Overview

- Stochastic Gradient Descent
Features for Linear Methods

Polynomial

Example: $x(s) = [1, s_1, s_2, s_1 s_2, s_1^2, s_2^2, s_1 s_2^2, s_1^2 s_2, s_1^2, s_2^2]$
Cartpole problem

Actual Cartpole video  Computer Cartpole Video

State:

- Cart Position \([-2.4, 2.4]\)
- Cart velocity: \(\mathbb{R}\)
- Pole Angle: \([-41.8^\circ, 41.8^\circ]\)
- Pole tip velocity: \(\mathbb{R}\)

Action:

- Left
- Right
Cartpole problem

Reward: 1 for every step taken, including termination step

Episode Termination:

- Pole angle more than $±12°$
- Cart position more than $±24$
- Episode length $≥ 500$
OpenAI Gym

Gym is a toolkit for developing and comparing reinforcement learning algorithms

```python
import gym
env = gym.make("CartPole-v1")
observation = env.reset()
for _ in range(1000):
    env.render()
    action = env.action_space.sample()  # your agent here (this takes random actions)
    observation, reward, done, info = env.step(action)
    if done:
        observation = env.reset()
env.close()
```