Experiment design

Bandit problems and Markov decision processes

Christos Dimitrakakis

UiO

November 13, 2019

Sequential problems: full observation

Example 1

- n meteorological stations $\{\mu_i \mid i = 1, \ldots, n\}$
- The i-th station gives a rain probability $x_{t,i} = P_{\mu_i}(y_t \mid y_1, \ldots, y_{t-1})$.

- Observation $\boldsymbol{x}_t = (x_{t,1}, \ldots, x_{t,n})$: the predictions of all stations.
- Decision a_t: Guess if it will rain
- Outcome y_t: Rain or not rain.
- Steps $t = 1, \ldots, T$.

Linear utility function

Reward function is $\rho(y_t, a_t) = \mathbb{I}\{y_t = a_t\}$ simply rewarding correct predictions with utility being

$$U(y_1, y_2, \ldots, y_T, a_1, \ldots, a_T) = \sum_{t=1}^{T} \rho(y_t, a_t),$$

the total number of correct predictions.

The n meteorologists problem is simple, as:

- You always see their predictions, as well as the weather, no matter whether you bike or take the tram (full information)
- Your actions do not influence their predictions (independence events)

In the remainder, we'll see two settings where decisions are made with either partial information or in a dynamical system. Both of these settings can be formalised with Markov decision processes.

Experimental design and Markov decision processes

The following problems

- Shortest path problems.
- Optimal stopping problems.
- Reinforcement learning problems.
- Experiment design (clinical trial) problems
- Advertising.

can be all formalised as Markov decision processes.

Applications

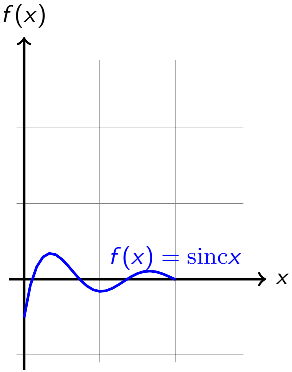
- Robotics.
- Economics.
- Automatic control.
- Resource allocation

Bandit problems

Bandit problems

Applications

- Efficient optimisation.



Bandit problems

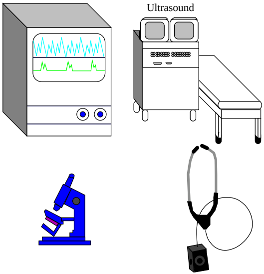
Applications

- Efficient optimisation.
- Online advertising.

Bandit problems

Applications

- Efficient optimisation.
- Online advertising.
- Clinical trials.



Bandit problems

Applications

- Efficient optimisation.
- Online advertising.
- Clinical trials.
- ROBOT SCIENTIST.

The stochastic n-armed bandit problem

Actions and rewards

- A set of actions $\mathcal{A} = \{1, \ldots, n\}$.
- Each action gives you a random reward with distribution $\mathbb{P}(r_t \mid a_t = i)$.
- The expected reward of the i-th arm is $\rho_i \triangleq \mathbb{E}(r_t \mid a_t = i)$.

Interaction at time t

1. You choose an action $a_t \in \mathcal{A}$.
2. You observe a random reward r_t drawn from the i-th arm.

The utility is the sum of the rewards obtained

$$U \triangleq \sum_t r_t.$$

We must maximise the expected utility, without knowing the values ρ_i.

Policy

Definition 2 (Policies)

A policy π is an algorithm for taking actions given the observed history $h_t \triangleq a_1, r_1, \ldots, a_t, r_t$

$$\mathbb{P}^\pi(a_{t+1} \mid h_t)$$

is the probability of the next action a_{t+1}.

Exercise 1

Why should our action depend on the complete history?

- A The next reward depends on all the actions we have taken.
- B We don't know which arm gives the highest reward.
- C The next reward depends on all the previous rewards.
- D The next reward depends on the complete history.
- E No idea.

Policy

Definition 2 (Policies)

A policy π is an algorithm for taking actions given the observed history $h_t \triangleq a_1, r_1, \dots, a_t, r_t$

$$\mathbb{P}^\pi(a_{t+1} \mid h_t)$$

is the probability of the next action a_{t+1}.

