# 几个概念

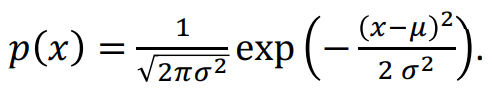
## Probability Density Function (PDF) 概率密度函数

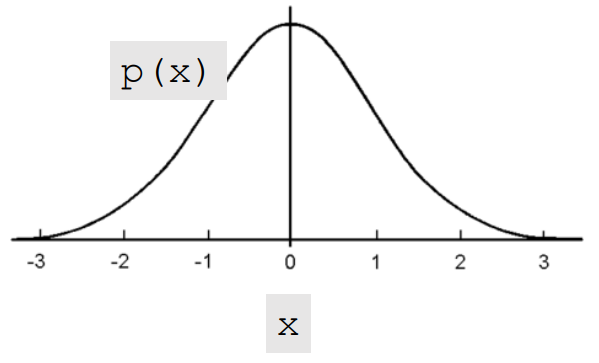
PDF provides a relative likelihood that the value of the random variable would equal that sample

**Example**: Gaussian distribution

It is a continuous distribution.

PDF 概率密度函数:



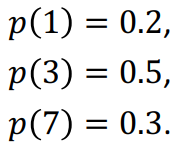


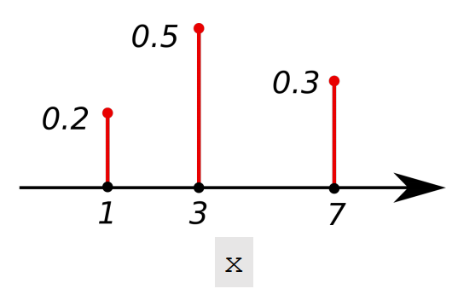
这张图里，横轴是随机变量X的取值，纵轴是概率密度，这条曲线就是高斯分布的密度函数p(x)，这个概率密度说明X在原点附近取值的概率比较大，在远离远点的地方取值的概率比较小

Example: Discrete distribution

Discrete random variable: X ∈ {1, 3, 7} .

PDF 概率密度函数:





取值是1的概率是0.2

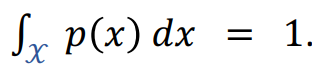
取值是3的概率是0.5

取值是7的概率是0.3

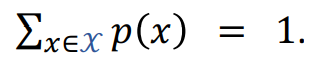
把所有可能的取值都算上，概率取值的加和或者积分等于1

Random variable X is in the domain

For continuous distribution,



For discrete distribution



## Expectation 期望

Random variable X is in the domain

For continuous distribution, the expectation of f(x) is:



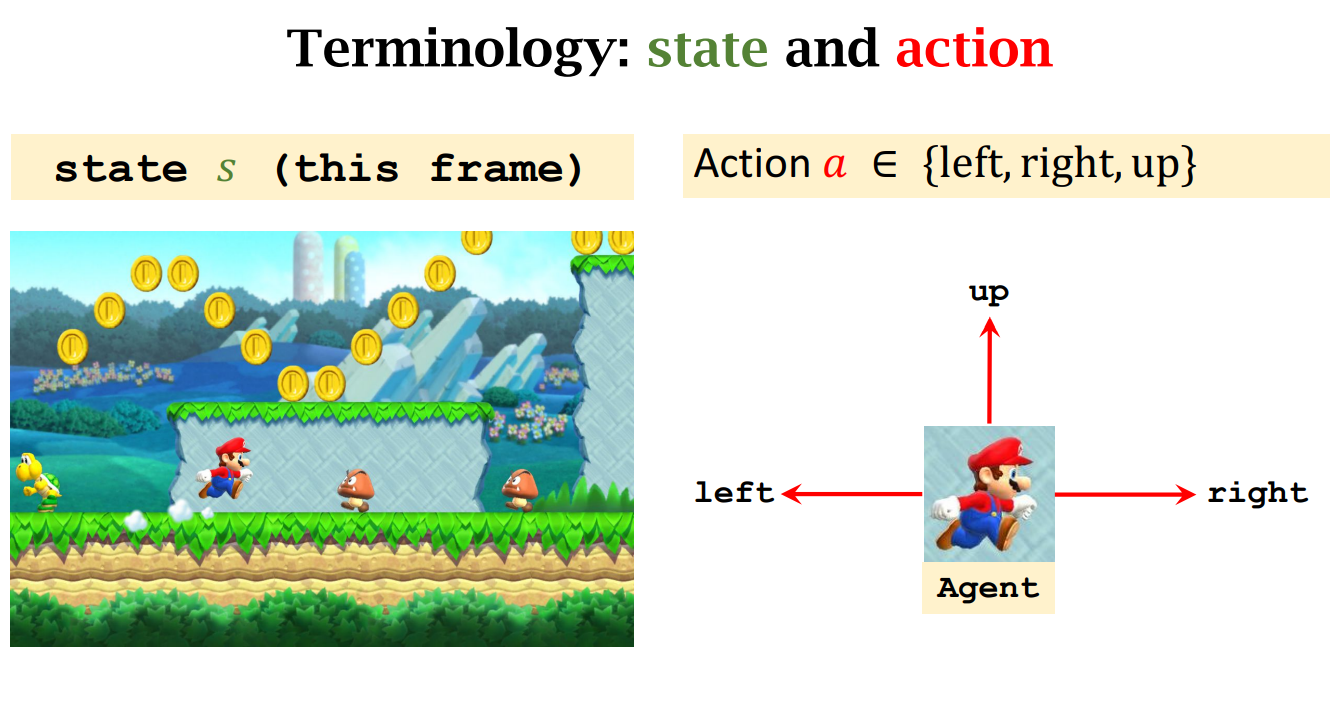
For discrete distribution, the expectation of f(x) is:

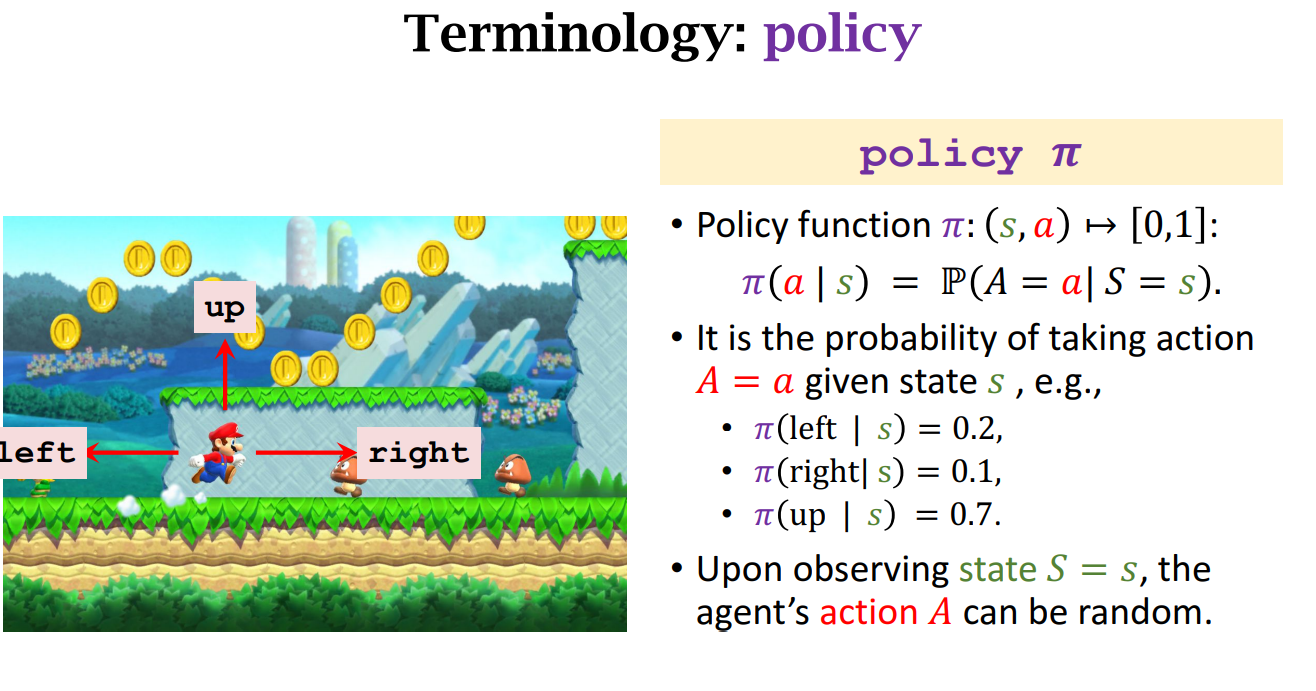


## Random Sampling 随机采样

# Terminologies

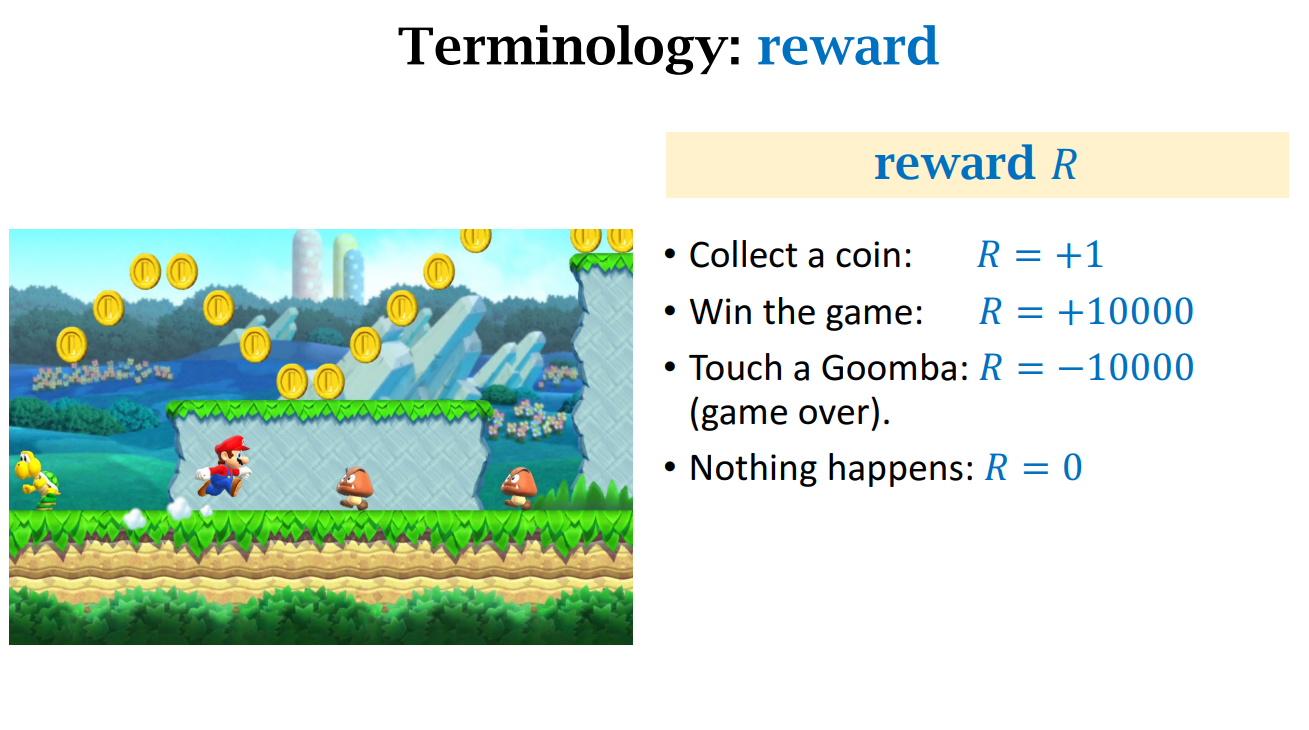
## Terminology: state and action

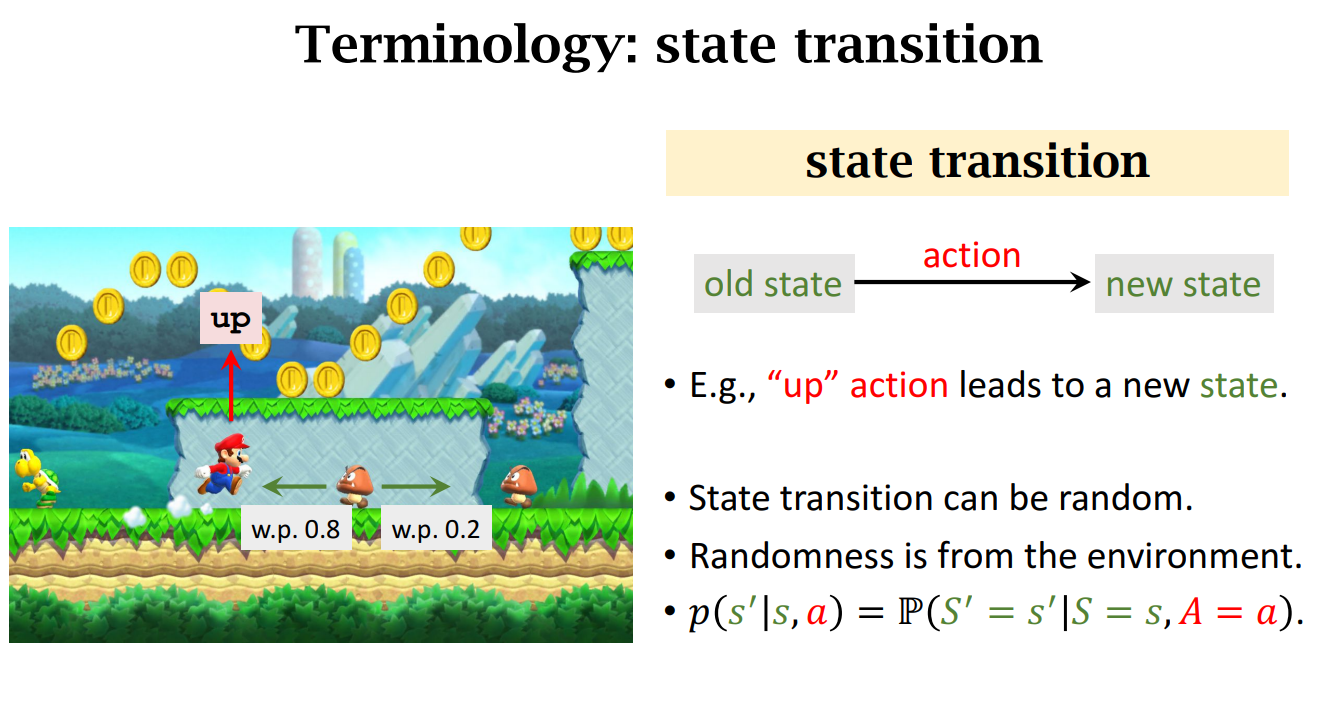




强化学习就是学这个policy函数

这个动作最好是随机，比如拳击，如果你的动作很确定，别人就有办法赢。所以，很多应用里面，policy最好是个概率密度函数，动作最好是随机抽样得到的，要有随机性。





状态转移时随机的，因为环境是随机的，比如马里奥网上跳这个东西，敌人可能往左或者往右，敌人不同的动作会影响马里奥的下一个状态。

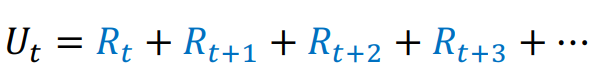


是一个条件概率密度函数，意思是如果观察到当前得状态s和动作a，p函数输出状态s’的概率

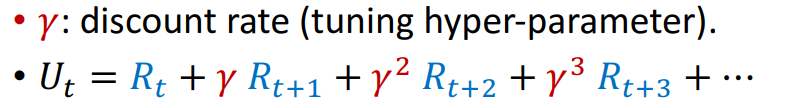
## Rewards and Returns

### Return

Definition: Return (aka cumulative future reward)



Definition: Discounted return (aka cumulative discounted future reward).



