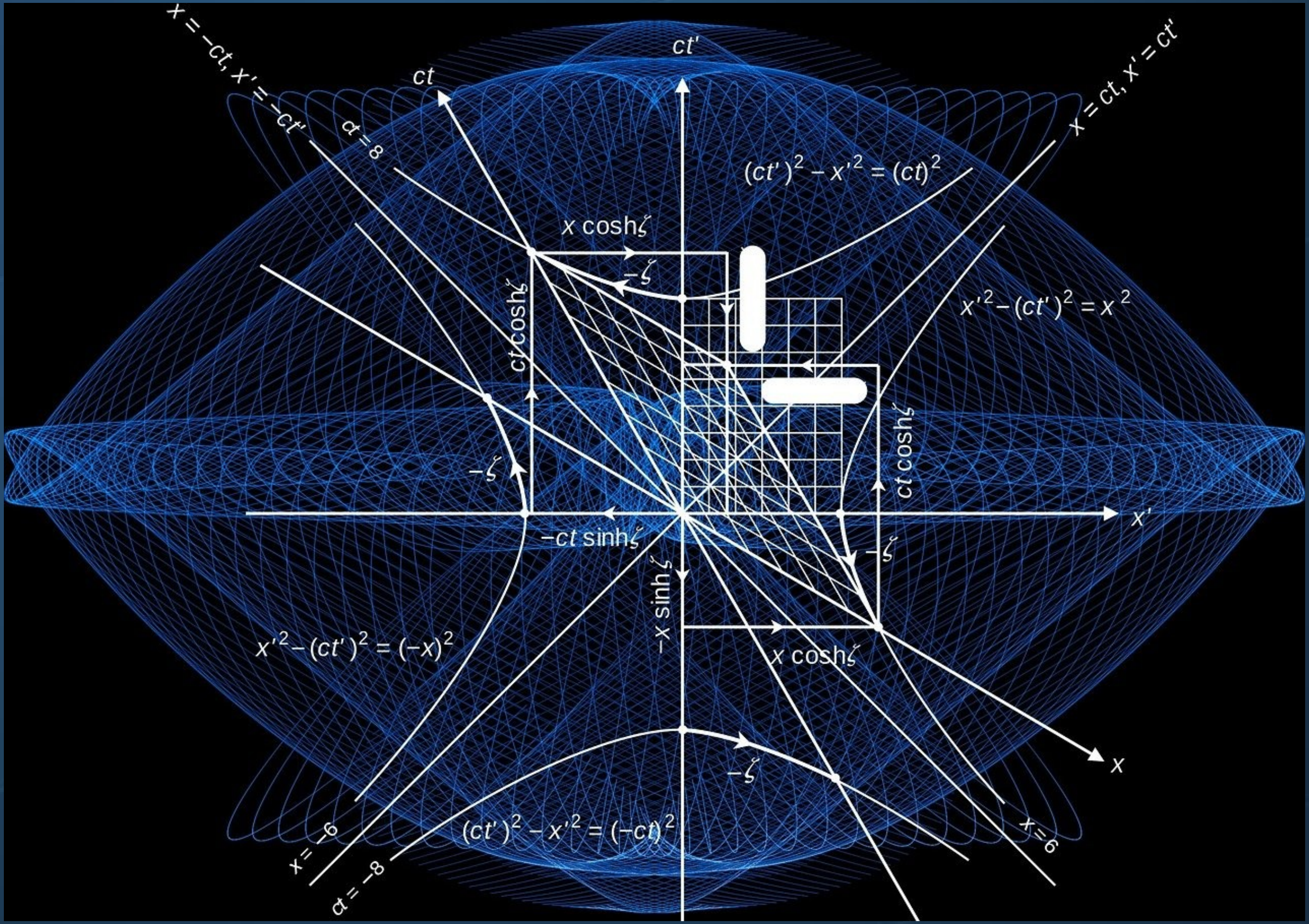
Soft Actor Critic Crash Course

Fundamental Concepts &
Implementation Notes

A Quick Introduction to SAC



How to use a maximum entropy framework in actor critic?