Example 3 (The expected utility of a uniformly random policy)

If $\mathbb{P}^\pi(a_{t+1} \mid \cdot) = 1/n$ for all t, then

Policy

Definition 2 (Policies)

A policy π is an algorithm for taking actions given the observed history $h_t \triangleq a_1, r_1, \ldots, a_t, r_t$

$$\mathbb{P}^\pi(a_{t+1} \mid h_t)$$

is the probability of the next action a_{t+1}.

Example 3 (The expected utility of a uniformly random policy)

If $\mathbb{P}^\pi(a_{t+1} \mid \cdot) = 1/n$ for all t, then

$$\mathbb{E}^\pi U = \mathbb{E}^\pi \left(\sum_{t=1}^{T} r_t \right) = \sum_{t=1}^{T} \mathbb{E}^\pi r_t = \sum_{t=1}^{T} \sum_{i=1}^{n} \frac{1}{n} \rho_i = \frac{T}{n} \sum_{i=1}^{n} \rho_i$$

Policy

Definition 2 (Policies)

A policy π is an algorithm for taking actions given the observed history $h_t \triangleq a_1, r_1, \ldots, a_t, r_t$

$$\mathbb{P}^\pi(a_{t+1} \mid h_t)$$

is the probability of the next action a_{t+1}.

The expected utility of a general policy

$$\mathbb{E}^\pi U = \mathbb{E}^\pi \left(\sum_{t=1}^T r_t \right)$$

Policy

Definition 2 (Policies)

A policy π is an algorithm for taking actions given the observed history $h_t \triangleq a_1, r_1, \ldots, a_t, r_t$

$$\mathbb{P}^\pi(a_{t+1} \mid h_t)$$

is the probability of the next action a_{t+1}.

The expected utility of a general policy

$$\mathbb{E}^\pi U = \mathbb{E}^\pi \left(\sum_{t=1}^T r_t \right) = \sum_{t=1}^T \mathbb{E}^\pi(r_t) \tag{1.1}$$

Policy

Definition 2 (Policies)

A policy π is an algorithm for taking actions given the observed history $h_t \triangleq a_1, r_1, \ldots, a_t, r_t$

$$\mathbb{P}^\pi(a_{t+1} \mid h_t)$$

is the probability of the next action a_{t+1}.

The expected utility of a general policy

$$\begin{aligned}\mathbb{E}^\pi U = \mathbb{E}^\pi \left(\sum_{t=1}^{T} r_t \right) &= \sum_{t=1}^{T} \mathbb{E}^\pi(r_t) \\ &= \sum_{t=1}^{T} \sum_{a_t \in \mathcal{A}} \mathbb{E}(r_t \mid a_t) \sum_{h_{t-1}} \mathbb{P}^\pi(a_t \mid h_{t-1}) \, \mathbb{P}^\pi(h_{t-1})\end{aligned} \tag{1.1}$$

A simple heuristic for the unknown reward case

Say you keep a running average of the reward obtained by each arm

$$\hat{\theta}_{t,i} = R_{t,i}/n_{t,i}$$

- $n_{t,i}$ the number of times you played arm i
- $R_{t,i}$ the total reward received from i.

Whenever you play $a_t = i$:

$$R_{t+1,i} = R_{t,i} + r_t, \qquad n_{t+1,i} = n_{t,i} + 1.$$

Greedy policy:

$$a_t = \arg\max_i \hat{\theta}_{t,i}.$$

What should the initial values $n_{0,i}, R_{0,i}$ be?

Bernoulli bandits

Decision-theoretic approach

- Assume $r_t \mid a_t = i \sim P_{\theta_i}$, with $\theta_i \in \Theta$.
- Define prior belief ξ_1 on Θ.
- For each step t, find a policy π selecting action $a_t \mid \xi_t \sim \pi(a \mid \xi_t)$ to

$$\max_\pi \mathbb{E}^\pi_{\xi_t}(U_t) = \max_\pi \mathbb{E}^\pi_{\xi_t} \sum_{a_t} \left(\sum_{k=1}^{T-t} r_{t+k} \;\middle|\; a_t \right) \pi(a_t \mid \xi_t).$$

- Obtain reward r_t.
- Calculate the next belief

$$\xi_{t+1} = \xi_t(\cdot \mid a_t, r_t)$$

How can we implement this?