折扣率是个超参数，需要我们自己来调

### Value Function 价值函数

Action-Value Function Q(s,a)

Definition: Discounted return (aka cumulative discounted future reward).



Definition: Action-value function for policy �

State-Value Function Q(s,a)

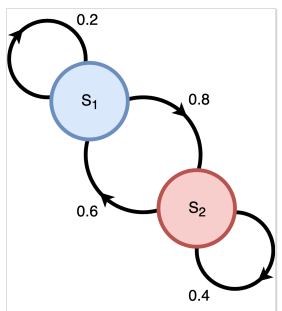
# 马尔科夫决策

## 一、马尔科夫属性

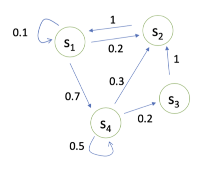
马尔可夫属性（Markov property）是概率论中的一个概念。意思是当在一个随机过程在给定现在状态及所有过去状态情况下，其未来状态的条件概率分布**仅依赖于当前状态**；其实也就是一句话，这个事件的状态的下一秒变化是由这个状态决定，和以前没关系。

## 二、马尔科夫链

马尔可夫链是一组具有马尔可夫性质的离散随机变量的集合



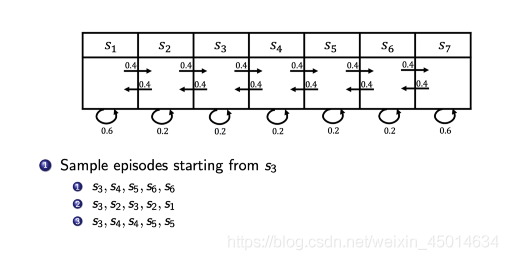
他是一组概率转换的一组集合。其具有马尔科夫性质、存在离散指数集和状态空间的随机过程。如下面：



其马尔科夫链就是从这个图有四个状态，s1,s2,s3,s4这个相互转移。可以从任意状态开始，比如s1

* s1有0.1概率继续s1，有0.7概率变成s4, 0.2概率变成s2  
  其他依次类似

因此在数学中我们可以构建状态转移矩阵。



由状态转移矩阵去描述到其他节点概率

## 三、马尔科夫奖励过程

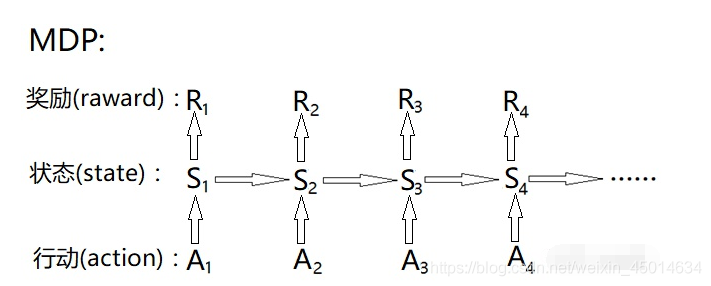
这个过程就是在原来马尔科夫链上加上一个奖励矩阵。也就是在上述所说的，我们四个过程，如果说我们这个过程到达s4节点就比较好我们就奖励他10个值。到达s1就奖励5即：[5,0,0,10]

我们在加上奖励之后，这个马尔科夫链就将会趋向于s1和s4了

## 四、马尔可夫决策过程（MDP）

在前面三种定义介绍之后，我们知道强化学习这是一个行为学习，所以我们要应该根据信息去做一些事情。因此我们就该加上一个行为，

在MDP中，我们为系统增加一个行为（action）变量 A t AtAt 与一个“奖励”（reward）变量 R ( t ) R(t)R(t) 。其中 A ( t ) A(t)A(t)代表了我们“主观能动性”的部分，相当于系统的“输入”；而 R ( t ) R(t)R(t) 代表着在 A ( t ) A(t)A(t)时刻我们采取的行动带来的回报，相当于系统的“输出”。



