<table>
<thead>
<tr>
<th>Time</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left arm</td>
<td>$1</td>
<td>$0</td>
<td>$1</td>
<td>$1</td>
<td>$1</td>
<td>$0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Right arm</td>
<td>$1</td>
<td>$0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Five rounds to go. Which arm would you play next?
Overview

• What are bandits, and why you should care
• Finite-armed stochastic bandits
• A brief intro to finite-armed adversarial bandits (if time)
• Break
• Contextual and linear bandits
• Summary and discussion

• Details for the core ideas, rather than a broad overview
• Plenty of references on where to find more
• Please ask questions!
What’s in a name? A tiny bit of history

First bandit algorithm proposed by Thompson (1933)

Bush and Mosteller (1953) were interested in how mice behaved in a T-maze
Why care about bandits?

1. Many applications
2. They isolate an important component of reinforcement learning: exploration-vs-exploitation
3. Rich and beautiful (we think) mathematically
Applications

• Clinical trials/dose discovery
• Recommendation systems (movies/news/etc)
• Advert placement
• A/B testing
• Network routing
• Dynamic pricing (eg., for Amazon products)
• Waiting problems (when to auto-logout your computer)
• Ranking (eg., for search)
• A component of game-playing algorithms (MCTS)
• Resource allocation
• A way of isolating one interesting part of reinforcement learning
Applications

- Clinical trials/dose discovery
- Recommendation systems (movies/news/etc)
- Advert placement
- A/B testing
- Network routing
- Dynamic pricing (eg., for Amazon products)
- Waiting problems (when to auto-logout your computer)
- Ranking (eg., for search)
- A component of game-playing algorithms (MCTS)
- Resource allocation
- A way of isolating one interesting part of reinforcement learning

Lots for you to do!
Finite-armed Bandits

- $K$ actions
- $n$ rounds
- In each round $t$ the learner chooses an action

$$A_t \in \{1, 2, \ldots, K\}.$$ 

- Observes reward $X_t \sim P_{A_t}$ where $P_1, P_2, \ldots, P_K$ are unknown distributions
Distributional assumptions

While $P_1, P_2, \ldots, P_K$ are not known in advance, we make some assumptions:

- $P_i$ is Bernoulli with unknown bias $\mu_i \in [0, 1]$
- $P_i$ is Gaussian with unit variance and unknown mean $\mu_i \in \mathbb{R}$
- $P_i$ is subgaussian
- $P_i$ is supported on $[0, 1]$
- $P_i$ has variance less than one
- ...

As usual, stronger assumptions lead to stronger bounds

This tutorial All reward distributions are Gaussian (or subgaussian) with unit variance
What makes a bandit problem?

How to tell if your problem is a bandit problem?

Three core properties:

1. Sequentially taking actions of unknown quality
2. The feedback provides information about quality of chosen action
3. There is no state

Things are considerably easier if the problem is close to stationary, but it is not a defining feature of a bandit problem
Example: A/B testing

- Business wants to optimize their webpage
- Actions correspond to ‘A’ and ‘B’
- Users arrive at webpage sequentially
- Algorithm chooses either ‘A’ or ‘B’
- Receives activity feedback (the reward)
• Let $\mu_i$ be the mean reward of distribution $P_i$
• $\mu^* = \max_i \mu_i$ is the maximum mean
• The regret is

$$R_n = n \mu^* - \mathbb{E} \left[ \sum_{t=1}^{n} X_t \right]$$

• Policies for which the regret is sublinear are learning
• Of course we would like to make it as ‘small as possible’
Measuring performance – the regret

- A learner minimising the regret tries to collect as much reward as possible
- Sometimes you only care about finding the best action after \( n \) rounds
- Captured by the **simple regret**

\[
R_{\text{simple}}^n = \mathbb{E}[\Delta A_n]
\]

- Learner’s shooting for this objective are solving the **pure exploration** problem
- We don’t focus on this here though
Measuring performance – the regret

\[ R_n = n \mu^* - E \left[ \sum_{t=1}^{n} X_t \right] \]

- The regret is an expectation
- Does not take risk into account
Measuring performance – the regret

Let \( \Delta_i = \mu^* - \mu_i \) be the **suboptimality gap** for the \( i \)th arm and \( T_i(n) \) be the number of times arm \( i \) is played over all \( n \) rounds

