

# Aggressive Learning for Contextual Bandits

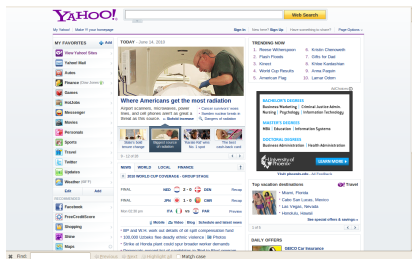
John Langford (Yahoo!)

{ With Miroslav Dudik, Daniel Hsu, Satyen Kale, Nikos  
Karampatziakis, Lev Reyzin, and Tong Zhang }

And Alina Beygelzimer

Snowbird, April 16, 2011

# Example of Learning through Exploration

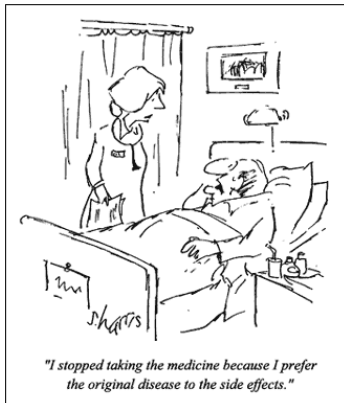


Repeatedly:

- 1 A user comes to Yahoo! (with history of previous visits, IP address, data related to his Yahoo! account)
- 2 Yahoo! chooses information to present (from urls, ads, news stories)
- 3 The user reacts to the presented information (clicks on something, clicks, comes back and clicks again, et cetera)

Yahoo! wants to interactively choose content and use the observed feedback to improve future content choices.

# Another Example: Clinical Decision Making



Repeatedly:

- 1 A patient comes to a doctor with symptoms, medical history, test results
- 2 The doctor chooses a treatment
- 3 The patient responds to it

The doctor wants a policy for choosing targeted treatments for individual patients.

# The Contextual Bandit Setting

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

- 1 The world produces some context  $x_t \in X$
- 2 The learner chooses an action  $a_t \in \{1, \dots, K\}$
- 3 The world reacts with reward  $r_t(a_t) \in [0, 1]$

Goal:

# The Contextual Bandit Setting

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

- 1 The world produces some context  $x_t \in X$
- 2 The learner chooses an action  $a_t \in \{1, \dots, K\}$
- 3 The world reacts with reward  $r_t(a_t) \in [0, 1]$

**Goal:** Learn a good policy for choosing actions given context.

# The Contextual Bandit Setting

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

- 1 The world produces some context  $x_t \in X$
- 2 The learner chooses an action  $a_t \in \{1, \dots, K\}$
- 3 The world reacts with reward  $r_t(a_t) \in [0, 1]$

**Goal:** Learn a good policy for choosing actions given context.

What does learning mean?

# The Contextual Bandit Setting

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

- 1 The world produces some context  $x_t \in X$
- 2 The learner chooses an action  $a_t \in \{1, \dots, K\}$
- 3 The world reacts with reward  $r_t(a_t) \in [0, 1]$

**Goal:** Learn a good policy for choosing actions given context.

**What does learning mean?** Efficiently competing with a large reference class of possible policies  $\Pi = \{\pi : X \rightarrow \{1, \dots, K\}\}$ :

$$\text{Regret} = \max_{\pi \in \Pi} \sum_{t=1}^T r_t(\pi(x_t)) - \sum_{t=1}^T r_t(a_t)$$

# The Contextual Bandit Setting

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

- 1 The world produces some context  $x_t \in X$
- 2 The learner chooses an action  $a_t \in \{1, \dots, K\}$
- 3 The world reacts with reward  $r_t(a_t) \in [0, 1]$

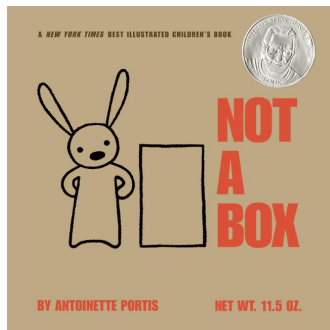
**Goal:** Learn a good policy for choosing actions given context.

