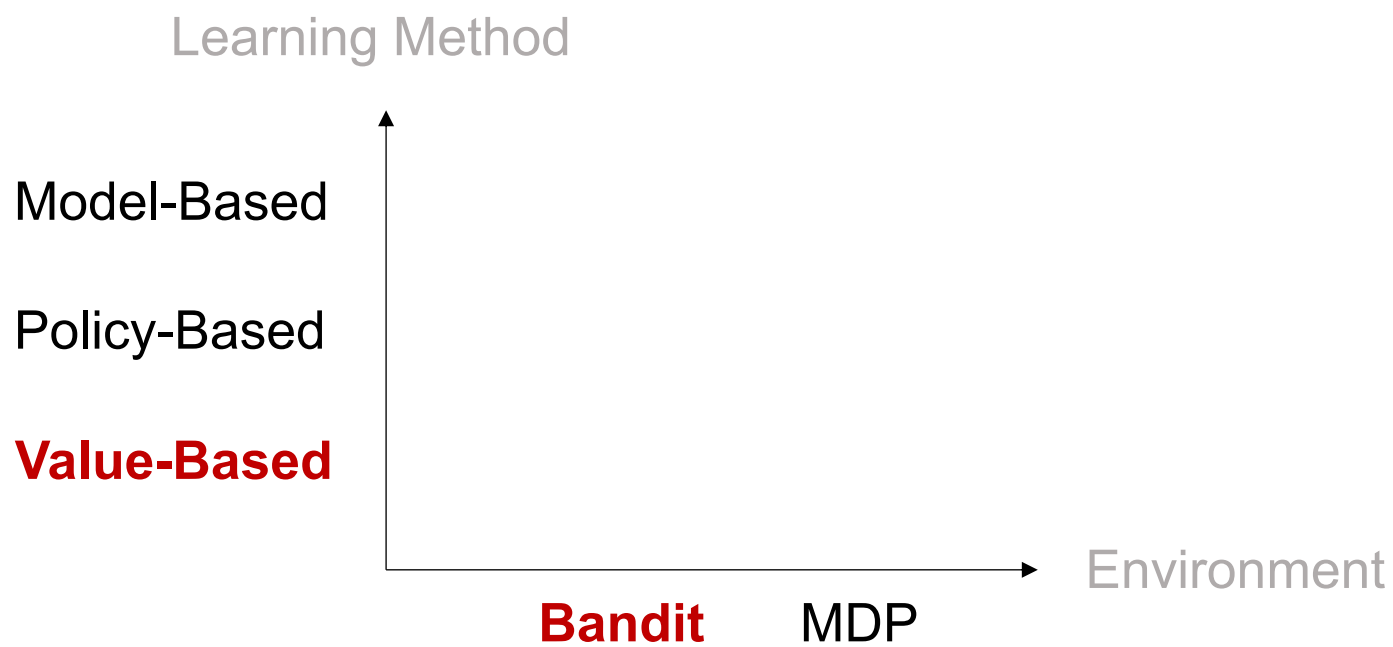
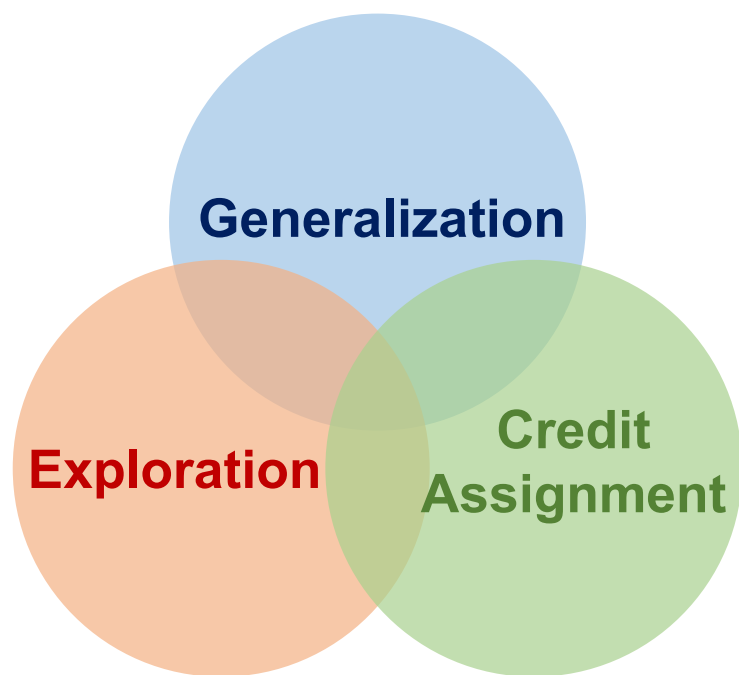


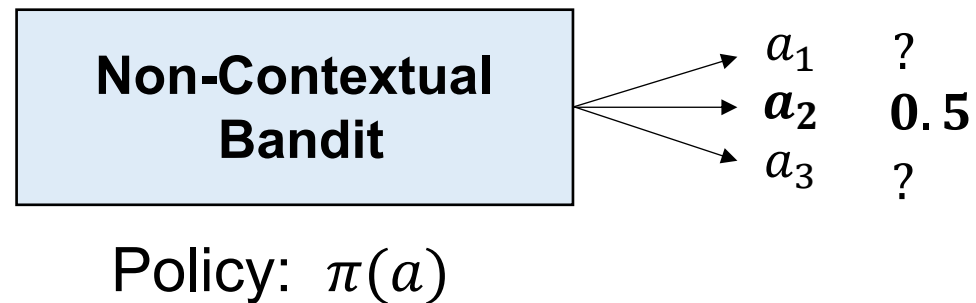
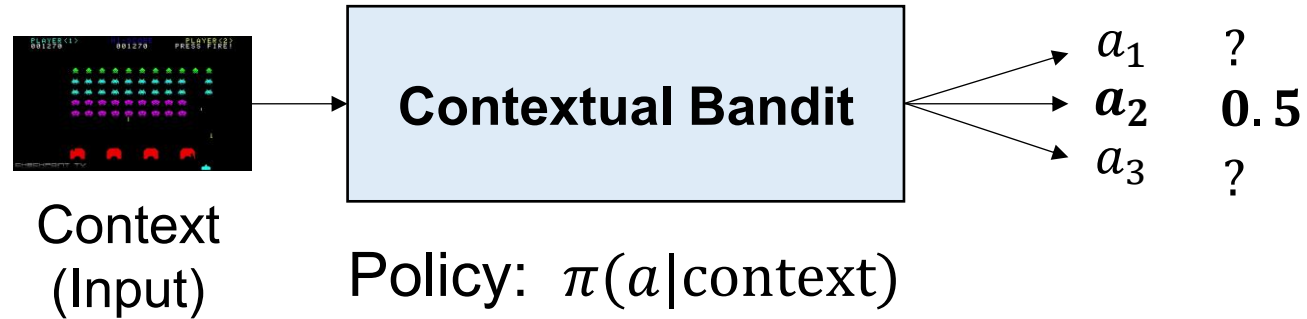
# **Value-Based Bandit Algorithms**