Bayesian inference on Bernoulli bandits

- Likelihood: $\mathbb{P}_\theta(r_t = 1) = \theta$.
- Prior: $\xi(\theta) \propto \theta^{\alpha-1}(1-\theta)^{\beta-1}$ (i.e. $\mathcal{B}eta(\alpha, \beta)$).

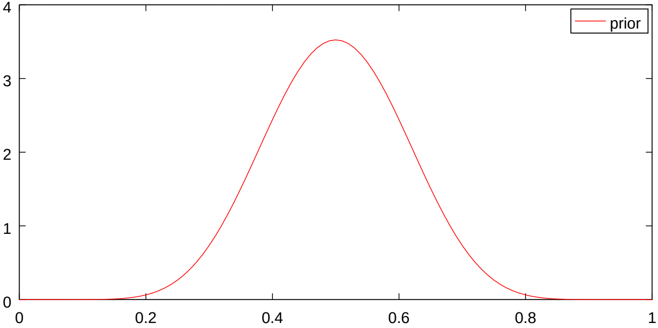


Figure: Prior belief ξ about the mean reward θ.

Bayesian inference on Bernoulli bandits

For a sequence $r = r_1, \ldots, r_n$, $\Rightarrow P_\theta(r) \propto \theta_i^{\#1(\mathrm{r})}(1-\theta_i)^{\#0(\mathrm{r})}$

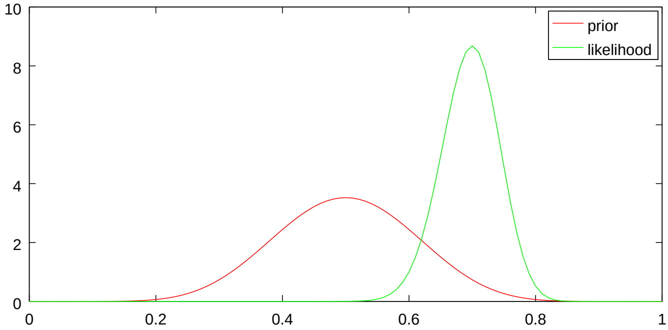


Figure: Prior belief ξ about θ and likelihood of θ for 100 plays with 70 1s.

Bayesian inference on Bernoulli bandits

Posterior: $\mathcal{B}eta(\alpha + \#1(\mathrm{r}), \beta + \#0(\mathrm{r}))$.

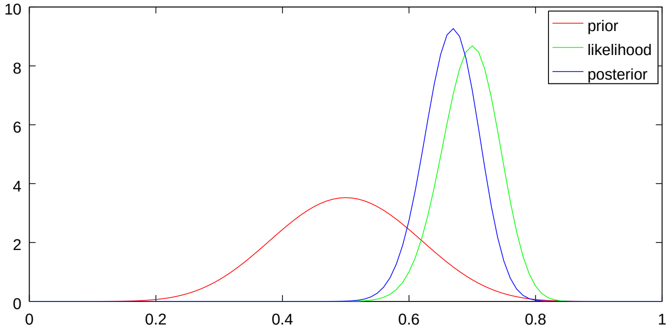


Figure: Prior belief $\xi(\theta)$ about θ, likelihood of θ for the data r, and posterior belief $\xi(\theta \mid r)$

Bernoulli example.

