**Soft-Actor Critic:** [**https://www.youtube.com/watch?v=HaZiurNC3Fc&t=3240s**](https://www.youtube.com/watch?v=HaZiurNC3Fc&t=3240s)

**Useful material:**

[**https://spinningup.openai.com/en/latest/algorithms/sac.html**](https://spinningup.openai.com/en/latest/algorithms/sac.html)

**SAC implementation using tensorflow or pytorch:**

[**https://github.com/openai/spinningup/blob/master/spinup/algos/tf1/sac/sac.py**](https://github.com/openai/spinningup/blob/master/spinup/algos/tf1/sac/sac.py)

[**https://github.com/openai/spinningup/blob/master/spinup/algos/pytorch/sac/sac.py**](https://github.com/openai/spinningup/blob/master/spinup/algos/pytorch/sac/sac.py)

**[Argomenti precedenti utili per ripassare]**

**Policy gradient methods:** [**https://www.youtube.com/watch?v=xAtGobAkjMI**](https://www.youtube.com/watch?v=xAtGobAkjMI)

**Slides:** [**https://classroom.google.com/u/0/c/MTYzMDQzMDIwMzcy/m/MTYzMDQzMDIwNDcy/details**](https://classroom.google.com/u/0/c/MTYzMDQzMDIwMzcy/m/MTYzMDQzMDIwNDcy/details)

• Learn a parameterized policy πθ(a|s) and select actions without consulting a value function.

• Value function may still be used to learn policy parameters: it is not required for action selection and can be learned as well using approximation, as (s, w)

Attempt to maximize performance through gradient ascent θt+1 = θt + α∇

(…Slides 14-37 to deepen the topic of policy gradient methods…)

Methods that learn approximations to both policy and value functions are a subset of policy-gradient methods and called **actor-critic methods**.

1:03:35 to 1:10:00 on YouTube for actor-critic methods explanation + slides 38-41.

**Approccio Actor-Critic:**

L’idea fondamentale di questo metodo è di separare il modello in due parti:

• L’**Actor** che aggiorna i parametri θ della policy πθ(a|s) e perciò decide quale azione intraprendere. Essenzialmente il suo obiettivo è di imparare una policy ottima per far muovere l’agente. È evidente come questa parte sia ispirata ai metodi policy-based.

• Il **Critic** giudica le azioni dell’actor, aggiornando costantemente i parametri w della value function, la quale può essere o del tipo action-value qw(a|s) o del tipo state value Vw(s).

Grazie alle indicazioni del critico l’actor potrà quindi aggiustare la policy in modo molto più efficace rispetto ai metodi tradizionali che non fanno uso di questo accorgimento. L’idea che sta alla base di questo approccio è davvero intuitiva, basandosi sul fatto che abbiamo due componenti separata (ma interagenti) che miglioreranno in modo incrementale la loro capacità, rispettivamente, di decidere cosa fare e di valutare quanto vantaggiosa è stata la scelta intrapresa.