**What does learning mean?** Efficiently competing with a large reference class of possible policies  $\Pi = \{\pi : X \rightarrow \{1, \dots, K\}\}$ :

$$\text{Regret} = \max_{\pi \in \Pi} \sum_{t=1}^T r_t(\pi(x_t)) - \sum_{t=1}^T r_t(a_t)$$

Other names: associative reinforcement learning, associative bandits, learning with partial feedback, bandits with side information

# Basic Observation #1



## This is not a supervised learning problem:

- We don't know the reward of actions not taken—loss function is unknown even at training time.
- Exploration is required to succeed (but still simpler than reinforcement learning – we know which action is responsible for each reward)

## Basic Observation #2



This is not just a bandit problem:

- In the bandit setting, there is no  $x$ , and the goal is to compete with the set of constant actions. Too weak in practice.
- Generalization across  $x$  is required to succeed.

# One algorithm was known: EXP4

**Theorem:** [Auer et al. '95] For all oblivious sequences  $(x_1, r_1), \dots, (x_T, r_T)$ , EXP4 has expected regret

$$O\left(\sqrt{TK \ln |\Pi|}\right).$$

**Theorem:** [Auer et al. '95] For any  $T$ , there exists an iid sequence such that the expected regret of any player is  $\Omega(\sqrt{TK})$ .

EXP4 can be modified to succeed with high probability or over VC sets when the world is IID.

[Beygelzimer, et al. 2011].

EXP4 is slow

$$\Omega(T|\Pi|)$$

Exponentially slower than is typical for supervised learning. Can we make an algorithm taking advantage of supervised learning systems?

## Policy\_Elimination

Let  $\Pi_0 = \Pi$

$$\mu_t = 1/\sqrt{Kt}$$

$$\eta_t(\pi) = \frac{1}{t} \sum_{x,a,r_a} \frac{r_a I(\pi(x)=a)}{p(a|x)}$$

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

- 1 Choose distribution  $P_t$  over  $\Pi_{t-1}$  s.t.  $\forall \pi \in \Pi_{t-1}$ :

$$\mathbf{E}_{x \sim D_X} \left[ \frac{1}{(1 - K\mu_t) \Pr_{\pi' \sim P_t}(\pi'(x) = \pi(x)) + \mu_t} \right] \leq 2K$$

- 2 observe  $x_t$
- 3 Let  $p_t(a) = (1 - K\mu_t) \Pr_{\pi \sim P_t}(\pi(x) = a) + \mu_t$
- 4 Choose  $a_t \sim p_t$  and observe reward  $r_t$
- 5 Let  $\Pi_t = \{\pi \in \Pi_{t-1} : \eta_t(\pi) \geq \max_{\pi' \in \Pi_{t-1}} \eta_t(\pi') - K\mu_t\}$

## Policy\_Elimination

Let  $\Pi_0 = \Pi$

$\mu_t =$  minimum action probability

$\eta_t(\pi) =$  importance weighted empirical reward estimate

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

- 1 Choose distribution  $P_t$  over  $\Pi_{t-1}$  s.t.  $\forall \pi \in \Pi_{t-1}$ :

$$\mathbf{E}_{x \sim D_x} \left[ \frac{1}{(1 - K\mu_t) \Pr_{\pi' \sim P_t}(\pi'(x) = \pi(x)) + \mu_t} \right] \leq 2K$$

- 2 observe  $x_t$
- 3 Let  $p_t(a) = (1 - K\mu_t) \Pr_{\pi \sim P_t}(\pi(x) = a) + \mu_t$
- 4 Choose  $a_t \sim p_t$  and observe reward  $r_t$
- 5 Let  $\Pi_t = \{\pi \in \Pi_{t-1} : \eta_t(\pi) \geq \max_{\pi' \in \Pi_{t-1}} \eta_t(\pi') - K\mu_t\}$