Consider n Bernoulli distributions with unknown parameters θ_i ($i = 1, \dots, n$) such that

$$r_t \mid a_t = i \sim \mathcal{Bernoulli}(\theta_i), \qquad \mathbb{E}(r_t \mid a_t = i) = \theta_i. \tag{1.2}$$

Our belief for each parameter θ_i is $\mathcal{Beta}(\alpha_i, \beta_i)$, with density $f(\theta \mid \alpha_i, \beta_i)$ so that

$$\xi(\theta_1, \dots, \theta_n) = \prod_{i=1}^{n} f(\theta_i \mid \alpha_i, \beta_i). \qquad \text{(a priori independent)}$$

$$N_{t,i} \triangleq \sum_{k=1}^{t} \mathbb{I}\left\{a_k = i\right\}, \qquad \hat{r}_{t,i} \triangleq \frac{1}{N_{t,i}} \sum_{k=1}^{t} r_t \, \mathbb{I}\left\{a_k = i\right\}$$

Then, the posterior distribution for the parameter of arm i is

$$\xi_t = \mathcal{Beta}(\alpha_i^t, \beta_i^t), \qquad \alpha_i^t = \alpha_i + N_{t,i}\hat{r}_{t,i}\ , \ \beta_i^t = \beta_i N_{t,i}(1 - \hat{r}_{t,i})).$$

Since $r_t \in \{0, 1\}$ there are $O((2n)^T)$ possible belief states for a T-step bandit problem.

Belief states

- The state of the decision-theoretic bandit problem is the state of our belief.
- A sufficient statistic is the number of plays and total rewards.
- Our belief state ξ_t is described by the priors α, β and the vectors

$$N_t = (N_{t,1}, \ldots, N_{t,i}) \tag{1.3}$$
$$\hat{r}_t = (\hat{r}_{t,1}, \ldots, \hat{r}_{t,i}). \tag{1.4}$$

- The next-state probabilities are defined as:

$$\mathbb{P}_{\xi_t}(r_t = 1 \mid a_t = i) = \frac{\alpha_i^t}{\alpha_i^t + \beta_i^t}$$

  as ξ_{t+1} is a deterministic function of ξ_t, r_t and a_t
- Optimising this results in a Markov decision process.

Markov process

$$s_{t-1} \to s_t \to s_{t+1}$$

Definition 3 (Markov Process – or Markov Chain)

The sequence $\{s_t \mid t = 1, \ldots\}$ of random variables $s_t : \Theta \to \mathcal{S}$ is a Markov process if

$$\mathbb{P}(s_{t+1} \mid s_t, \ldots, s_1) = \mathbb{P}(s_{t+1} \mid s_t). \tag{1.5}$$

- s_t is state of the Markov process at time t.
- $\mathbb{P}(s_{t+1} \mid s_t)$ is the transition kernel of the process.

The state of an algorithm

Observe that the α, β form a Markov process. They also summarise our belief about which arm is the best.

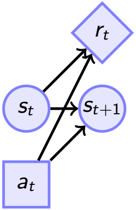
Markov decision processes

In a Markov decision process (MDP), the state s includes all the information we need to make predictions.

Markov decision processes (MDP).

At each time step t:

- We observe state $s_t \in \mathcal{S}$.
- We take action $a_t \in \mathcal{A}$.
- We receive a reward $r_t \in \mathbb{R}$.

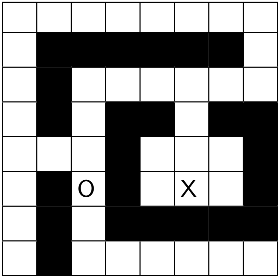


Markov property of the reward and state distribution

$$\mathbb{P}_\mu(s_{t+1} \mid s_t, a_t) \qquad \text{(Transition distribution)}$$

$$\mathbb{P}_\mu(r_t \mid s_t, a_t) \qquad \text{(Reward distribution)}$$

Stochastic shortest path problem with a pit



Properties

- $T \to \infty$.
- $r_t = -1$, but $r_t = 0$ at X and -100 at O and the problem ends.
- $\mathbb{P}_\mu(s_{t+1} = X | s_t = X) = 1$.
- $\mathcal{A} = \{\mathrm{North}, \mathrm{South}, \mathrm{East}, \mathrm{West}\}$
- Moves to a random direction with probability ω. Walls block.

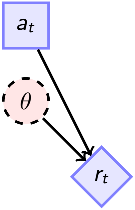


Figure: The basic bandit MDP. The decision maker selects a_t, while the parameter θ of the process is hidden. It then obtains reward r_t. The process repeats for $t = 1, \ldots, T$.