## 五、MDP

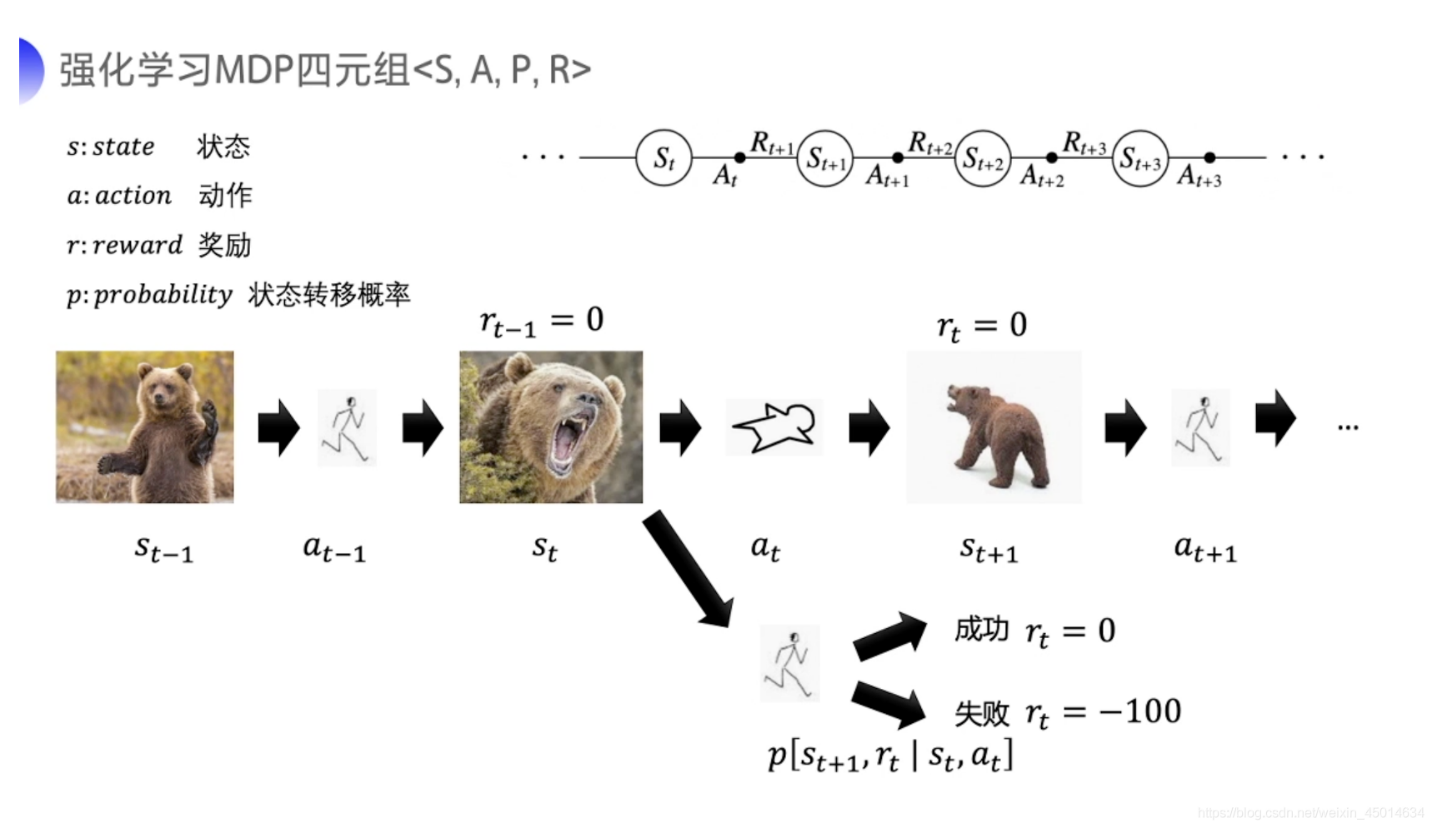
当我们定义了状态、动作、奖励等要素，并确定它们之间的转移关系具有马尔可夫性质时，就可以定义一个MDP。MDP由四元组而后面根据学习目标，我们可能会加上一个衰减因子的（在后文中介绍）< S , A , P , R > <S,A,P,R><S,A,P,R>即：