## Policy\_Elimination

Let  $\Pi_0 = \Pi$

$\mu_t =$  minimum action probability

$\eta_t(\pi) =$  importance weighted empirical reward estimate

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

- 1 Find a distribution  $P$  over remaining policies  $\Pi_{t-1}$  which makes the probability of each remaining policy's action  $> \frac{1}{K}$ .
- 2 observe  $x_t$
- 3 Let  $p_t(a) = (1 - K\mu_t) \Pr_{\pi \sim P_t}(\pi(x) = a) + \mu_t$
- 4 Choose  $a_t \sim p_t$  and observe reward  $r_t$
- 5 Let  $\Pi_t = \{\pi \in \Pi_{t-1} : \eta_t(\pi) \geq \max_{\pi' \in \Pi_{t-1}} \eta_t(\pi') - K\mu_t\}$

## Policy\_Elimination

Let  $\Pi_0 = \Pi$

$\mu_t$  = minimum action probability

$\eta_t(\pi)$  = importance weighted empirical reward estimate

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

- 1 Find a distribution  $P$  over remaining policies  $\Pi_{t-1}$  which makes the probability of each remaining policy's action  $> \frac{1}{K}$ .
- 2 observe  $x_t$
- 3 Project the distribution over policies onto a distribution over actions  $p_t$ .
- 4 Choose  $a_t \sim p_t$  and observe reward  $r_t$
- 5 Let  $\Pi_t = \{\pi \in \Pi_{t-1} : \eta_t(\pi) \geq \max_{\pi' \in \Pi_{t-1}} \eta_t(\pi') - K\mu_t\}$

## Policy\_Elimination

Let  $\Pi_0 = \Pi$

$\mu_t =$  minimum action probability

$\eta_t(\pi) =$  importance weighted empirical reward estimate

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

- 1 Find a distribution  $P$  over remaining policies  $\Pi_{t-1}$  which makes the probability of each remaining policy's action  $> \frac{1}{K}$ .
- 2 observe  $x_t$
- 3 Project the distribution over policies onto a distribution over actions  $p_t$ .
- 4 Choose  $a_t \sim p_t$  and observe reward  $r_t$
- 5 Let  $\Pi_t =$  those policies not much worse than the empirical best.

# Analysis

For all sets of policies  $\Pi$ , for all distributions  $D(x, \vec{r})$ , if the world is IID w.r.t.  $D$ , with high probability Policy\_Elimination has expected regret

$$O\left(\sqrt{TK \ln |\Pi|}\right).$$

A key lemma: For any set of policies  $\Pi$  and any distribution over  $x$ , step 1 is possible.

# Analysis

For all sets of policies  $\Pi$ , for all distributions  $D(x, \vec{r})$ , if the world is IID w.r.t.  $D$ , with high probability Policy\_Elimination has expected regret

$$O\left(\sqrt{TK \ln |\Pi|}\right).$$

A key lemma: For any set of policies  $\Pi$  and any distribution over  $x$ , step 1 is possible.

Proof: Consider the game:

$$\min_P \max_Q E_{\pi \sim Q} E_x \frac{1}{(1 - K\mu_t) \Pr_{\pi' \sim P}(\pi(x) = \pi'(x)) + \mu_t}$$

# Analysis

For all sets of policies  $\Pi$ , for all distributions  $D(x, \vec{r})$ , if the world is IID w.r.t.  $D$ , with high probability Policy\_Elimination has expected regret

$$O\left(\sqrt{TK \ln |\Pi|}\right).$$

A key lemma: For any set of policies  $\Pi$  and any distribution over  $x$ , step 1 is possible.

