Problem Solving with AI Techniques
Multi-Armed Bandits

Paul Weng

UM-SJTU Joint Institute

VE593, Fall 2018

How to Win at the Casino?

How to Win at the Casino?

- Goal: find "quickly" which of $X_1, \ldots X_K$ has highest mean

Online advertisement (taken from Busa-Fekete)

Online advertisement (taken from Busa-Fekete)

1. click

Online advertisement (taken from Busa-Fekete)

1. click
2. not click

Online advertisement (taken from Busa-Fekete)

1. click
2. not click
3. not click

Online advertisement (taken from Busa-Fekete)

1. click
2. not click
3. not click
4. click

Online Learning

- Batch Learning aka offline learning aka traditional ML
  - Data available as a batch
  - Learn model then use it

Online Learning

- Batch Learning aka offline learning aka traditional ML
  - Data available as a batch
  - Learn model then use it
- Online learning
  - Data available as a stream
  - Continuously improve model and use it

Online Learning

- Batch Learning aka offline learning aka traditional ML
  - Data available as a batch
  - Learn model then use it
- Online learning
  - Data available as a stream
  - Continuously improve model and use it
- Big Data
  - Data continuously generated
  - Large batch can also be consumed in an online way

Applications

- Medical treatment
  - Choose treatment to give patient
  - Find most efficient treatment

Applications

- Medical treatment
  - Choose treatment to give patient
  - Find most efficient treatment
- Financial investment
  - Choose stock to buy and hold for a given period
  - Find stock with highest performance

Applications

- Medical treatment
  - Choose treatment to give patient
  - Find most efficient treatment
- Financial investment
  - Choose stock to buy and hold for a given period
  - Find stock with highest performance
- Hyperparameter tuning
  - Choose hyperparameter value, train model, observe validation error
  - Find best value

Applications

- Medical treatment
  - Choose treatment to give patient
  - Find most efficient treatment
- Financial investment
  - Choose stock to buy and hold for a given period
  - Find stock with highest performance
- Hyperparameter tuning
  - Choose hyperparameter value, train model, observe validation error
  - Find best value
- Model selection
  - Choose trained classifier/regressor for new data point
  - Find best model

Stochastic Multi-Armed Bandits

Model:

- Set of K actions (called arms), defined by unknown distributions $\nu_1, \ldots, \nu_K$ with support in $[0, 1]$
- Each ν_k has mean μ_k (also unknown); $\mu^* = \max_k \mu_k$
- At each time step t, an agent/learner chooses an arm k and receives a random reward (i.e., sample from ν_k)

Stochastic Multi-Armed Bandits

Model:

- Set of K actions (called arms), defined by unknown distributions $\nu_1, \ldots, \nu_K$ with support in $[0, 1]$
- Each ν_k has mean μ_k (also unknown); $\mu^* = \max_k \mu_k$
- At each time step t, an agent/learner chooses an arm k and receives a random reward (i.e., sample from ν_k)

Goal:

- Infinite horizon: maximize sum of received rewards
- Finite horizon: find arm with highest mean

Exploration-Exploitation Dilemma

- Need of solving this dilemma when learning in sequential decision-making problems

- Exploration: Try novel actions, which may reveal to be suboptimal
- Exploitation: Play best action found so far

Exploration-Exploitation Dilemma

- Need of solving this dilemma when learning in sequential decision-making problems

- Exploration: Try novel actions, which may reveal to be suboptimal
- Exploitation: Play best action found so far

- We faced this dilemma in MCTS

Regret Formulation

- Maximizing cumulative reward equivalent to minimizing regret:

$$R_n = \max_k \sum_{t=1}^{n} X_{k,t} - \sum_{t=1}^{n} X_{I_t,t}$$

where $X_{k,t}$ =reward for arm k at time t, I_t =arm chosen at time step t

Regret Formulation

- Maximizing cumulative reward equivalent to minimizing regret:

$$R_n = \max_k \sum_{t=1}^{n} X_{k,t} - \sum_{t=1}^{n} X_{I_t,t}$$

  where $X_{k,t}$=reward for arm k at time t, I_t=arm chosen at time step t
- Performance with respect to best fixed choice

Regret Formulation

- Maximizing cumulative reward equivalent to minimizing regret:

$$R_n = \max_k \sum_{t=1}^{n} X_{k,t} - \sum_{t=1}^{n} X_{I_t,t}$$

where $X_{k,t}$=reward for arm k at time t, I_t=arm chosen at time step t

- Performance with respect to best fixed choice
- Pseudo regret:

$$\begin{aligned}\overline{R}_n &= \max_k \mathbb{E}[\sum_{t=1}^{n} X_{k,t} - \sum_{t=1}^{n} X_{I_t,t}] \\ &= n\mu^* - \sum_k \mathbb{E}[T_k(n)]\mu_k \\ &= \sum_k \mathbb{E}[T_k(n)]\Delta_k\end{aligned}$$

where $\Delta_k = \mu^* - \mu_k$, $T_k(n)$=# of times learner selected arm k after n rounds