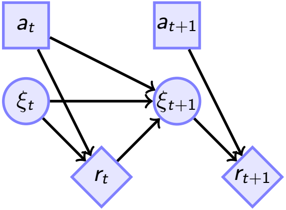


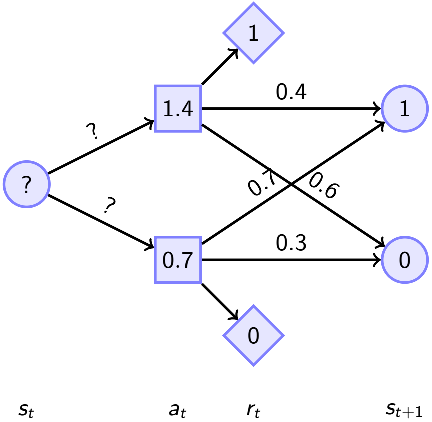
Figure: The decision-theoretic bandit MDP. While θ is not known, at each time step t we maintain a belief ξ_t on Θ. The reward distribution is then defined through our belief.

Backwards induction (Dynamic programming)

for $n = 1, 2, \ldots$ and $s \in \mathcal{S}$ **do**

$$\mathbb{E}(U_t \mid \xi_t) = \max_{a_t \in \mathcal{A}} \mathbb{E}(r_t \mid \xi_t, a_t) + \sum_{\xi_{t+1}} \mathbb{P}(\xi_{t+1} \mid \xi_t, a_t)\, \mathbb{E}(U_{t+1} \mid \xi_{t+1})$$

end for



Exercise 1

What is the value $v_t(s_t)$ of the first state?

A 1.4

B 1.05

C 1.0

D 0.7

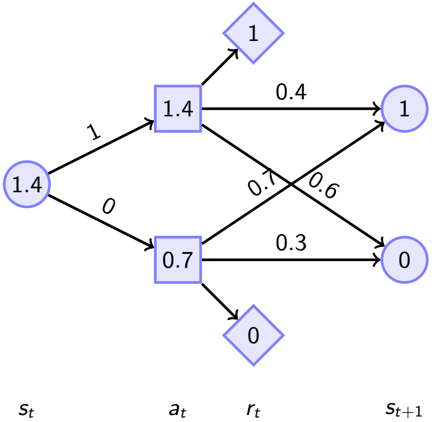
E 0

Backwards induction (Dynamic programming)

for $n = 1, 2, \ldots$ and $s \in \mathcal{S}$ **do**

$$\mathbb{E}(U_t \mid \xi_t) = \max_{a_t \in \mathcal{A}} \mathbb{E}(r_t \mid \xi_t, a_t) + \sum_{\xi_{t+1}} \mathbb{P}(\xi_{t+1} \mid \xi_t, a_t)\, \mathbb{E}(U_{t+1} \mid \xi_{t+1})$$

end for



Exercise 1

What is the value $v_t(s_t)$ of the first state?

A 1.4

B 1.05

C 1.0

D 0.7

E 0

Heuristic algorithms for the n-armed bandit problem

Algorithm 1 UCB1

Input $\mathcal{A}$

$\hat{\theta}_{0,i} = 1, \forall i$

for $t = 1, \ldots$ **do**

$a_t = \arg\max_{i \in \mathcal{A}} \left\{ \hat{\theta}_{t-1,i} + \sqrt{\frac{2 \ln t}{N_{t-1,i}}} \right\}$

$r_t \sim P_\theta(r \mid a_t)$ `// play action and get reward // update model`

$N_{t,a_t} = N_{t-1,a_t} + 1$

$\hat{\theta}_{t,a_t} = [N_{t-1,a_t} \theta_{t-1,a_t} + r_t]/N_{t,a_t}$

$\forall i \neq a_t,\ N_{t,i} = N_{t-1,i},\ \hat{\theta}_{t,i} = \hat{\theta}_{t-1,i}$

end for

Algorithm 2 Thompson sampling

Input $\mathcal{A}, \xi_0$

for $t = 1, \ldots$ **do**

$\hat{\theta} \sim \xi_{t-1}(\theta)$

$a_t \in \arg\max_a \mathbb{E}_{\hat{\theta}}[r_t \mid a_t = a]$.