Chen-Yu Wei

# Roadmap



# Contextual Bandits and Non-Contextual Bandits



# **Multi-Armed Bandits**

Non-Contextual Bandits with Discrete Actions

# Multi-Armed Bandits



A slot machine

**One-armed bandit**



A row of slot machines

**Multi-armed bandit**

# Multi-Armed Bandits

**Given:** arm set  $\mathcal{A} = \{1, \dots, A\}$

For time  $t = 1, 2, \dots, T$ :

Learner chooses an arm  $a_t \in \mathcal{A}$

Learner observes  $r_t = R(a_t) + w_t$

**Arm = Action**

**Assumption:**  $R(a)$  is the (hidden) ground-truth reward function

$w_t$  is (zero-mean) noise

**Goal:** maximize the total reward  $\sum_{t=1}^T R(a_t)$

# How to Evaluate an Algorithm's Performance?

$$\text{Regret} := \underbrace{\max_{\pi} \sum_{t=1}^T R(\pi)}_{\text{The total reward of the best policy}} - \sum_{t=1}^T R(a_t) = \max_a \underbrace{TR(a)}_{\substack{\uparrow \\ \text{In MAB}}} - \sum_{t=1}^T R(a_t)$$

- “My algorithm obtains  $0.3T$  total reward within  $T$  rounds” – Is my algorithm good?

# Multi-Armed Bandits (MAB)

- Key challenge in MAB: **Exploration**
- The other challenges of RL are not presented in MAB:
  - Generalization (there is no input in MAB)
  - Credit assignments (there is no delayed feedback)
- We will discuss about two categories of exploration strategies
  - Based on mean estimation
  - Based on mean and uncertainty estimation



# **Multi-Armed Bandits**

Based on mean estimation

# The Exploration and Exploitation Trade-off in MAB

- To perform as well as the best policy (i.e., best arm), the learner has to pull the best arm most of the time  
⇒ need to **exploit**
- To identify the best arm, the learner has to try every arm sufficiently many times  
⇒ need to **explore**

# A Simple Strategy: Explore-then-Commit

**Explore-then-commit** (Parameter:  $T_0$ )

In the first  $T_0$  rounds, sample each arm  $T_0/A$  times. **(Explore)**

Compute the **empirical mean**  $\hat{R}(a)$  for each arm  $a$

In the remaining  $T - T_0$  rounds, draw  $\hat{a} = \operatorname{argmax}_a \hat{R}(a)$  **(Exploit)**

What is the *right* amount of exploration ( $T_0$ )?

# Another Simple Strategy: $\epsilon$ -Greedy

Mixing exploration and exploitation in time

**$\epsilon$ -Greedy** (Parameter:  $\epsilon$ )

Take action

$$a_t = \begin{cases} \text{uniform}(\mathcal{A}) & \text{with prob. } \epsilon & \text{(Explore)} \\ \operatorname{argmax}_a \hat{R}_t(a) & \text{with prob. } 1 - \epsilon & \text{(Exploit)} \end{cases}$$

where  $\hat{R}_t(a) = \frac{\sum_{s=1}^{t-1} \mathbb{I}\{a_s=a\} r_s}{\sum_{s=1}^{t-1} \mathbb{I}\{a_s=a\}}$  is the empirical mean of arm  $a$  using samples up to time  $t - 1$ .

What is the *right* amount of exploration ( $\epsilon$ )?

# Comparison

- $\epsilon$ -Greedy is more **robust to non-stationarity** than Explore-then-Exploit
- $\epsilon$ -Greedy has a better performance in the early phase of the learning process

# Quantifying the Estimation Error

In Explore-then-Commit, we obtain  $N = T_0/A$  i.i.d. samples of each arm.

**Key Question:**

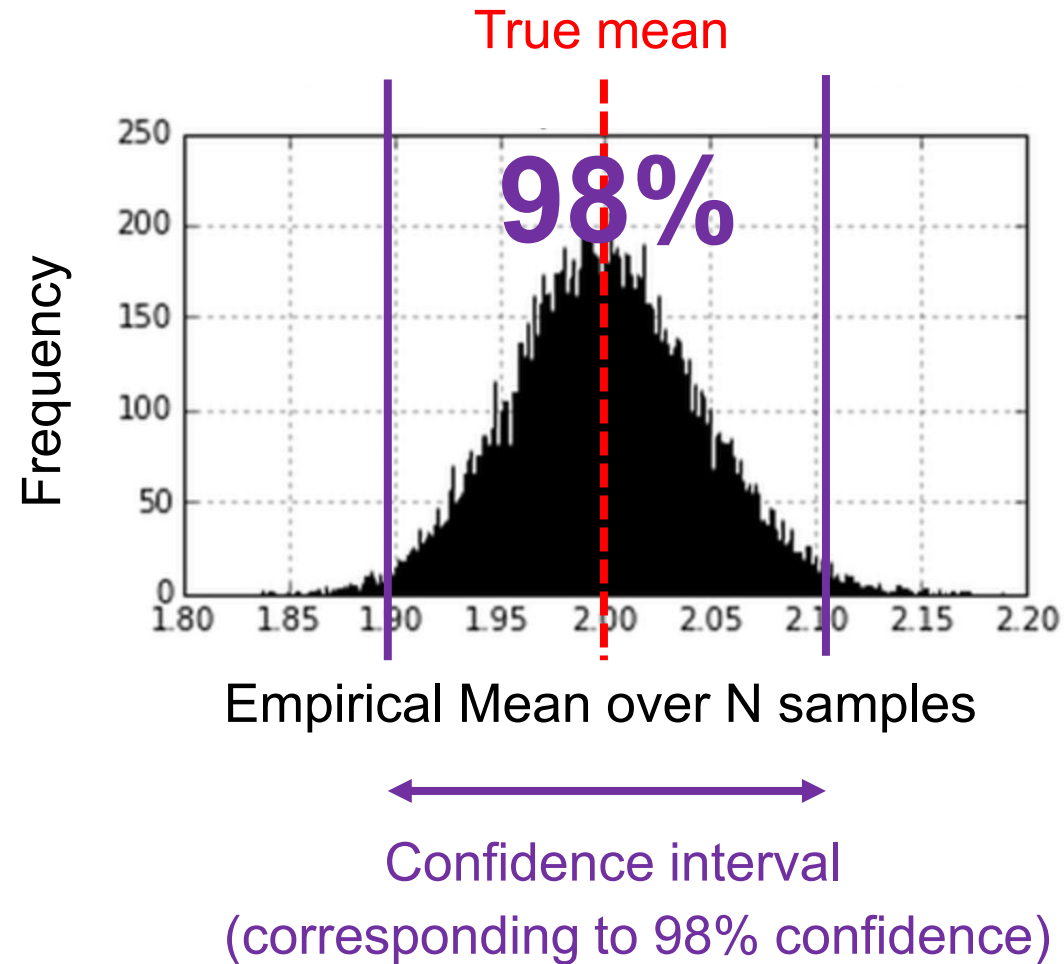
$$\left| \hat{R}(a) - R(a) \right| \leq ? \quad f(N)$$

