1-D random walk ↴ gambler's ruin

Normal greedy → Only exploitation; increase cumulative reward

ε-greedy approach
↓
Consider both exploration and exploitation
↓
choose greedy action with 1-ε AND random action with ε

ADAM → RMS prop + momentum
ADAM + NAG

Nesterov Accelerated Gradients (NAG)
↓
Smoothing convex functions
↓
convergence rates → $1/t^2$
↓
Newton method → $1/t^3$
↓
Normal SGD → $1/t$

Number of states/actions are large
↓
Neural Networks

Define DQN cost function

$$\text{Cost} = \left[Q(s,a;\theta) - \left(r(s,a) + \gamma \max_a Q(s',a;\theta) \right) \right]^2$$

→ Policy Improvement methods
→ Temporal Difference methods

Tikhonov regularization / Ridge Rigression
↓
ill-posed problems
↓
Co-Colinearity

Deep Q-Networks
↓ Off-Model Learning

Markov-states assumption for RL
↓ Fails sometimes (lose data)

Bellman Equation

$$Q(s,a) = r(s,a) + \gamma \max_a Q(s',a)$$

Q(s,a): Quality value
r(s,a): current reward
γ: discount factor — describes importance of immediate reward over future reward
$\max_a Q(s',a)$: future reward.

→ Even if some values are wrong, we will converge towards the correct value in the long run

Gradient Boosting - Ensemble } strong classifier from weak classifiers - decision stumps
↓
Decision Trees
| classification Trees
| Regression Trees