ε-Greedy Algorithm

- Choose best arm found so far, but explore with small probability

---

for $k = 1, \ldots, K$ **do**
    $X_{k,1} \sim \nu_k$
    $\hat{\mu}_{k,1} = X_{k,1}$
for $t = K+1, K+2, \ldots$ **do**
    $\varepsilon_t \leftarrow \min(1, \frac{cK}{d^2 t})$
    if $\mathcal{U}([0,1]) < \varepsilon_t$ **then** $I_t \sim \mathcal{U}(\{1, 2, \ldots, K\})$;
    else $I_t \leftarrow \arg\max_k \hat{\mu}_{k, T_k(t)}$;
    $X_{I_t, T_{I_t}(t)} \sim \nu_{I_t}$
    $\hat{\mu}_{I_t, T_{I_t}(t)} = \frac{T_{I_t}(t)-1}{T_{I_t}(t)} \hat{\mu}_{I_t, T_{I_t}(t)} + \frac{1}{T_{I_t}(t)} X_{I_t, T_{I_t}(t)}$

Pseudo Regret Bound for ε-Greedy

- Choose $c > 5$ and $0 < d \leq \min_{k:\mu_k<\mu^*} \Delta_k$
- Theorem: If ε-greedy is run over T steps, its pseudo regret is bounded by $O(K \log T)$

- Issue: d is not known, incorrect value may lead to bad performance

UCB Algorithm

- Optimism in the face of uncertainty

---

for $k = 1, \ldots, K$ **do**

$\quad X_{k,1} \sim \nu_k$

$\quad \hat{\mu}_{k,1} = X_{k,1}$

for $t = K+1, K+2, \ldots$ **do**

$\quad I_t \leftarrow \arg\max_k \hat{\mu}_{k, T_{k(t)}} + \sqrt{\frac{2 \log t}{T_k(t)}}$

$\quad X_{I_t, T_{I_t}(t)} \sim \nu_{I_t}$

$\quad \hat{\mu}_{I_t, T_{I_t}(t)} = \frac{T_{I_t}(t)-1}{T_{I_t}(t)} \hat{\mu}_{I_t, T_{I_t}(t)} + \frac{1}{T_{I_t}(t)} X_{I_t, T_{I_t}(t)}$

Pseudo Regret Bound for UCB

- Theorem: If algorithm UCB is run over T steps, its pseudo regret is bounded by $O(K \log T)$

- Hoeffding's inequality: $\mathbb{P}\big(|\frac{1}{n}\sum_{i=1}^{n} X_i - \mu| \geq \varepsilon\big) \leq 2e^{-2\varepsilon^2 n}$

Adversarial Multi-Armed Bandits

Model

- Set of K arms, losses are chosen by an adversary at each time step
- At each time step t, an agent/learner chooses an arm and corresponding loss is revealed

Goal:

- Infinite horizon: minimize sum of received losses

Regret Formulation

- Minimizing cumulative loss equivalent to minimizing regret:

$$R_n = \sum_{t=1}^{n} \ell_{I_t,t} - \min_k \sum_{t=1}^{n} \ell_{k,t}$$

  where I_t is the arm chosen at time step t

- Performance with respect to best choice (known a posteriori)
- Pseudo regret:

$$\overline{R}_n = \mathbb{E}[\sum_{t=1}^{n} \ell_{I_t,t}] - \min_k \mathbb{E}[\sum_{t=1}^{n} \ell_{k,t}]$$

  where expectation is w.r.t. randomization of learner and adversary

Exp3

- Randomization is necessary!
- Exp3 = Exponential Weights for Exploration and Exploitation

---

$\forall k = 1, \ldots, K,\ p_k = 1/K$

for $t = 1, 2, \ldots$ **do**

  I_t sampled from distribution $\boldsymbol{p} = (p_1, \ldots, p_K)$

  for $k = 1, \ldots, K$ **do**

    $\tilde{\ell}_{k,t} = \frac{\ell_{k,t}}{p_k}[I_t = k]$

    $\tilde{L}_{k,t} = \tilde{L}_{k,t-1} + \tilde{\ell}_{k,t}$

  $\forall k = 1, \ldots, K,\ p_k = \frac{\exp(-\eta_t \tilde{L}_{k,t})}{\sum_i \exp(-\eta_t \tilde{L}_{i,t})}$

Regret Bound

- Theorem: If Exp3 is run with $\eta_t = \eta = \sqrt{\frac{2 \ln K}{TK}}$, then

$$\overline{R}_T \leq \sqrt{2TK \ln K}$$

Regret Bound

- Theorem: If Exp3 is run with $\eta_t = \eta = \sqrt{\frac{2 \ln K}{TK}}$, then

$$\overline{R}_T \leq \sqrt{2TK \ln K}$$

- Issue: T may not be known in advance

Regret Bound

- Theorem: If Exp3 is run with $\eta_t = \eta = \sqrt{\frac{2 \ln K}{TK}}$, then

$$\overline{R}_T \leq \sqrt{2TK \ln K}$$

- Issue: T may not be known in advance
- Theorem: If Exp3 is run with $\eta_t = \sqrt{\frac{\ln K}{tK}}$, then

$$\overline{R}_T \leq 2\sqrt{TK \ln K}$$

Extensions

- Expert setting
- Combinatorial MAB
- Contextual MAB
- Duelling MAB
- Mortal MAB
- And many more!