s：status（状态）

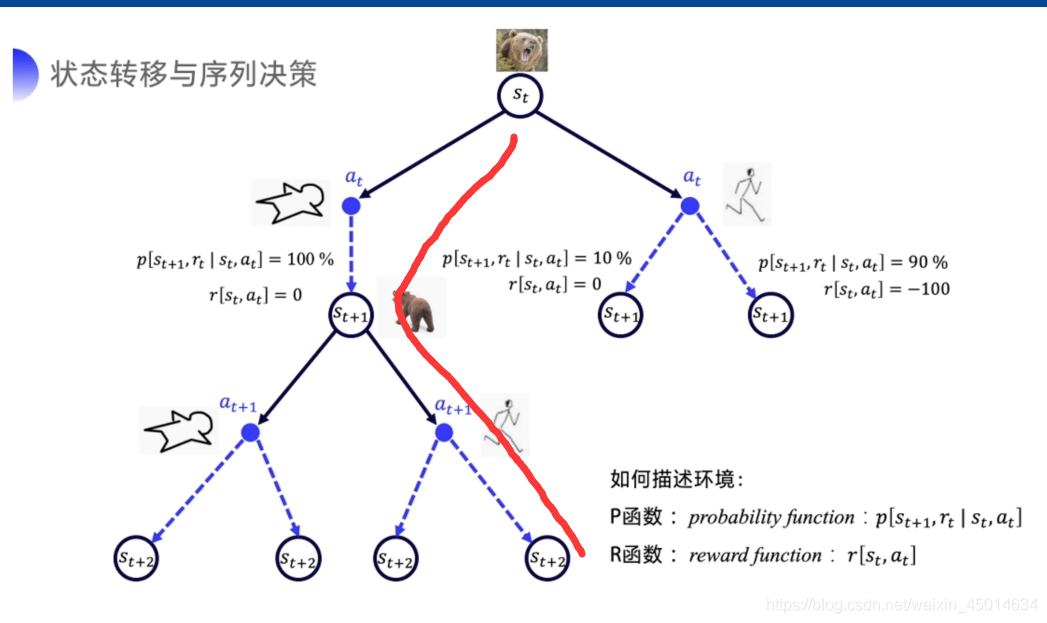
A：action（动作）

R：reward（奖励）

P：probability（状态转移）

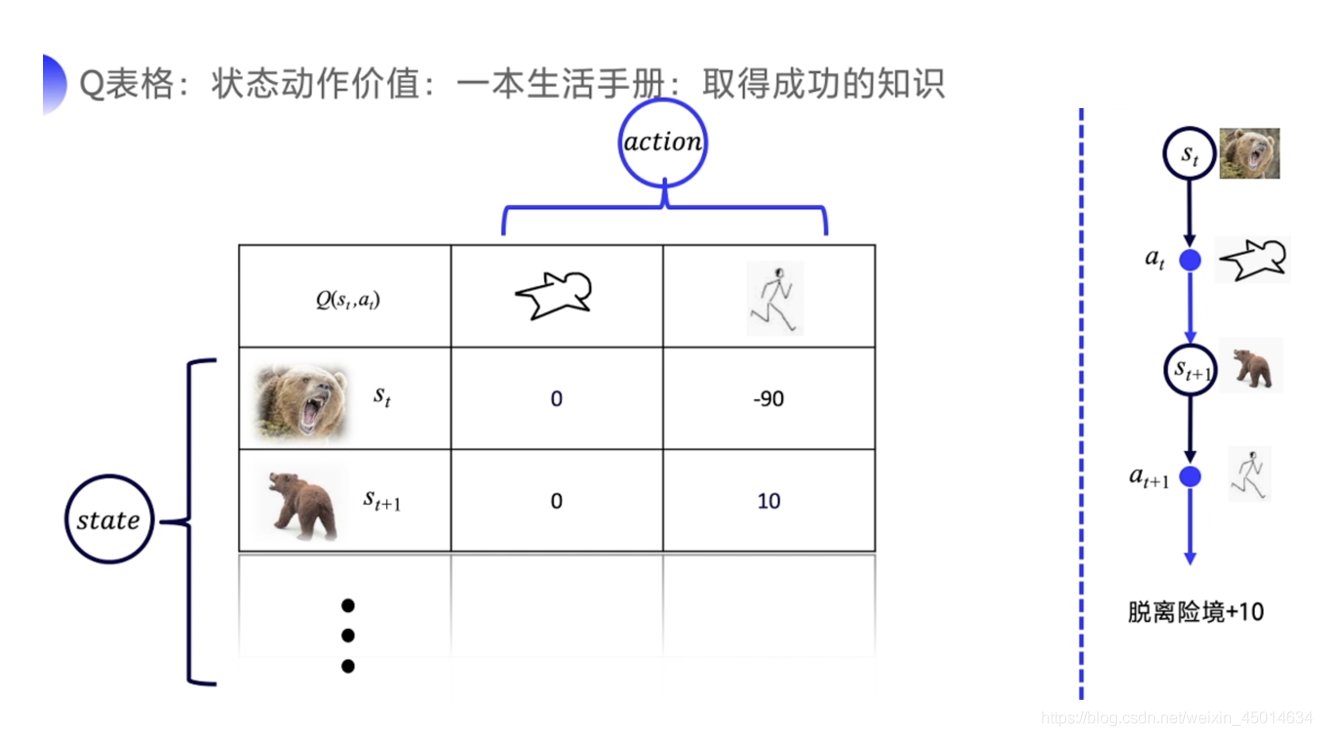


根据图中人物小人，再根据每一刻状态时候，人物小人就会做出一定的决策，而每一刻，熊还是人他们都有不同行为概率的。我们就可把它变成树结构：



根据图片中结果，我们可以定义一个价值函数V去表示好与坏。  
再知道他这个是好还是坏的时候，我们就可以记录下来之后，知道什么样的状态做什么样的事情了。我们把它用Q函数去表示。即：状态–动作值

1）Q-table



我们知道这个训练好的表格之后，这个表格就像小人的生活准则一样，遇到什么样的情况，记录之后查询，做出选择。他指导我们每一个行为动作。

2）折扣因子

在我们有的时候训练目标不同情况下，比如下棋过程， 训练迭代过程特别大，目标有点远，这样下去可能无数迭代，所以我们就加入一个折扣因子，能够加快我们步骤。

而在简单走简单迷宫时：迷宫比较小，我们就不需要加入，目标很近没必要加入了。

而我们的折扣因子一般在0----1之间的。因为我们要活在当下啊。赖以当情况学习。

3）时序差分（Temporal Difference）：

一种Q函数（Q值）的更新方式，也就是可以拿下一步的 Q 值 ,公式是：



4）SARSA算法：

一种更新前一时刻状态的单步更新的强化学习算法，也是一种on-policy策略。该算法由于每次更新值函数需要知道前一步的状态(state)，前一步的动作(action)、奖励(reward)、当前状态(state)、将要执行的动作(action)，即 这几个值，所以被称为SARSA算法。agent每进行一次循环，都会用

( S ( t ) , A ( t ) , R ( t + 1 ) , A ( t + 1 ) ) ( S(t), A(t), R(t+1), A(t+1) )(S(t),A(t),R(t+1),A(t+1)) 来对于Q值更新

<https://blog.csdn.net/weixin_45014634/article/details/109249465>