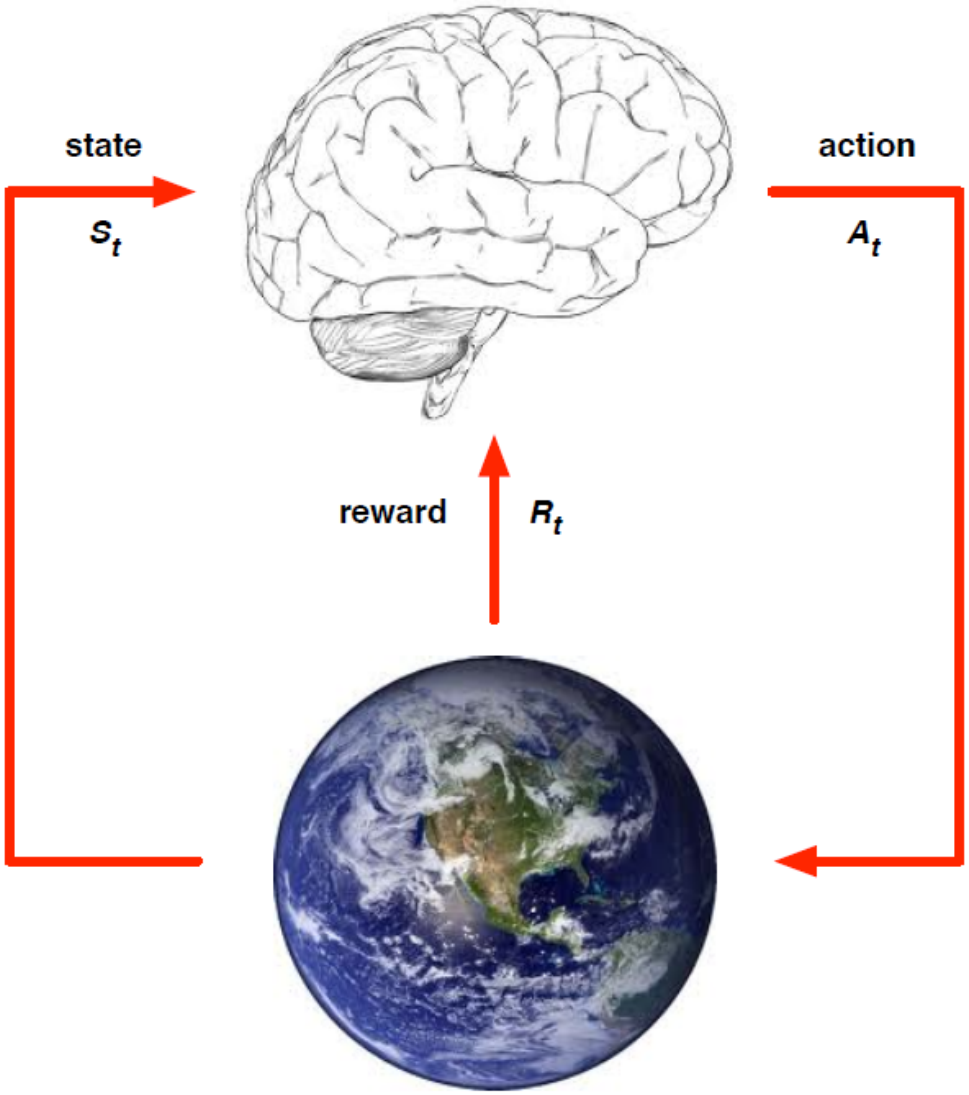
# 强化学习基础篇（三十二）基于模型的强化学习算法

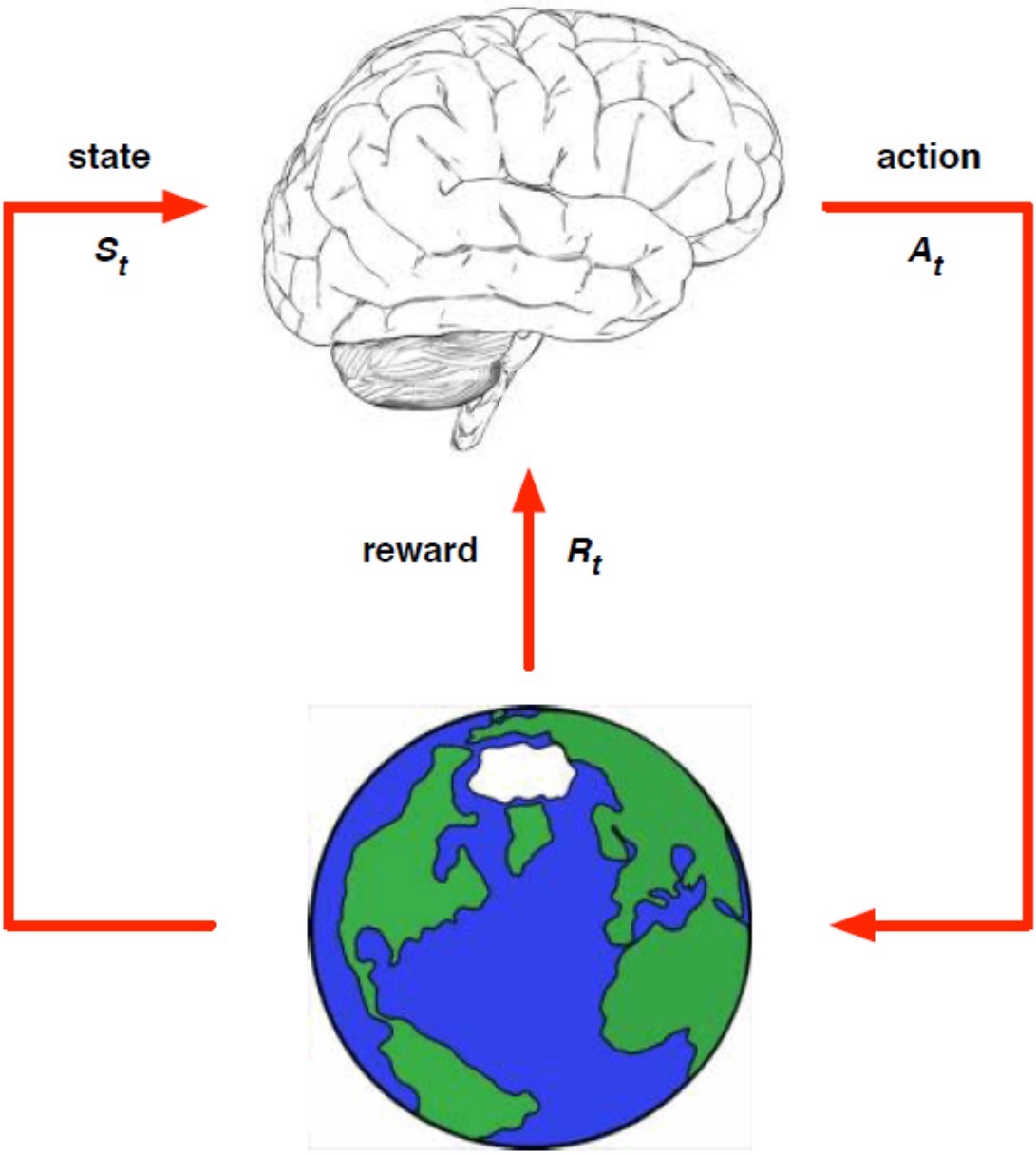
在策略梯度算法中，智能体是直接从经验中去学习策略。之前value-based的方法中，智能体是直接从经验中去学习价值函数（value