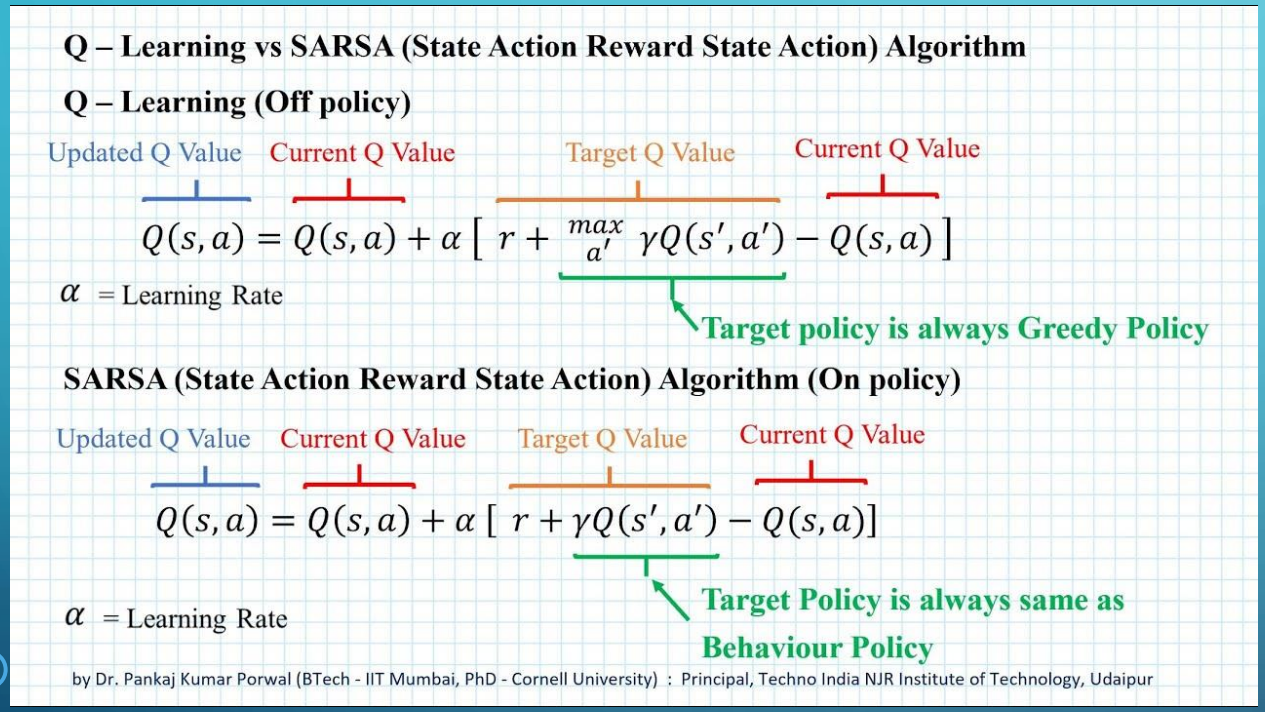
تابع ارزش، به عامل کمک می‌کند تخمین بزند که یک حالت یا اقدام چقدر خوب است و در بلندمدت چه پاداشی به همراه دارد.
دو نوع اصلی:

- **تابع ارزش حالت :** مشخص می‌کند که بودن در یک حالت چقدر ارزشمند است.
- **تابع ارزش اقدام :** نشان می‌دهد که انجام یک اقدام در یک حالت خاص چقدر خوب است.
مثال: ارزش حرکت یک مهره شطرنج ممکن است به احتمال برد یا بهبود وضعیت تخته بستگی داشته باشد.


+1


-1

الگوریتم‌های معروف یادگیری تقویتی



- **الف) الگوریتم‌های کلاسیک (Classic RL)**

1. **مبتنی بر ارزش، بدون مدل (Q-Learning)**

یادگیری ارزش حالت-اقدام Q(s,a)

ویژگی: یادگیری خارج از سیاست

2. SARSA (State, Action, Reward, State', Action')
مبتنی بر ارزش، بدون مدل

مشابه Q-Learningاما
یادگیری داخل سیاست (On-Policy)

ویژگی	Q-Learning	SARSA
سیاست یادگیری	یادگیری خارج از سیاست (Off-Policy)	یادگیری داخل سیاست (On-Policy)
تصمیم‌گیری	بر اساس بهترین اقدام ممکن در حالت بعدی	بر اساس اقدامی که عامل طبق سیاست فعلی‌اش انجام می‌دهد
کاربرد	مناسب برای کاوش و یادگیری کلی	مناسب برای بهبود تدریجی یک سیاست مشخص

- ب) **الگوریتم‌های یادگیری عمیق تقویتی (Deep Reinforcement Learning)**

تعریف کلی:

الگوریتم‌های یادگیری عمیق تقویتی ترکیبی از **یادگیری تقویتی RL** و **یادگیری عمیق DL**هستند. در این روش‌ها، از **شبکه‌های عصبی عمیق** برای تقریب توابع پیچیده مانند تابع ارزش یا سیاست استفاده می‌شود. این رویکرد برای حل مسائل پیچیده‌تر با فضای حالت و اقدام بزرگ طراحی شده است.