should decrease with  $N$

Empirical mean  
of  $N$  independent  
samples

True mean

# Quantifying the Estimation Error



# Quantifying the Error: Concentration Inequality

## Theorem. Hoeffding's Inequality

Let  $X_1, \dots, X_N \in [-1, 1]$  be independent random variables with mean  $\mu$ .

Then with probability at least  $1 - \delta$ ,

$$\left| \frac{1}{N} \sum_{i=1}^N X_i - \mu \right| \leq \sqrt{\frac{2 \log(2/\delta)}{N}} .$$



# Quantifying the Estimation Error

In Explore-then-Commit, we obtain  $N = T_0/A$  independent samples of each arm.

$$\left| \hat{R}(a) - R(a) \right| \leq ? \quad f(N) \approx \text{const} \sqrt{\frac{1}{N}}$$

Empirical mean  
of  $N$  independent  
samples

True mean

# Can We Do Better?

In explore-then-commit and  $\epsilon$ -greedy, the probability to choose arms do not depend on the estimated mean (except for the empirically best arm).

... Maybe, the probability of choosing arms can be adaptive to the estimated mean?

**Solution:** Refine the amount of exploration for each arm **based on the current mean estimation.**

(Has to do this carefully to avoid **under-exploration**)

# Refined Exploration

## Boltzmann Exploration (Parameter: $\lambda$ )

In each round, sample  $a_t$  according to

$$\pi_t(a) \propto \exp(\lambda \hat{R}_t(a))$$

where  $\hat{R}_t(a)$  is the empirical mean of arm  $a$  using samples up to time  $t - 1$ .

## Inverse Gap Weighting (Parameter: $\lambda$ )

$$\text{Gap}_t(a) \triangleq \max_b \hat{R}_t(b) - \hat{R}_t(a)$$

$$\pi_t(a) = \begin{cases} \frac{1}{A + \lambda \text{Gap}_t(a)} & \text{for } a \neq \operatorname{argmax}_b \hat{R}_t(b) \\ 1 - \sum_{a' \neq a} \pi_t(a') & \text{for } a = \operatorname{argmax}_b \hat{R}_t(b) \end{cases}$$

# Summary: MAB Based on Mean Estimation

For  $t = 1, 2, \dots, T$ ,

Design a distribution  $\pi_t(\cdot)$  based on the current mean estimation  $\hat{R}_t(\cdot)$

$$\mathbf{EG} \quad \pi_t(a) = (1 - \epsilon) \mathbb{I}\{a = \operatorname{argmax} \hat{R}_t(\cdot)\} + \frac{\epsilon}{A}$$

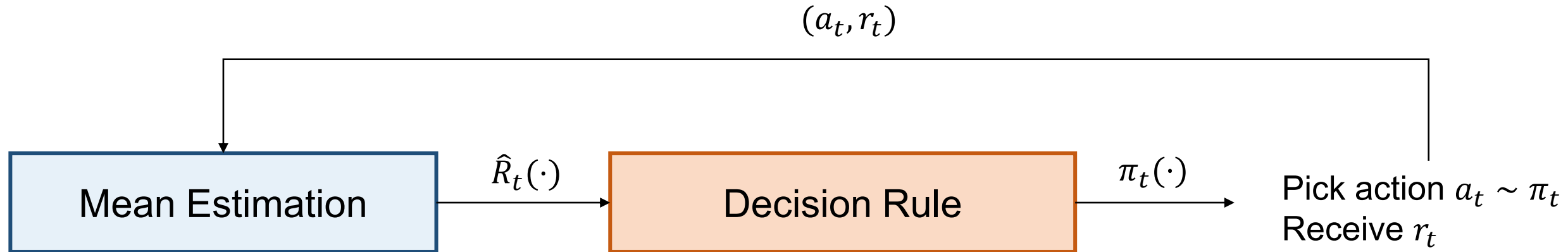
$$\mathbf{BE} \quad \pi_t(a) \propto \exp(\lambda \hat{R}_t(a))$$

$$\mathbf{IGW} \quad \pi_t(a) = \frac{1}{A + \lambda \left( \max_b \hat{R}_t(b) - \hat{R}_t(a) \right)}$$

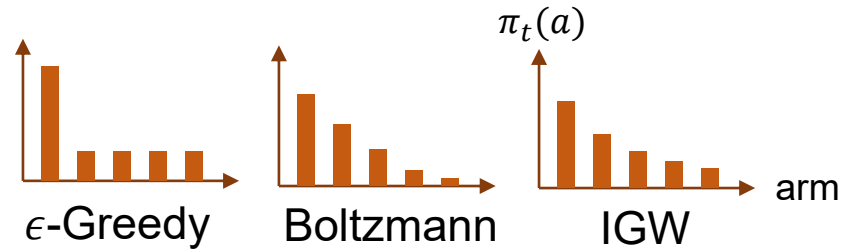
Sample an arm  $a_t \sim \pi_t$  and receive the corresponding reward  $r_t$ .

Refine the mean estimation  $\hat{R}_{t+1}(\cdot)$  with the new sample  $(a_t, r_t)$ .