function），这节我们介绍的基于模型的强化学习算法，是让智能体先去从经验中去学习模型，然后使用规划的方法去构建价值函数或策略。

## 1、Model-Free与Model-Based强化学习

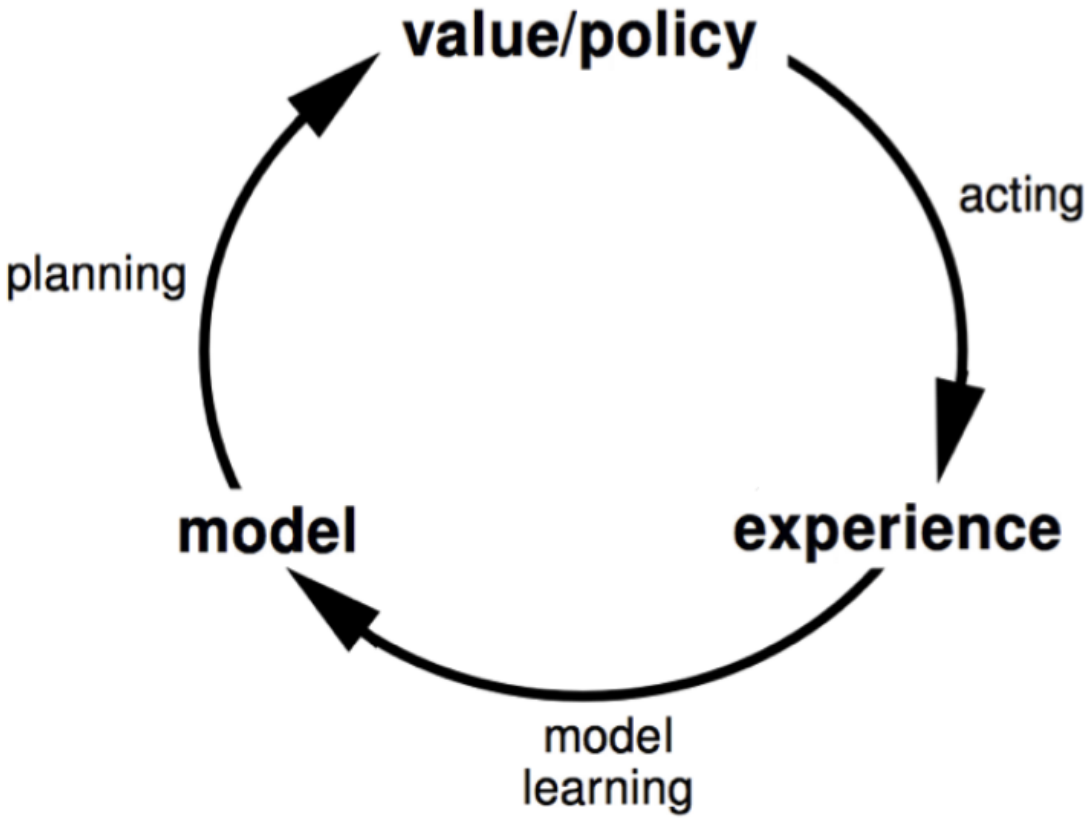
- Model-Free强化学习是智能体没有模型的相关信息，从经验中却学习价值函数与策略。智能体直接与真实环境进行交互。