1: Deep Q-Network (DQN)

ترکیب Q-Learning با شبکه‌های عصبی برای حل مسائل با فضای حالت بسیار بزرگ.

به جای ذخیره مقدار Q(s,a) در یک جدول، از یک شبکه عصبی برای تخمین آن استفاده می‌کنیم.

دو ابزار کلیدی در DQN

Replay Buffer : تجربیات گذشته عامل را ذخیره می‌کند و به‌صورت تصادفی از آن‌ها استفاده می‌شود تا داده‌ها همبستگی کمتری داشته باشند.

Target Network: یک کپی از شبکه اصلی برای پایدار کردن یادگیری استفاده می‌شود.

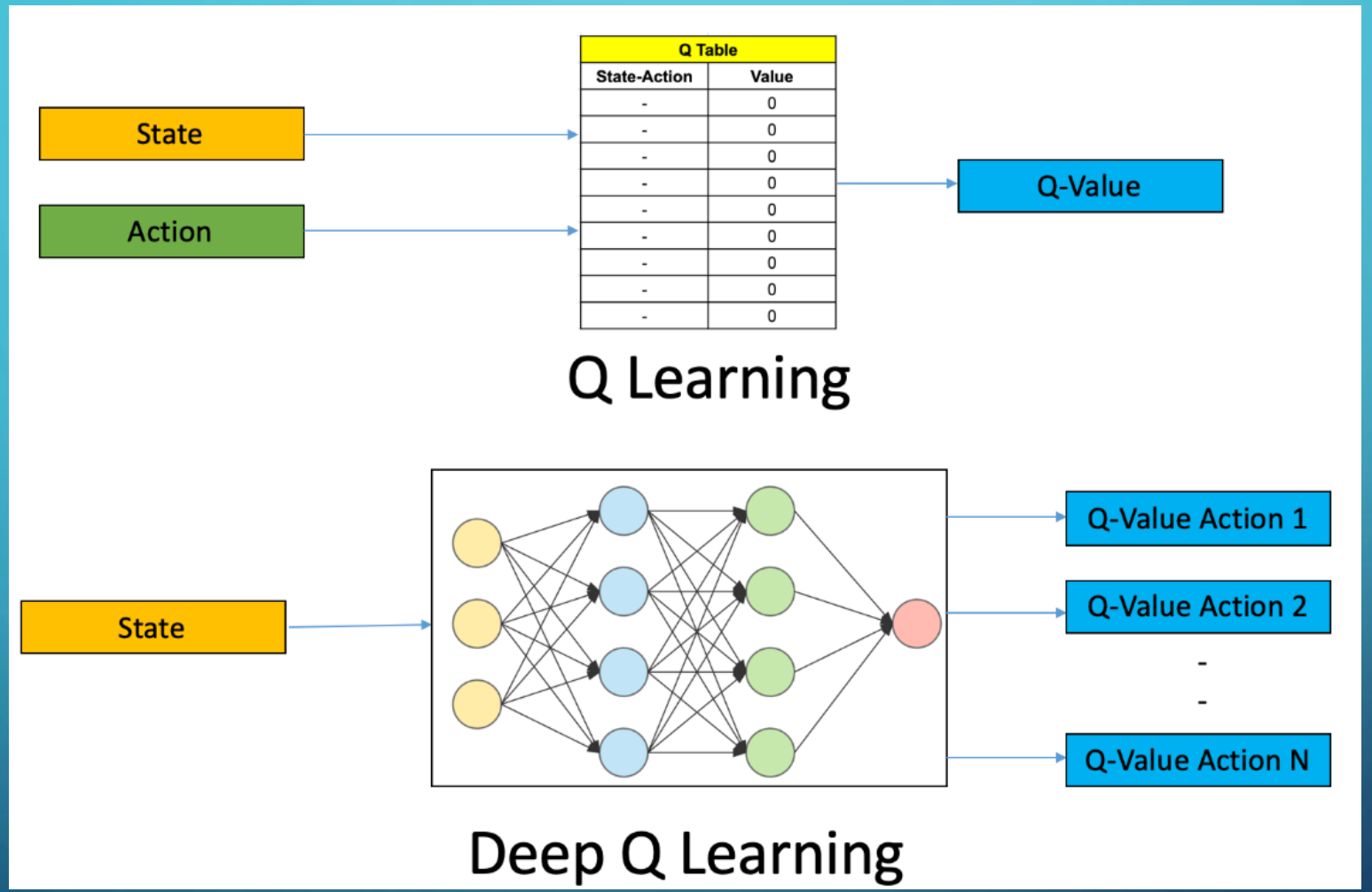
مثال ساده:

فرض کنید عامل در حال بازی کردن بازی آتاری Breakoutاست عامل تصویر صفحه بازی را می‌بیند و شبکه عصبی تخمین می‌زند کدام حرکت (چپ یا راست) بیشترین پاداش را دارد و عامل حرکت می‌کند، پاداش دریافت می‌کند و شبکه Qرا به‌روزرسانی می‌کند.

ویژگی‌ها:

مناسب برای مسائل گسسته Discrete

نقطه عطف در یادگیری تقویتی، اولین بار توسط DeepMindبرای حل بازی‌های آتاری معرفی شد.


State
Action
Q Table
State-Action
Value
-
0
Q-Value
Q Learning
State
Q-Value Action 1
Q-Value Action 2
-
-
Q-Value Action N
Deep Q Learning

A2C, A3C :2

ایده اصلی:

ترکیبی از دو بخش:

Actor: یاد می‌گیرد چه اقدامی در هر حالت انجام دهد (سیاست).

Critic: ارزیابی می‌کند که این اقدام چقدر خوب بوده است (ارزش).

نحوه کار:

Actor و Critic هر دو توسط شبکه‌های عصبی جداگانه مدل‌سازی می‌شوند.

Actor سیاست را تولید می‌کند، و Critic بازخورد می‌دهد تا Actor بهتر شود.

در A3C چند عامل Agents به صورت موازی اجرا می‌شوند و تجربیات خود را به اشتراک می‌گذارند.

مثال ساده:

تصور کنید عامل یک بازوی رباتیک است که باید یک جسم را بلند کند:

1. Actor تصمیم می‌گیرد که چگونه بازو را حرکت دهد.
2. Critic بازخورد می‌دهد که این حرکت چقدر به دستیابی به هدف کمک کرده است.
3. Actor بر اساس این بازخورد سیاست خود را بهبود می‌دهد.

ویژگی‌ها:

مناسب برای مسائل پیچیده با فضای حالت و اقدام بزرگ.

از اجرای موازی برای افزایش سرعت یادگیری استفاده می‌کند.

الگوریتم	معایب	مزایا	ویژگی بارز	نوع
Q-Learning	مناسب برای فضای گسسته	ساده و کارا	یادگیری Off-Policy	ارزش
SARSA	کندتر از Q-Learning	سیاست پایدارتر	یادگیری On-Policy	ارزش
DQN	پیچیدگی محاسباتی بیشتر	مناسب فضای حالت بزرگ	استفاده از شبکه عصبی	ارزش
A3C	نیازمند منابع بیشتر	سرعت بیشتر	یادگیری موازی	Actor-Critic

کاربردهای یادگیری تقویتی

رباتیک:

- یادگیری حرکات پیچیده برای مسیریابی و دستکاری اشیاء.
- مثال: ربات‌هایی که در کارخانه‌ها کار می‌کنند یا ربات‌های جراحی.

بازی‌ها:

- موفقیت چشمگیر در بازی‌هایی مانند شطرنج و Go.
- مثال: AlphaGo از یادگیری تقویتی برای شکست قهرمان جهانی استفاده کرد.

وسایل نقلیه خودمختار:

- تصمیم‌گیری در زمان واقعی برای رانندگی ایمن و کارآمد.
- مثال: پارک کردن ماشین با آزمون‌وخطا.

مدیریت منابع:

- بهینه‌سازی تخصیص منابع و زمان‌بندی.
- مثال: مدیریت انرژی یا ترافیک.

محدودیت‌های یادگیری تقویتی

کارایی نمونه Sample Efficiency

- نیاز به تعامل‌های فراوان برای یادگیری.
- بسیار محاسباتی و زمان‌بر.

تریدآف اکتشاف و بهره‌برداری Exploration vs. Exploitation

- تعادل میان کاوش اقدامات جدید و بهره‌برداری از دانش موجود دشوار است.

مهندسی پاداش Reward Engineering

- طراحی پاداش مناسب بسیار پیچیده است.
- پاداش نادرست ممکن است به رفتارهای غیرمنتظره منجر شود.

ملاحظات اخلاقی Ethical Considerations

- ممکن است عامل رفتارهای مضر یا غیراخلاقی بیاموزد

- Book
  https://www.amazon.com/Deep-Reinforcement-Learning-Hands-Q-networks/dp/1788834240

- Code
  https://github.com/PacktPublishing/Deep-Reinforcement-Learning-Hands-On