软演员-评论家（SAC）：原理、公式、应用与实战

2025 年 9 月 21 日

1 引言

软演员-评论家（Soft Actor-Critic，SAC）在最大熵框架下同时最大化期望回报与策略熵，通过双 Q 网络、随机策略与自适应温度，实现连续控制任务中的高性能与稳定性。

2 原理与公式

2.1 最大熵目标

SAC 优化目标为：

$$J(\pi) = \mathbb{E}_{\tau\sim\pi}\Big[\sum_t \gamma^t (r_{t+1} + \alpha \mathcal{H}(\pi(\cdot \mid s_t)))\Big], \tag{1}$$

其中 α 权衡奖励与熵。

2.2 评论家与策略更新

双 Q 网络 Q_{w_i} 最小化：

$$L(w_i) = \mathbb{E}\Big[(Q_{w_i}(s,a) - y)^2\Big], \quad y = r + \gamma(\min_j Q_{w_j^-}(s',a') - \alpha \log \pi_\theta(a' \mid s')). \tag{2}$$

策略通过重参数化技巧更新：

$$\nabla_\theta J_\pi = \mathbb{E}_{s,\varepsilon}\big[\nabla_\theta \alpha \log \pi_\theta(f_\theta(\varepsilon; s) \mid s) - \nabla_\theta Q_{w_1}(s, f_\theta(\varepsilon; s))\big]. \tag{3}$$

2.3 温度自适应

温度 α 通过最小化：

$$J(\alpha) = \mathbb{E}_{a\sim\pi_\theta}\big[-\alpha(\log \pi_\theta(a \mid s) + \mathcal{H}_{\text{target}})\big], \tag{4}$$

自动调整熵水平，平衡探索与利用。

3 应用与技巧

- **连续控制：**广泛用于机器人、自动驾驶等场景。
- **采样效率：**结合经验回放的离策略更新极具效率。
- **策略鲁棒性：**最大熵目标鼓励多样化动作。
- **实用建议：**规范化观测、保持双 Q 网络、使用对数参数化的 α、监控熵与 critic 损失。

4 Python 实战

脚本 `gen_sac_figures.py` 在一维连续控制任务中实现简化版 SAC，记录回报曲线与温度参数的变化趋势。

Listing 1: 脚本 $\text{gen}_s ac_f igures.py$

```
q_target = reward + gamma * (min_q_target - alpha * log_prob_next)
critic_w1 += critic_lr * (q_target - q1) * features(state, action)
critic_w2 += critic_lr * (q_target - q2) * features(state, action)

alpha_grad = -np.mean(log_prob + target_entropy)
log_alpha += alpha_lr * alpha_grad
alpha = np.exp(log_alpha)
```

5 实验结果

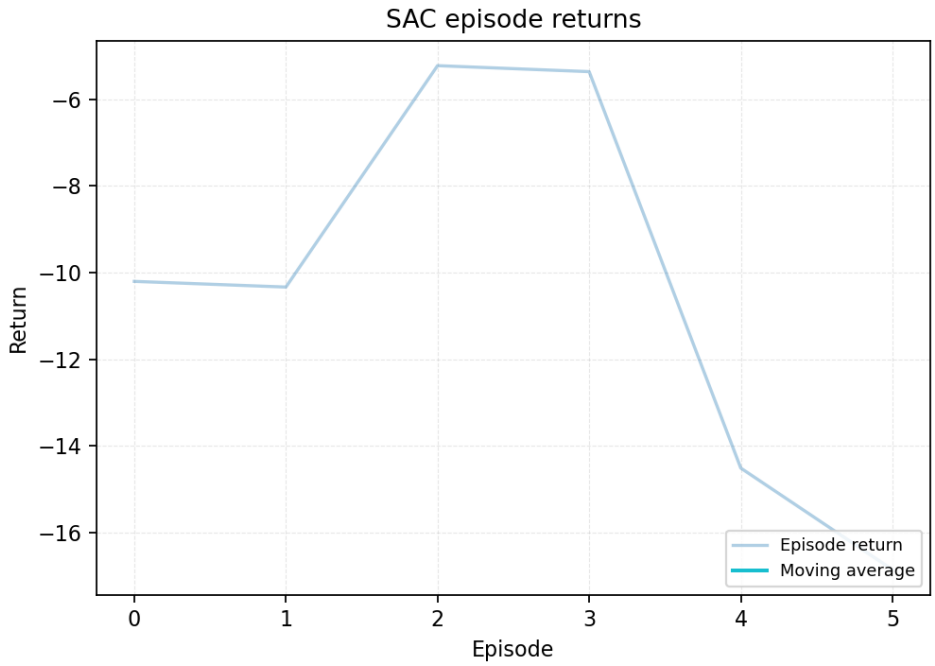

图 1: SAC 训练回报的稳步提升

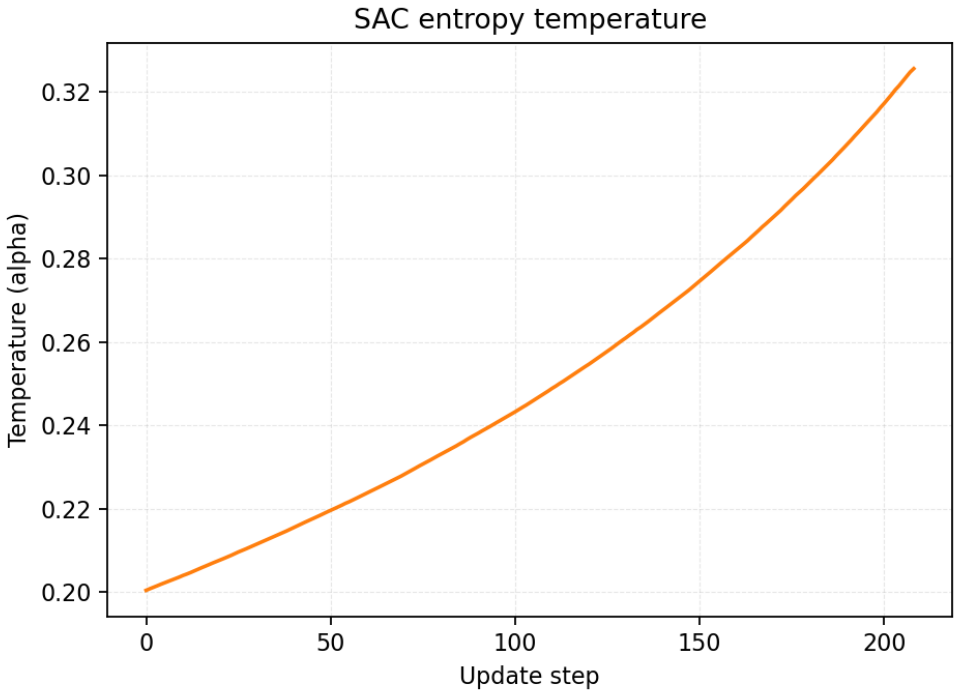

图 2: 温度参数收敛至目标熵附近

6 总结

SAC 通过双 Q 网络、随机策略与温度自适应实现稳定高效的连续控制学习。恰当的归一化、经验管理与熵监控可显著提升性能。示例展示了回报提升及温度动态符合预期。