- Model-Based强化学习是智能体从经验中学习模型，然后从模型去规划价值函数和策略。智能体直接与模拟环境进行交互。




也可以按照下面的图形来表示：




## 2、基于模型的强化学习的算法的优劣

基于模型当前强化学习算法的优点是，我们能够通过监督学习高效率地习得模型，并且由于已知模型的形式，我们可以推断该模型的不确定程度。其缺点是它将引入模型的误差，加上我们值函数估计的误差，这就有了两个误差源。

## 3、模型的学习

### 模型

对于环境建模实际上就是建立MDP模型$< S, A, P, R >$。MDP模型通常包括状态集S，动作集A，转移概率矩阵P以及奖励函数R。一般我们默认智能体是知道状态集S、动作集A的全部信息的，所以我们所谓的对环境建模也就变成了求取P与R：

$$S_{t+1} \sim \mathcal{P}_\eta \left( S_{t+1} \mid S_t, A_t \right)$$

$$R_{t+1} = \mathcal{R}_\eta \left( R_{t+1} \mid S_t, A_t \right)$$

这里，我们假定状态转移分布与奖励分布是独立的：

$$\mathbb{P} \left[ S_{t+1}, R_{t+1} \mid S_t, A_t \right] = \mathbb{P} \left[ S_{t+1} \mid S_t, A_t \right] \mathbb{P} \left[ R_{t+1} \mid S_t, A_t \right]$$

注意，R与值函数V是不一样的，R指的是简单的reward函数，比如下棋，开始一直为0，最后赢了为1，输了为0.而V则会将最后的奖励向前面的状态进行折算。

# 学习模型

模型学习是通过监督学习的方法进行学习的：

$$
\begin{aligned}
S_1, A_1 &\rightarrow R_2, S_2 \\
S_2, A_2 &\rightarrow R_3, S_3 \\
&\vdots \\
S_{T-1}, A_{T-1} &\rightarrow R_T, S_T
\end{aligned}
$$

我们学习奖励函数的过程$s, a \rightarrow r$是一个回归的问题(regression)，并使用MSE作为损失函数，在最小化经验损失的过程中找到奖励函数模型的参数$\eta$。

学习转移概率$s, a \rightarrow s'$是一个密度估计问题（density estimation），使用KL散度作为损失函数，在最小化经验损失的过程中找到转移概率模型的参数$\eta$。

因为是一个监督学习问题，所以我们需要指定假设空间（也即模型的学习范围），比如Table Lookup Model、Linear Expectation Model、Linear Gaussian Model、Gaussian Process Model、Deep Belief Network Model等。下面我们以Table Lookup Model为例来说说如何学习一个模型，并利用该模型进行规划。

# Table Lookup模型的学习

Table Lookup模型的学习可以直接对访问到的$(s, a)$对进行计数来计算转移概率与奖励函数：

$$
\begin{aligned}
\hat{\mathcal{P}}_{s,s'}^{a} &= \frac{1}{N(s,a)} \sum_{t=1}^{T} \mathbf{1}\left(S_t, A_t, S_{t+1} = s, a, s'\right) \\
\hat{\mathcal{R}}_{s}^{a} &= \frac{1}{N(s,a)} \sum_{t=1}^{T} \mathbf{1}\left(S_t, A_t = s, a\right) R_t
\end{aligned}
$$