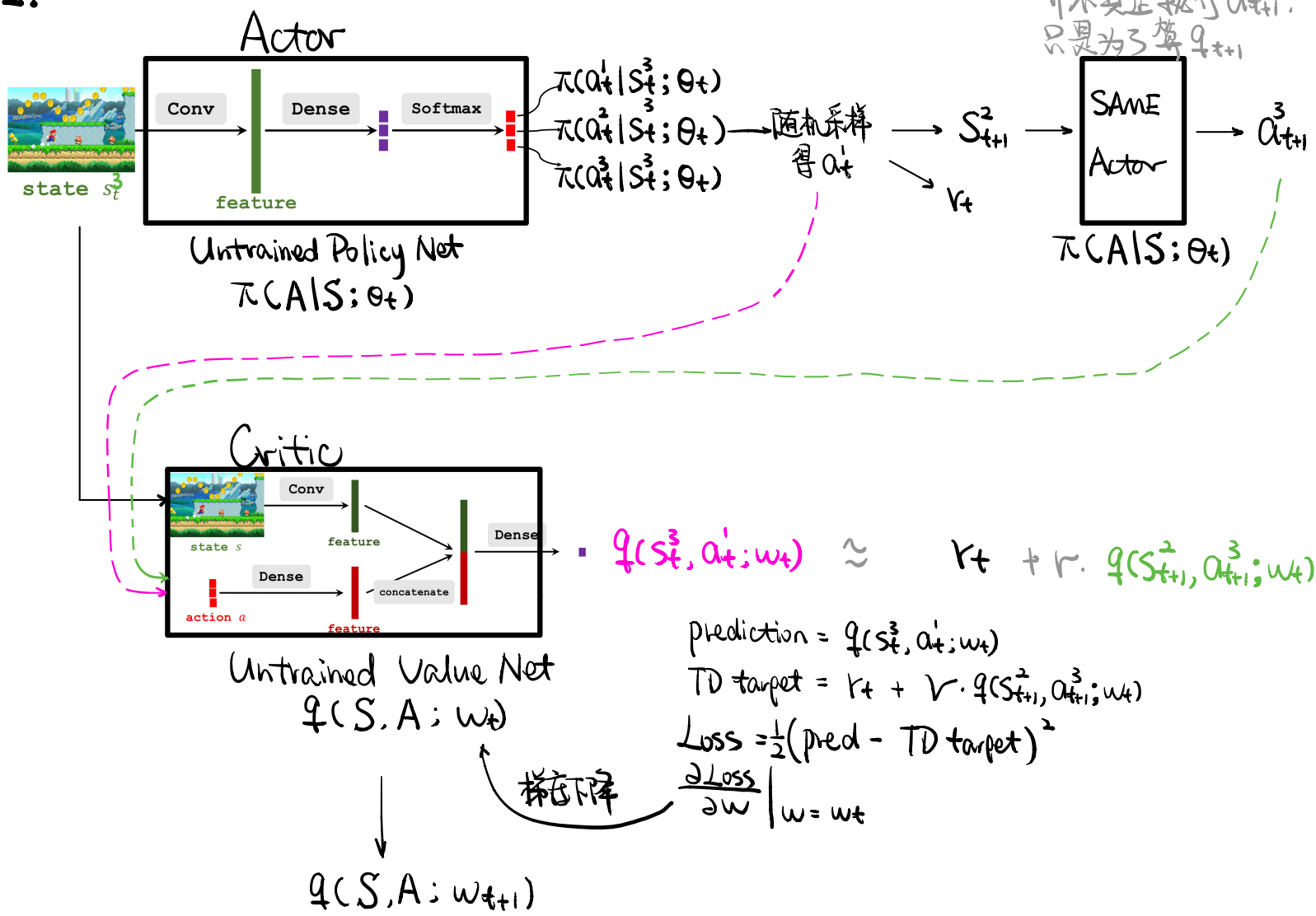
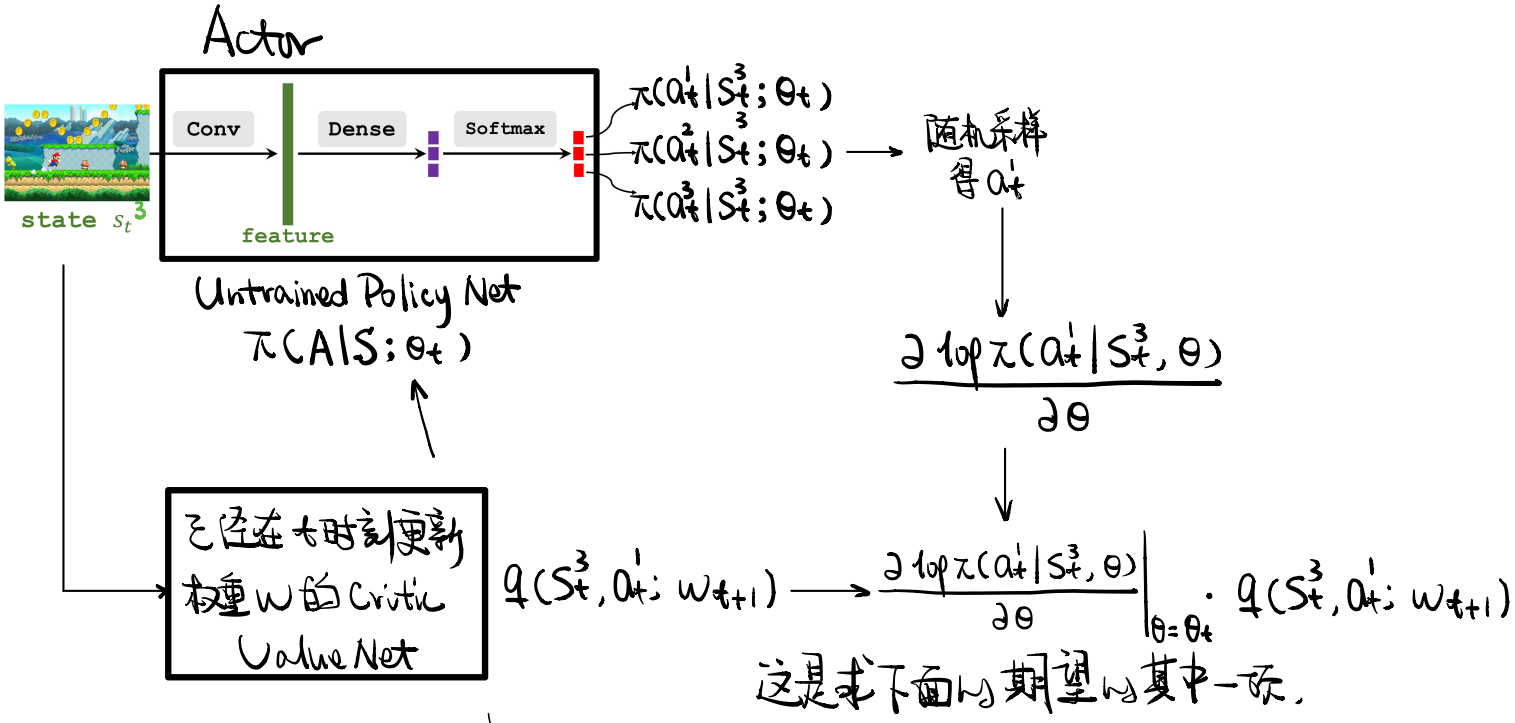
1. Update **w** (in value network) using temporal difference (TD)