$r_t \sim P_\theta(r \mid a_t)$ `// play action and get reward // update model`

$\xi_t(\theta) = \xi_{t-1}(\theta \mid a_t, r_t)$.

end for

Example 4 (Clinical trials)

Consider an example where we have some information x_t about an individual patient t, and we wish to administer a treatment a_t. For whichever treatment we administer, we can observe an outcome y_t. Our goal is to maximise expected utility.

Definition 5 (The contextual bandit problem.)

At time t,

- We observe $x_t \in \mathcal{X}$.
- We play $a_t \in \mathcal{A}$.
- We obtain $r_t \in \mathbb{R}$ with $r_t \mid a_t = a, x_t = x \sim P_\theta(r \mid a, x)$.

Example 6 (The linear bandit problem)

- $\mathcal{A} = [n]$, $\mathcal{X} = \mathbb{R}^k$, $\theta = (\theta_1, \ldots, \theta_n)$, $\theta_i \in \mathbb{R}^k$, $r \in \mathbb{R}$.
- $r \sim \mathcal{N}(\theta_a^\top x), 1)$

Example 7 (A clinical trial example)

- $\mathcal{A} = [n]$, $\mathcal{X} = \mathbb{R}^k$, $\theta = (\theta_1, \ldots, \theta_n)$, $\theta_i \in \mathbb{R}^k$, $y \in \{0, 1\}$.
- $y \sim \mathcal{Bernoulli}(1/(1 + exp[-(\theta_a^\top x)^2])$.
- $r = U(a, y)$.

Example 8 (One-stage problems)

- Initial belief ξ_0
- Side information $\boldsymbol{x}$
- Simultaneously takes actions $\boldsymbol{a}$.
- Observes outcomes $\boldsymbol{y}$.

$$\mathbb{E}^{\pi}_{\xi_0}(U \mid \boldsymbol{x}) = \sum_{\boldsymbol{x},\boldsymbol{y}} \mathbb{P}_{\xi_0}(\boldsymbol{y} \mid \boldsymbol{a}, \boldsymbol{x}) \pi(\boldsymbol{a} \mid \boldsymbol{x}) \underbrace{\mathbb{E}^{\pi}_{\xi_0}(U \mid \boldsymbol{x}, \boldsymbol{a}, \boldsymbol{y})}_{\text{post-hoc value}} \tag{4.1}$$

Example 8 (One-stage problems)

- Initial belief ξ_0
- Side information $\boldsymbol{x}$
- Simultaneously takes actions $\boldsymbol{a}$.
- Observes outcomes $\boldsymbol{y}$.

Definition 9 (Expected information gain)

$$\mathbb{E}^{\pi}_{\xi_0}\left(\mathbb{D}\left(\xi_1 \| \xi_0\right) \mid \boldsymbol{x}\right) = \sum_{\boldsymbol{x},\boldsymbol{y}} \mathbb{P}_{\xi_0}(\boldsymbol{y} \mid \boldsymbol{a}, \boldsymbol{x}) \pi(\boldsymbol{a} \mid \boldsymbol{x}) \mathbb{D}\left(\xi_0(\cdot \mid \boldsymbol{x}, \boldsymbol{a}, \boldsymbol{y}) \| \xi_0\right) \tag{4.1}$$

Example 8 (One-stage problems)

- Initial belief ξ_0
- Side information $\boldsymbol{x}$
- Simultaneously takes actions $\boldsymbol{a}$.
- Observes outcomes $\boldsymbol{y}$.