# Summary: MAB Based on Mean Estimation



$$\hat{R}_t(a) = \frac{\sum_{s=1}^{t-1} \mathbb{I}\{a_s = a\} r_s}{\sum_{s=1}^{t-1} \mathbb{I}\{a_s = a\}}$$



$$\pi_t(a) = (1 - \epsilon) \mathbb{I}\{a = \operatorname{argmax} \hat{R}_t(\cdot)\} + \frac{\epsilon}{A}$$

$$\pi_t(a) \propto \exp(\lambda \hat{R}_t(a))$$

$$\pi_t(a) = \frac{1}{A + \lambda \left( \max_b \hat{R}_t(b) - \hat{R}_t(a) \right)}$$

# Summary: MAB Based on Mean Estimation

- All 3 methods are based on the same **mean estimation**
  - $\epsilon$ -Greedy, Boltzmann exploration, Inverse gap weighting
- The key difference is in the **decision rule**, i.e., the mapping from estimated means  $\hat{R}_t$  to a distribution  $\pi_t$ .
  - The **shape** of the mapping makes differences
- There is a **scalar hyperparameter** that allows for a tradeoff between exploration and exploitation ( $\epsilon$  in EG,  $\lambda$  in BE or IGW)

# Some Experiments

$T = 10000$  rounds

$A = 2$  arms

Reward mean  $R = [0.5, 0.5 - \Delta]$

Bernoulli distribution

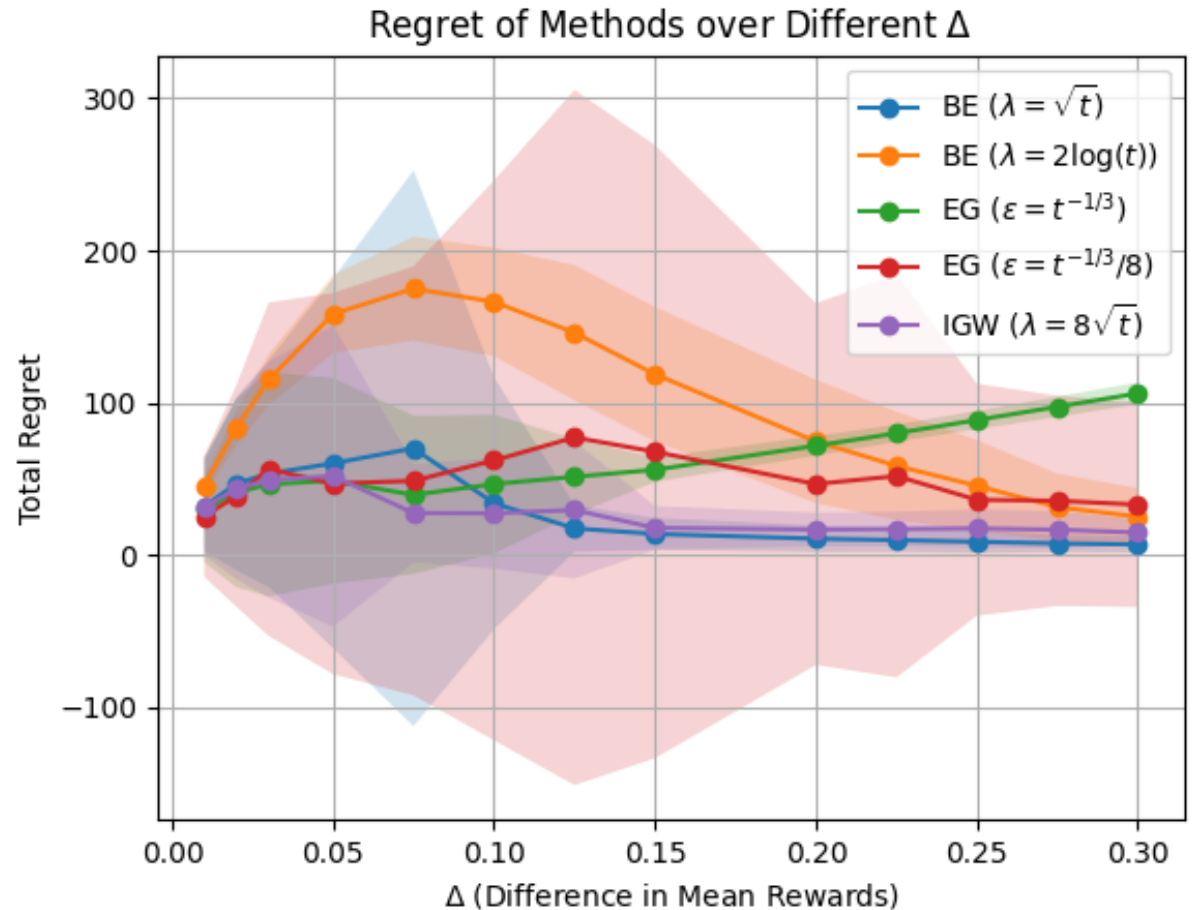
Time-dependent parameters

30 random seeds

[code](#)

## Observations:

- Most algorithms have its worst regret at some intermediate  $\Delta$  value
- Smaller exploration leads to larger variation in performance



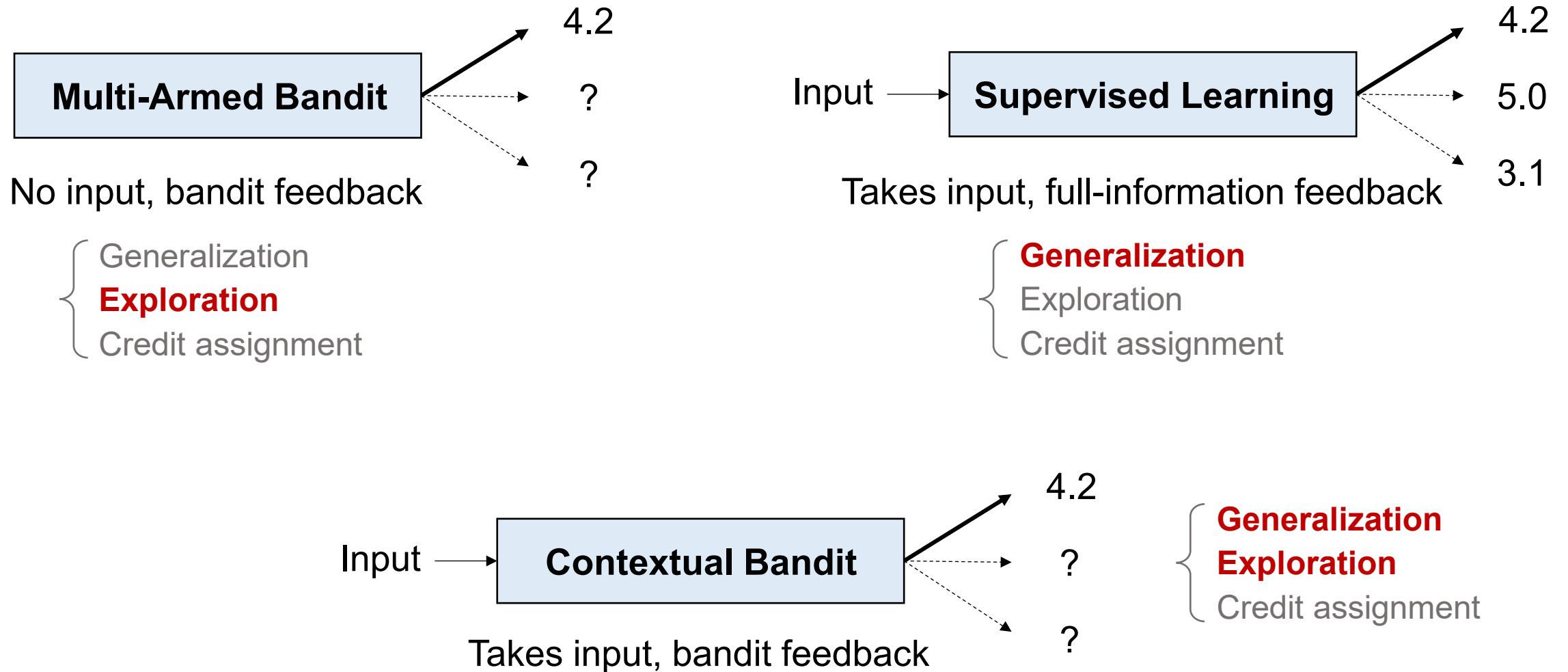
Small  $\Delta$  is easy: don't need to distinguish the two arms  
Large  $\Delta$  is also easy: easy to distinguish the two arms