2. Update **θ** (in policy network) using policy gradient.



Form 2: $\frac{\partial V(s;\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \mathbb{E}_{A\sim\pi(\cdot|s;\boldsymbol{\theta})}\left[\frac{\partial \log \pi(A|s,\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \cdot Q_\pi(s,A)\right]$

梯度上升

我们用这一项来近似对期望的蒙特卡罗估计

$$\frac{\partial V(s_t^3;\theta)}{\partial \theta} \approx \left.\frac{\partial \log \pi(a_t^1|s_t^3,\theta)}{\partial \theta}\right|_{\theta=\theta_t} \cdot q(s_t^3, a_t^1; w_{t+1})$$

$$= d_{\theta,t} \cdot q_t = g(a_t^1, \theta_t)$$

Policy Net 训练时的"监督"完全来自 Critic. 即 $q(s_t^3, a_t^1; w_{t+1})$.

Critic 的好坏将会影响到 Policy Net 的训练效果;

Critic 训练时的"监督"完全来自环境中的 Reward.

Summary of Algorithm

1. Observe state s_t and randomly sample $a_t \sim \pi(\cdot\,|s_t;\boldsymbol{\theta}_t)$.
2. Perform a_t; then environment gives new state s_{t+1} and reward r_t.
3. Randomly sample $\tilde{a}_{t+1} \sim \pi(\cdot\,|s_{t+1};\boldsymbol{\theta}_t)$. (Do not perform $\tilde{a}_{t+1}$!)

1
4. Evaluate value network: $q_t = q(s_t, a_t; \mathbf{w}_t)$ and $q_{t+1} = q(s_{t+1}, \tilde{a}_{t+1}; \mathbf{w}_t)$.
5. Compute TD error: $\delta_t = q_t - (r_t + \gamma \cdot q_{t+1})$.
6. Differentiate value network: $\mathbf{d}_{w,t} = \left.\frac{\partial q(s_t,a_t;\mathbf{w})}{\partial \mathbf{w}}\right|_{\mathbf{w}=\mathbf{w}_t}$.
7. Update value network: $\mathbf{w}_{t+1} = \mathbf{w}_t - \alpha \cdot \delta_t \cdot \mathbf{d}_{w,t}$.

2
8. Differentiate policy network: $\mathbf{d}_{\theta,t} = \left.\frac{\partial \log \pi(a_t|s_t,\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right|_{\boldsymbol{\theta}=\boldsymbol{\theta}_t}$.
9. Update policy network: $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \beta \cdot q_t \cdot \mathbf{d}_{\theta,t}$.