Proof: Consider the game:

$$\min_P \max_Q E_{\pi \sim Q} E_x \frac{1}{(1 - K\mu_t) \Pr_{\pi' \sim P}(\pi(x) = \pi'(x)) + \mu_t}$$

Minimax magic!

$$= \max_Q \min_P E_{\pi \sim Q} E_x \frac{1}{(1 - K\mu_t) \Pr_{\pi' \sim P}(\pi(x) = \pi'(x)) + \mu_t}$$

# Analysis

For all sets of policies  $\Pi$ , for all distributions  $D(x, \vec{r})$ , if the world is IID w.r.t.  $D$ , with high probability Policy\_Elimination has expected regret

$$O\left(\sqrt{TK \ln |\Pi|}\right).$$

A key lemma: For any set of policies  $\Pi$  and any distribution over  $x$ , step 1 is possible.

Proof: Consider the game:

$$\min_P \max_Q E_{\pi \sim Q} E_x \frac{1}{(1-K\mu_t) \Pr_{\pi' \sim P}(\pi(x)=\pi'(x))+\mu_t}$$

Minimax magic!

$$= \max_Q \min_P E_{\pi \sim Q} E_x \frac{1}{(1-K\mu_t) \Pr_{\pi' \sim P}(\pi(x)=\pi'(x))+\mu_t}$$

Let  $P = Q$

$$\leq \max_Q E_{\pi \sim Q} E_x \frac{1}{(1-K\mu_t) \Pr_{\pi' \sim Q}(\pi(x)=\pi'(x))+\mu_t}$$

# Analysis

For all sets of policies  $\Pi$ , for all distributions  $D(x, \vec{r})$ , if the world is IID w.r.t.  $D$ , with high probability Policy\_Elimination has expected regret

$$O\left(\sqrt{TK \ln |\Pi|}\right).$$

A key lemma: For any set of policies  $\Pi$  and any distribution over  $x$ , step 1 is possible.

Proof: Consider the game:

$$\min_P \max_Q E_{\pi \sim Q} E_x \frac{1}{(1-K\mu_t) \Pr_{\pi' \sim P}(\pi(x)=\pi'(x)) + \mu_t}$$

Minimax magic!

$$= \max_Q \min_P E_{\pi \sim Q} E_x \frac{1}{(1-K\mu_t) \Pr_{\pi' \sim P}(\pi(x)=\pi'(x)) + \mu_t}$$

Let  $P = Q$

$$\leq \max_Q E_{\pi \sim Q} E_x \frac{1}{(1-K\mu_t) \Pr_{\pi' \sim Q}(\pi(x)=\pi'(x)) + \mu_t}$$

Linearity of Expectation

$$= \max_Q E_x \sum_a \frac{\Pr_{\pi \sim Q}(\pi(x)=a)}{(1-K\mu_t) \Pr_{\pi' \sim Q}(\pi'(x)=a) + \mu_t}$$

# Another Use of minimax

[DHKKLRZ11]

## Randomized\_UCB

Let  $\mu_t =$  minimum action probability.

$$\Delta_t(\pi) = \max_{\pi'} \eta_t(\pi') - \eta_t(\pi)$$

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

- 1 Choose distribution  $P$  over  $\Pi$  minimizing  $E_{\pi \sim P}[\Delta_t(\pi)]$  s.t.  $\forall \pi$ :

$$E_{x \sim h_t} \left[ \frac{1}{(1 - K\mu_t) \Pr_{\pi' \sim P}(\pi'(x) = \pi(x)) + \mu_t} \right] \leq \max\{2K, Ct(\Delta_t(\pi))^2\}$$

using oracle learning algorithm(\*).

- 2 observe  $x_t$
- 3 Let  $p_t(a) = (1 - K\mu_t) \Pr_{\pi \sim P_t}(\pi(x) = a) + \mu_t$
- 4 Choose  $a_t \sim p_t$  and observe reward  $r_t$

(\* ) Much complexity hidden here.