# Contextual Bandits

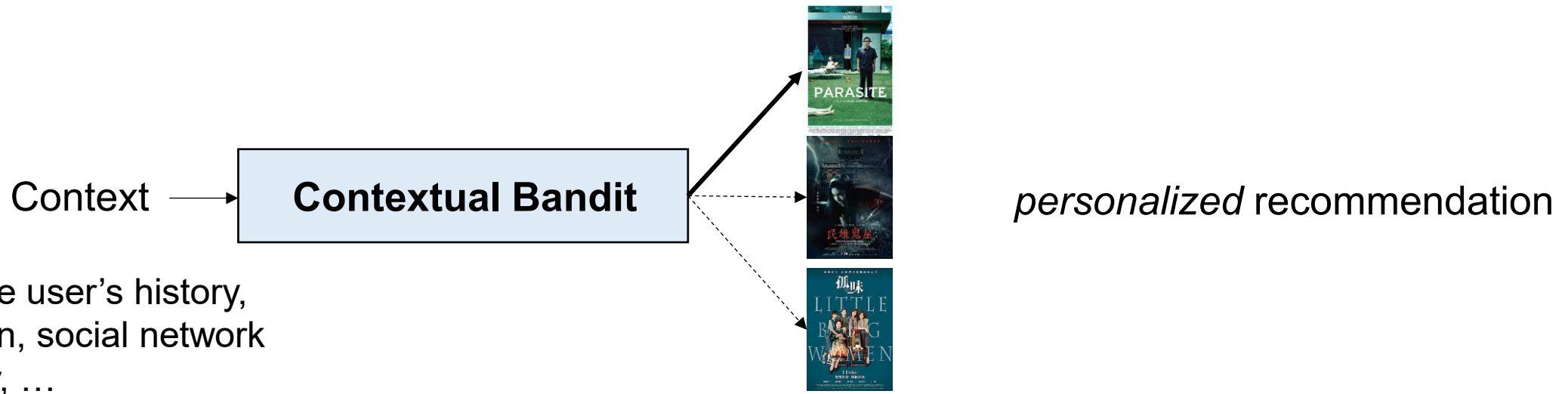
Based on reward function estimation



# Contextual Bandits Generalizes MAB and SL



# Multi-Armed Bandits vs. Contextual Bandits



# Contextual Bandits

For time  $t = 1, 2, \dots, T$ :

Environment generates a context  $x_t \in \mathcal{X}$

Learner chooses an action  $a_t \in \mathcal{A}$

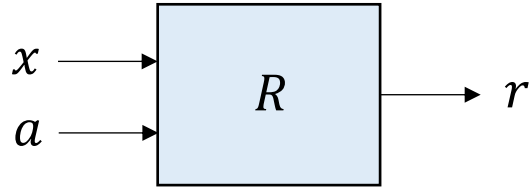
Learner observes  $r_t = R(x_t, a_t) + w_t$

# Discussion

- Contextual bandits is a minimal simultaneous generalization of supervised learning (SL) and multi-armed bandits (MAB)
- We learned about SL in machine learning courses
- We just learned some simple MAB algorithms
  - 3 strategies based on mean estimation
- **Question:** Can you design a contextual bandits algorithm based on the techniques you know for SL and MAB?

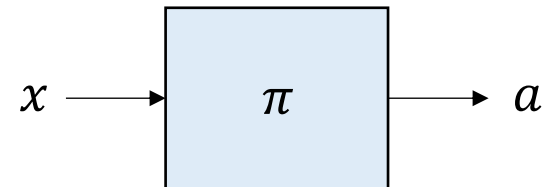
# Two ways to leverage SL techniques in CB

$x$ : context,  $a$ : action,  $r$ : reward



Learn a mapping from  
(context, action) to reward

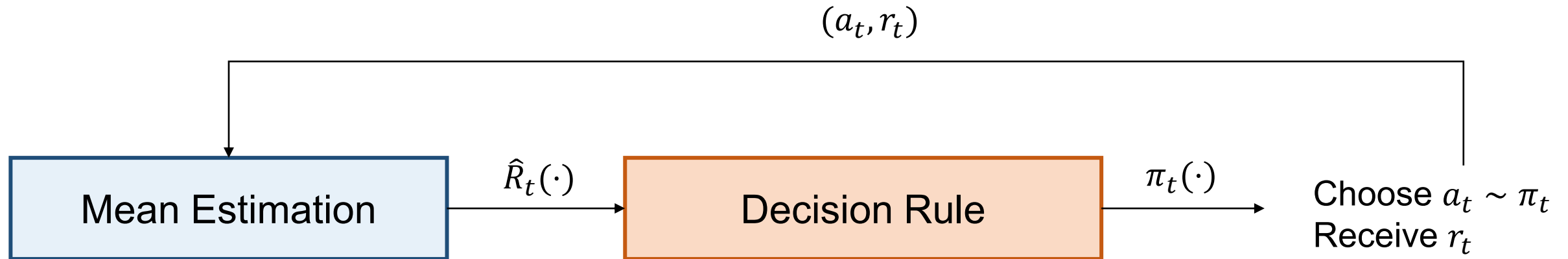
CB with **regression oracle**  
**Value-based** approach  
(discussed next)



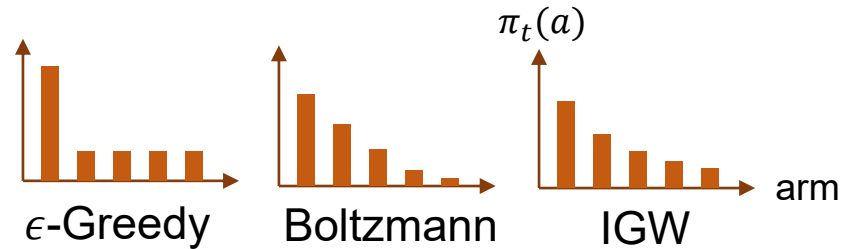
Learn a mapping from  
context to action (or action distribution)