Definition 9 (Expected utility of final policy)

$$\mathbb{E}^{\pi}_{\xi_0}\left(\max_{\pi_1} \mathbb{E}^{\pi_1}_{\xi_1} \rho \middle| \boldsymbol{x}\right) = \sum_{\boldsymbol{x},\boldsymbol{y}} \mathbb{P}_{\xi_0}(\boldsymbol{y} \mid \boldsymbol{a}, \boldsymbol{x}) \pi(\boldsymbol{a} \mid \boldsymbol{x}) \max_{\pi_1} \mathbb{E}^{\pi_1}_{\xi_0}(\rho \mid \boldsymbol{a}, \boldsymbol{x}, \boldsymbol{y}) \tag{4.1}$$

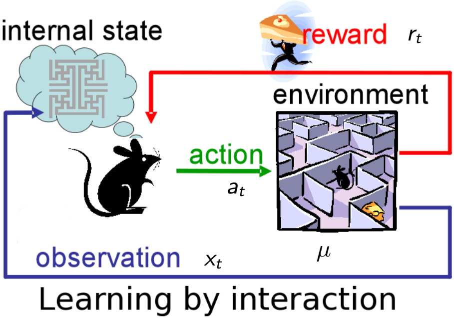
$$\mathbb{E}^{\pi_1}_{\xi_0}(\rho \mid \boldsymbol{a}, \boldsymbol{x}, \boldsymbol{y}) = \sum_{a,x,y} \rho(a, y) \, \mathbb{P}_{\xi_1}(y \mid x, a) \pi_1(a \mid x) \, \mathbb{P}_{\xi_1}(x) \tag{4.2}$$

Experiment design for a one-stage problem

- Select some model $\mathbb{P}$ for generating data.
- Select an inference and/or decision making algorithm λ for the task.
- Select a performance measure U.
- Generate data D from $\mathbb{P}$ and measure the performance of λ on D.

The reinforcement learning problem

Learning to act in an unknown world, by interaction and reinforcement.



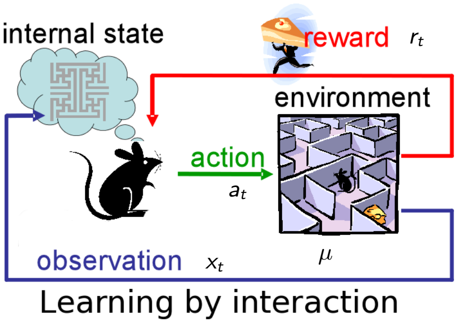
The reinforcement learning problem

Learning to act in an unknown world, by interaction and reinforcement.

Expected total reward

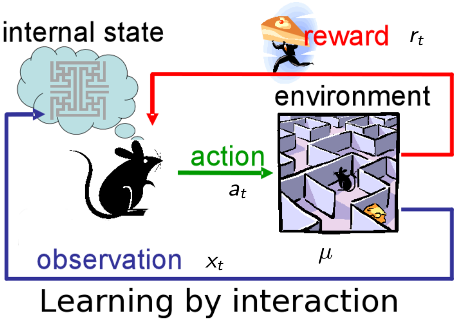
. . . when using policy π in μ:

$$U(\mu, \pi)$$



The reinforcement learning problem

Learning to act in an unknown world, by interaction and reinforcement.



Expected total reward

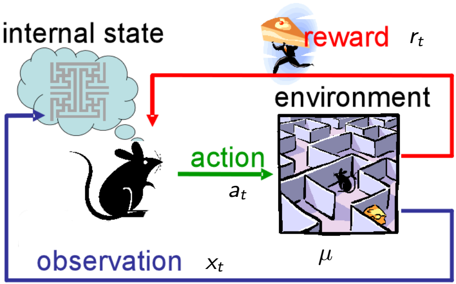
…when using policy π in μ:

$$U(\mu, \pi)$$

Can't we just $\max_\pi U(\mu, \pi)$?

The reinforcement learning problem

Learning to act in an unknown world, by interaction and reinforcement.



Learning by interaction

Expected total reward

…when using policy π in μ:

$$U(\mu, \pi)$$

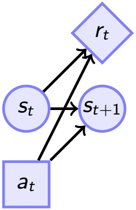
Knowing μ contradicts the problem definition

Solving a given MDP

Markov decision processes (MDP).