# Another Use of minimax

[DHKKLRZ11]

## Randomized\_UCB

Let  $\mu_t =$  minimum action probability.

$\Delta_t(\pi) =$  empirical regret

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

- 1 Choose distribution  $P$  over  $\Pi$  minimizing  $E_{\pi \sim P}[\Delta_t(\pi)]$  s.t.  $\forall \pi$ :

$$E_{x \sim h_t} \left[ \frac{1}{(1 - K\mu_t) \Pr_{\pi' \sim P}(\pi'(x) = \pi(x)) + \mu_t} \right] \leq \max\{2K, Ct(\Delta_t(\pi))^2\}$$

using oracle learning algorithm(\*).

- 2 observe  $x_t$
- 3 Let  $p_t(a) = (1 - K\mu_t) \Pr_{\pi \sim P_t}(\pi(x) = a) + \mu_t$
- 4 Choose  $a_t \sim p_t$  and observe reward  $r_t$

(\* ) Much complexity hidden here.

# Another Use of minimax

[DHKKLRZ11]

## Randomized\_UCB

Let  $\mu_t =$  minimum action probability.

$\Delta_t(\pi) =$  empirical regret

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

- 1 Use oracle learning algorithm to find a sparse distribution  $P$  over  $\Pi$  inducing large probability on all good policy's actions and possibly small probability on bad policy's actions.
- 2 observe  $x_t$
- 3 Let  $p_t(a) = (1 - K\mu_t) \Pr_{\pi \sim P_t}(\pi(x) = a) + \mu_t$
- 4 Choose  $a_t \sim p_t$  and observe reward  $r_t$

# Another Use of minimax

[DHKKLRZ11]

## Randomized\_UCB

Let  $\mu_t =$  minimum action probability.

$\Delta_t(\pi) =$  empirical regret

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

- 1 Use oracle learning algorithm to find a sparse distribution  $P$  over  $\Pi$  inducing large probability on all good policy's actions and possibly small probability on bad policy's actions.
- 2 observe  $x_t$
- 3 Project the distribution over policies onto a distribution over actions  $p_t$ .
- 4 Choose  $a_t \sim p_t$  and observe reward  $r_t$

# Randomized\_UCB analysis

For all sets of policies  $\Pi$ , for all distributions  $D(x, \vec{r})$ , if the world is IID w.r.t.  $D$ , with high probability Randomized\_UCB has expected regret

$$O\left(\sqrt{TK \ln |\Pi|}\right).$$

# Randomized\_UCB analysis

For all sets of policies  $\Pi$ , for all distributions  $D(x, \vec{r})$ , if the world is IID w.r.t.  $D$ , with high probability Randomized\_UCB has expected regret

$$O\left(\sqrt{TK \ln |\Pi|}\right).$$

And: Given an cost sensitive optimization oracle for  $\Pi$ , Randomized\_UCB runs in time  $\text{Poly}(t, K, \log |\Pi|)$ !

# Randomized\_UCB analysis

For all sets of policies  $\Pi$ , for all distributions  $D(x, \vec{r})$ , if the world is IID w.r.t.  $D$ , with high probability Randomized\_UCB has expected regret

$$O\left(\sqrt{TK \ln |\Pi|}\right).$$

And: Given an cost sensitive optimization oracle for  $\Pi$ , Randomized\_UCB runs in time  $\text{Poly}(t, K, \log |\Pi|)$ !

Uses ellipsoid algorithm for convex programming. First ever general  $\text{Poly}(\log |\Pi|)$  algorithm for contextual bandits.

## Final Thoughts and pointers

We can be aggressive in other ways as well. For example, if rewards are delayed, regret is additive rather than multiplicative in the delay.

Great Background in Exploration and Learning Tutorial.

[http://hunch.net/~exploration\\_learning](http://hunch.net/~exploration_learning)

Further Contextual Bandit discussion: <http://hunch.net/>