CB with **classification oracle**  
**Policy-based** approach  
(slightly later in the course)

# Recall: MAB Based on Mean Estimation



$$\hat{R}_t(a) = \frac{\sum_{s=1}^{t-1} \mathbb{I}\{a_s = a\} r_s}{\sum_{s=1}^{t-1} \mathbb{I}\{a_s = a\}}$$

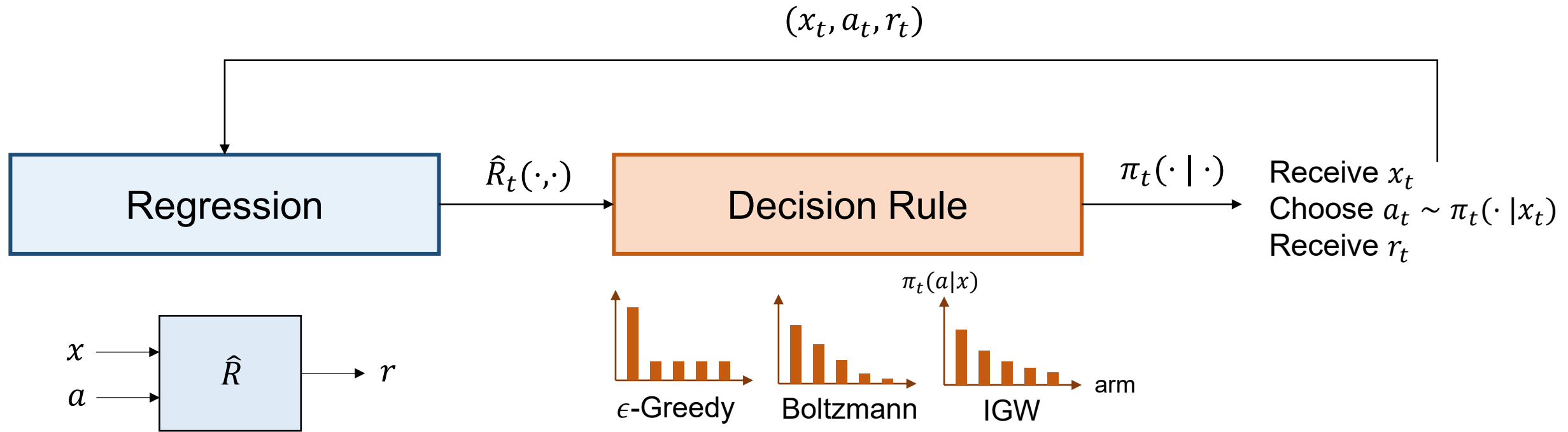


$$\pi_t(a) = (1 - \epsilon) \mathbb{I}\{a = \operatorname{argmax} \hat{R}_t(\cdot)\} + \frac{\epsilon}{A}$$

$$\pi_t(a) \propto \exp(\lambda \hat{R}_t(a))$$

$$\pi_t(a) = \frac{1}{A + \lambda \left( \max_b \hat{R}_t(b) - \hat{R}_t(a) \right)}$$

# CB Based on Reward Function Estimation (Regression)



Train a  $\hat{R}$  such that  $r_i \approx \hat{R}(x_i, a_i)$

$$\pi_t(a|x) = (1 - \epsilon) \mathbb{I}\{a = \operatorname{argmax} \hat{R}_t(x, \cdot)\} + \frac{\epsilon}{A}$$

$$\pi_t(a|x) \propto \exp(\lambda \hat{R}_t(x, a))$$

$$\pi_t(a|x) = \frac{1}{A + \lambda \left( \max_b \hat{R}_t(x, b) - \hat{R}_t(x, a) \right)}$$

# CB Based on Reward Function Estimation

Instantiate a regression procedure  $\hat{R}_1$

For  $t = 1, 2, \dots, T$ ,

Receive context  $x_t$

Design a distribution  $\pi_t(\cdot|x_t)$  based on the estimated reward  $\hat{R}_t(x_t, \cdot)$

$$\mathbf{EG} \quad \pi_t(a|x_t) = (1 - \epsilon)\mathbb{I}\{a = \operatorname{argmax} \hat{R}_t(x_t, \cdot)\} + \frac{\epsilon}{A}$$

$$\mathbf{BE} \quad \pi_t(a|x_t) \propto \exp(\lambda \hat{R}_t(x_t, a))$$

$$\mathbf{IGW} \quad \pi_t(a|x_t) = \frac{1}{A + \lambda \left( \max_b \hat{R}_t(x_t, b) - \hat{R}_t(x_t, a) \right)}$$

Sample an action  $a_t \sim \pi_t(\cdot | x_t)$  and receive the corresponding reward  $r_t$ .

Refine the reward estimator  $\hat{R}_{t+1}(\cdot, \cdot)$  with the new sample  $(x_t, a_t, r_t)$ .



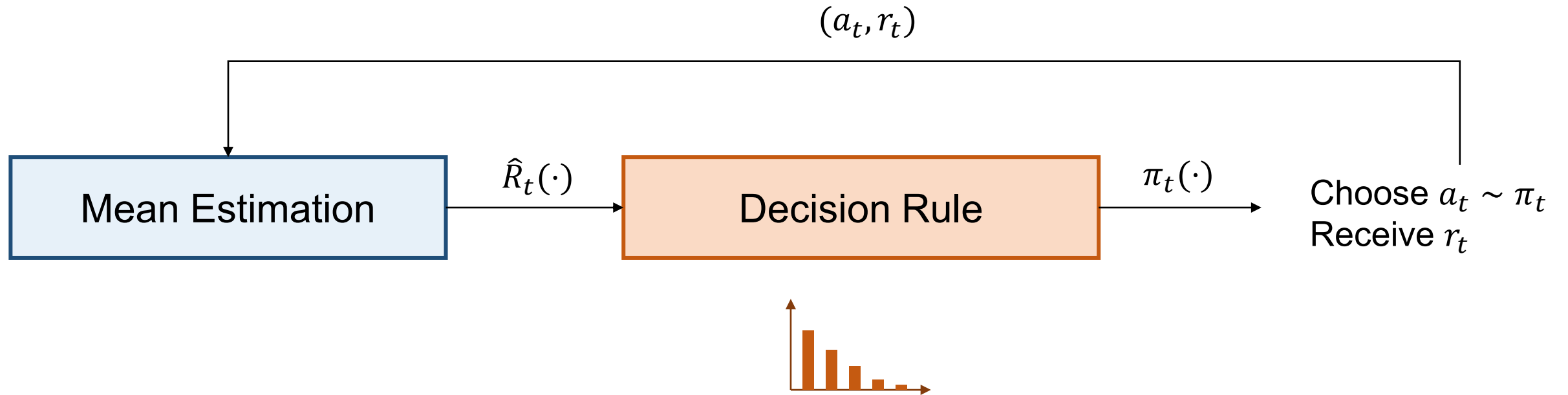
# Summary

- Contextual bandits (CB) simultaneously generalizes supervised learning (SL) and multi-armed bandits (MAB). It captures the challenges of **generalization** and **exploration** in online RL.
- Any MAB algorithm based on “**mean estimation**” can be lifted as a CB algorithm with “**reward function estimation**” by leveraging a regression oracle.
  - This gives a general framework for value-based CB