At each time step t:

- We observe state $s_t \in \mathcal{S}$.
- We take action $a_t \in \mathcal{A}$.
- We receive a reward $r_t \in \mathbb{R}$ with $r_t \sim P_\mu(r_t \mid s_t, a_t)$
- We go to the next state $s_{t+1} \in \mathcal{S}$ with $s_{t+1} \sim P_\mu(s_{t+1} \mid s_t, a_t)$



Backwards induction (Value iteration)

for $n = 1, 2, \ldots$ and $s \in \mathcal{S}$ **do**

$$\mathbb{E}_\mu^{\pi^*}(U_t \mid s_t) = \max_{a_t \in \mathcal{A}} \mathbb{E}_\mu(r_t \mid s_t, a_t) + \sum_{s_{t+1}} \mathbb{P}_\mu(s_{t+1} \mid s_t, a_t) \, \mathbb{E}_\mu^{\pi^*}(U_{t+1} \mid s_{t+1})$$

end for

The discounted setting

$$U_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k}, \qquad \gamma \in (0, 1)$$

Value functions

$$V^{\pi}_{\mu}(s) \triangleq \mathbb{E}(U_t \mid s_t = s), \qquad Q^{\pi}_{\mu}(s, a) \triangleq \mathbb{E}(U_t \mid s_t = s, a_t = a)$$

The discounted setting

$$U_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k}, \qquad \gamma \in (0, 1)$$

Value functions

$$V_\mu^\pi(s) \triangleq \mathbb{E}(U_t \mid s_t = s), \qquad Q_\mu^\pi(s, a) \triangleq \mathbb{E}(U_t \mid s_t = s, a_t = a)$$

Bellman equation

$$V_\mu^\pi(s) = \mathbb{E}_\mu^\pi(r_t \mid s_t = s) + \gamma \sum_{s_{t+1}} V_\mu^\pi(s_{t+1}) \, \mathbb{P}_\mu^\pi(s_{t+1} \mid s_t)$$

$$Q_\mu^\pi(s, a) = \mathbb{E}_\mu(r_t \mid s_t = s, a_t = a) + \gamma \sum_{s_{t+1}} Q_\mu^\pi(s_{t+1}, \pi(s_{t+1})) P_\mu(s_{t+1} \mid s_t, a_t = a)$$

The discounted setting

$$U_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k}, \qquad \gamma \in (0,1)$$

Value functions

$$V_\mu^\pi(s) \triangleq \mathbb{E}(U_t \mid s_t = s), \qquad Q_\mu^\pi(s,a) \triangleq \mathbb{E}(U_t \mid s_t = s, a_t = a)$$

Bellman equation

$$V_\mu^\pi(s) = \mathbb{E}_\mu^\pi(r_t \mid s_t = s) + \gamma \sum_{s_{t+1}} V_\mu^\pi(s_{t+1}) \, \mathbb{P}_\mu^\pi(s_{t+1} \mid s_t)$$

$$Q_\mu^\pi(s,a) = \mathbb{E}_\mu(r_t \mid s_t = s, a_t = a) + \gamma \sum_{s_{t+1}} Q_\mu^\pi(s_{t+1}, \pi(s_{t+1})) P_\mu(s_{t+1} \mid s_t, a_t = a)$$

Optimality condition

$$V_\mu^*(s) \geq V_\mu^\pi(s) \forall s$$

Q-learning and induction

Q-Value iteration

$$Q_{n+1}(s,a) = r(s,a) + \gamma \sum_{s_{t+1}} P_\mu(s_{t+1} \mid s_t, a_t = a) \max_{a'} Q_n(s_{t+1}, a')$$

Q-learning

$$\hat{R}_t = r_t + \gamma \max_{a'} \hat{Q}_t(s_{t+1}, a')$$

$$\hat{Q}_{t+1}(s,a) = (1-\alpha)\hat{Q}_n(s,a) + \alpha(\hat{R}_t)$$

Summary

Markov decision processes

- Formalise experiment design
- Formalise environments in reinforcement learning

Solving MDPs

- Discrete case: dynamic programming.
- General case: approximations, gradient methods, etc.

Reinforcement learning and experiment design

- Formal but intractable Bayesian solution.
- Convergent algorithms in simple settings.