Drawbacks of AC Methods

- Brittle convergence
- High sample complexity
- These limit real world applicability

A Quick Introduction to SAC

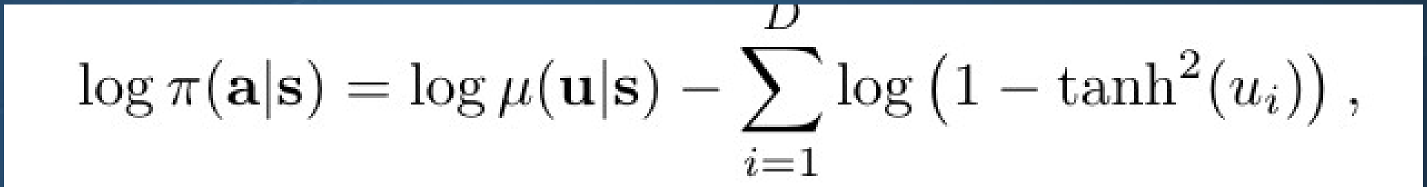
- Maximizes both long term rewards and entropy
- Similar to Q learning (epsilon greedy)
- Entropy modeled by reward scaling (inv. relationship)
- Leverages actor, value network, and critic networks
- Actually uses two critics like double Q learning/TD3
- Also makes use of a target value function (soft update)

A Quick Introduction to SAC

- Actor network models mean and sigma of distribution
- Original paper uses “reparameterization trick”
    - We won’t do this in this tutorial
- Use a special function to enforce action bounds
- Can (but won’t) use multiple steps of gradient descent

Implementation Notes

- Going to have a replay buffer based on numpy arrays



$$\log \pi(\mathbf{a}|\mathbf{s}) = \log \mu(\mathbf{u}|\mathbf{s}) - \sum_{i=1}^{D} \log \left(1 - \tanh^2(u_i)\right) ,$$

μ Sample of distribution with mean and sigma given by neural network

π Probability of selecting some action (continuous) given some state

- Also multiply by max action from env.

Actor Network Update

$$Cost = \frac{1}{N} \sum \left(\log \pi \left(a_t | s_t \right) - Q_{min} \left(s_t | a_t \right) \right)$$

Not sampled from buffer

Sampled from buffer

- Sample states from buffer but compute new actions
- Need the minimum value of the two critics
- The log is computed according to the previous slide

Value Network Update

$$Cost=\frac{1}{N}\sum\frac{1}{2}\left(V(S_t)-Q_{min}(s_t,a_t)-\log\pi(a_t|s_t)\right)^2$$

- Need value function (current params) for states
- Sample states from buffer but compute new actions
- Need the minimum value of the two critics
- The log is computed according to the previous slide

Target Value Network Update

$$\hat{\psi} \leftarrow \tau \psi + (1 - \tau) \hat{\psi}$$

- Tau is small, like 0.005
- Slowly moving average of online and target nets

Critic Network Update

$$Cost_1 = \frac{1}{N} \sum \frac{1}{2} \left(Q_1(s_t, a_t) - \hat{Q}(s_t | a_t) \right)^2$$

$$Cost_2 = \frac{1}{N} \sum \frac{1}{2} \left(Q_2(s_t, a_t) - \hat{Q}(s_t | a_t) \right)^2$$

$$\hat{Q} = r_{scaled} + \gamma \hat{V}(s_{t+1})$$

- Need target value function for new states
- Sample states and actions from buffer
- Our reward is scaled here!

Data Structures We Will Need

- Class for replay buffer → numpy arrays
- Class for actor network, critic network, value network
- Class for agent (ties everything together)
- Main loop to train and evaluate

Packages We Will Need

- Tensorflow-gpu, pybullet, gym, numpy, tensorflow-probability