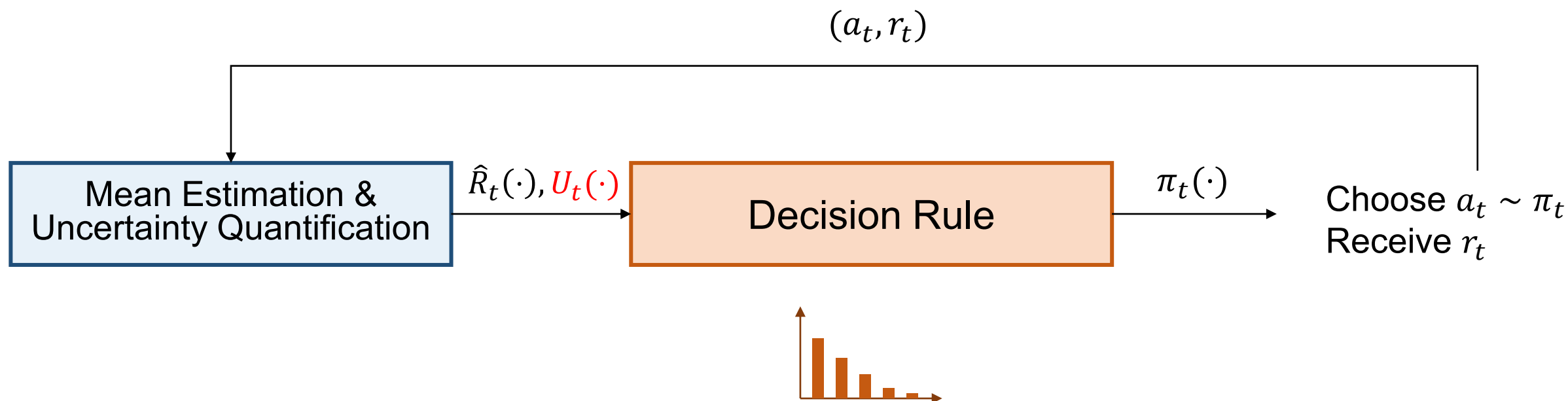
# **Multi-Armed Bandits**

Based on mean estimation and uncertainty quantification

# Recall: MAB Based on Mean Estimation



# MAB Based on Mean Estimation and Uncertainty Quantification



$U_t(a)$ : quantifies the uncertainty of  $\hat{R}_t(a)$

$$|\hat{R}_t(a) - R(a)| \leq \sqrt{\frac{2\log(2/\delta)}{N_t(a)}} \triangleq U_t(a)$$

# Useful Idea: “Optimism in the Face of Uncertainty”

In words:

Act according to the **best plausible world**.

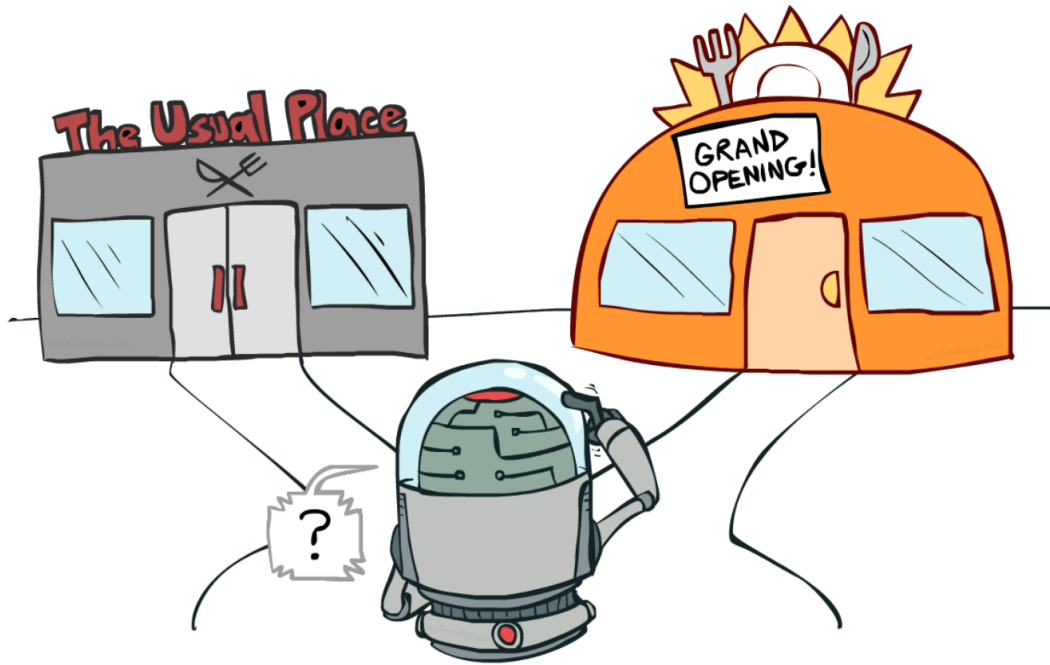


Image source: UC Berkeley CS188

# Another Idea: “Optimism in the Face of Uncertainty”

**In words:**

Act according to the **best plausible world**.

At time  $t$ , suppose that arm  $a$  has been drawn for  $N_t(a)$  times, with empirical mean  $\hat{R}_t(a)$ .

What can we say about the true mean  $R(a)$ ?

$$|R(a) - \hat{R}_t(a)| \leq \sqrt{\frac{2 \log(2/\delta)}{N_t(a)}} \quad \text{w.p.} \geq 1 - \delta$$

What's the most optimistic mean estimation for arm  $a$ ?

$$\hat{R}_t(a) + \sqrt{\frac{2 \log(2/\delta)}{N_t(a)}}$$

# Upper Confidence Bound (UCB)

P Auer, N Cesa-Bianchi, P Fischer. **Finite-time analysis of the multiarmed bandit problem**, 2002.

**UCB** (Parameter:  $\delta$ ) Usually decreases over time as  $\delta_t \sim 1/t$  (drives continual exploration)

In round  $t$ , draw

$$a_t = \operatorname{argmax}_a \hat{R}_t(a) + \sqrt{\frac{2 \log(2/\delta)}{N_t(a)}}$$

Exploration Bonus  
= Amount of Uncertainty

where  $\hat{R}_t(a)$  is the empirical mean of arm  $a$  using samples up to time  $t - 1$ .

$N_t(a)$  is the number of samples of arm  $a$  up to time  $t - 1$ .

cf. Mean-estimation-based algorithms samples  $a_t \sim \pi_t(\cdot) = \text{an increasing function of } \hat{R}_t(\cdot)$

In those algorithms, Hoeffding's inequality is used in the **regret analysis**, but not in the **algorithm**.

# Visualizing UCB

True mean: [0.2, 0.4, 0.6, 0.7] [animation](#) [code](#)



# Summary: Algorithms We Learned So Far

	Approach
Explore-then-Exploit $\epsilon$ -Greedy Boltzmann Exploration Inverse Gap Weighting	Mean estimation + decision rule
Upper Confidence Bound	Mean estimation + uncertainty quantification + decision rule

# Summary

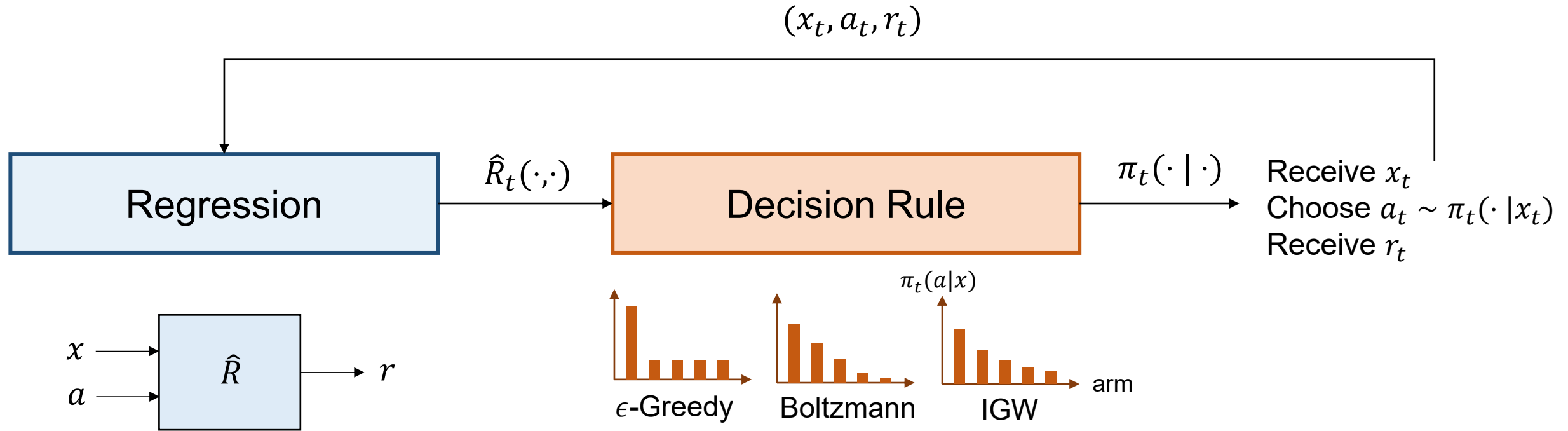
# Summary

## Value-based bandit algorithms

- Multi-armed bandits (non-contextual bandits)
  - Based on mean estimation
  - Based on mean estimation and uncertainty quantification
- Contextual bandits
  - Based on reward function estimation

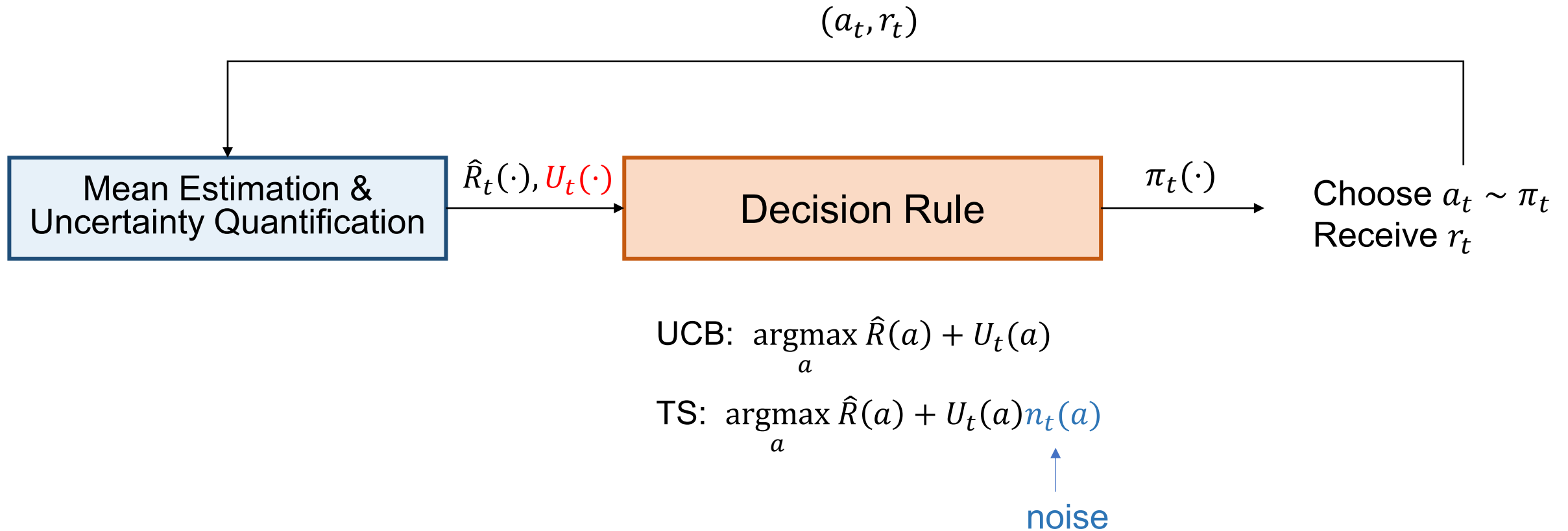
# CB Based on Reward Function Estimation

(Special Case: MAB Based on Mean Estimation)



Train a  $\hat{R}$  such that  $r_i \approx \hat{R}(x_i, a_i)$

# MAB Based on Mean and Uncertainty Estimation



Uncertainty quantification for CB is less trivial – discussed in the future (special topics).