**Lemma** \[ R_n = \sum_{i=1}^{K} \Delta_i \mathbb{E}[T_i(n)] \]

**Proof** Let \( \mathbb{E}_t[\cdot] = \mathbb{E}[\cdot | A_1, X_1, \ldots, X_{t-1}, A_t] \)

\[ R_n = n\mu^* - \mathbb{E} \left[ \sum_{t=1}^{n} X_t \right] = n\mu^* - \sum_{t=1}^{n} \mathbb{E}[\mathbb{E}_t[X_t]] = n\mu^* - \sum_{t=1}^{n} \mathbb{E}[\mu_{A_t}] \]

\[ = \sum_{t=1}^{n} \mathbb{E}[\Delta_{A_t}] = \mathbb{E} \left[ \sum_{t=1}^{n} \Delta_{A_t} \right] = \mathbb{E} \left[ \sum_{t=1}^{n} \sum_{i=1}^{K} 1(A_t = i) \Delta_i \right] \]

\[ = \mathbb{E} \left[ \sum_{i=1}^{K} \sum_{t=1}^{n} 1(A_t = i) \right] = \mathbb{E} \left[ \sum_{i=1}^{K} \Delta_i T_i(n) \right] = \sum_{i=1}^{K} \Delta_i \mathbb{E}[T_i(n)] \]
A simple policy: Explore-Then-Commit

1. Choose each action \( m \) times
2. Find the empirically best action \( I \in \{1, 2, \ldots, K\} \)
3. Choose \( A_t = I \) for all remaining rounds

In order to analyse this policy we need to bound the probability of committing to a suboptimal action.
A Crash Course in Concentration

Let $Z, Z_1, Z_2, \ldots, Z_n$ be a sequence of independent and identically distributed random variables with mean $\mu \in \mathbb{R}$ and variance $\sigma^2 < \infty$

\[
\text{empirical mean} = \hat{\mu}_n = \frac{1}{n} \sum_{t=1}^{n} Z_t
\]

How close is $\hat{\mu}_n$ to $\mu$?

Classical statistics says:

1. (law of large numbers) $\lim_{n \to \infty} \hat{\mu}_n = \mu$ almost surely
2. (central limit theorem) $\sqrt{n}(\hat{\mu}_n - \mu) \xrightarrow{d} \mathcal{N}(0, \sigma^2)$
3. (Chebyshev’s inequality) $\mathbb{P}(|\hat{\mu}_n - \mu| \geq \varepsilon) \leq \frac{\sigma^2}{n\varepsilon^2}$

We need something nonasymptotic and stronger than Chebysehv’s

Not possible without assumptions
Random variable $Z$ is $\sigma$-subgaussian if for all $\lambda \in \mathbb{R}$,

$$M_Z(\lambda) \doteq \mathbb{E}[\exp(\lambda Z)] \leq \exp\left(\frac{\lambda^2 \sigma^2}{2}\right)$$

**Lemma** If $Z, Z_1, \ldots, Z_n$ are independent and $\sigma$-subgaussian, then

- $aZ$ is $|a|\sigma$-subgaussian for any $a \in \mathbb{R}$
- $\sum_{t=1}^{n} Z_t$ is $\sqrt{n}\sigma$-subgaussian
- $\hat{\mu}_n$ is $n^{-1/2}\sigma$-subgaussian
A Crash Course in Concentration

**Theorem** If $Z_1, \ldots, Z_n$ are independent and $\sigma$-subgaussian, then

$$\mathbb{P} \left( \hat{\mu}_n \geq \sqrt{\frac{2\sigma^2 \log(1/\delta)}{n}} \right) \leq \delta$$

**Proof** We use Chernoff’s method. Let $\varepsilon > 0$ and $\lambda = \varepsilon n / \sigma^2$.

$$\mathbb{P} (\hat{\mu}_n \geq \varepsilon) = \mathbb{P} (\exp (\lambda \hat{\mu}_n) \geq \exp (\lambda \varepsilon))$$

$$\leq \mathbb{E} [\exp (\lambda \hat{\mu}_n)] \exp (-\lambda \varepsilon) \quad \text{(Markov’s)}$$

$$\leq \exp \left( \sigma^2 \lambda^2 / (2n) - \lambda \varepsilon \right)$$

$$= \exp \left( -n \varepsilon^2 / (2\sigma^2) \right)$$
A Crash Course in Concentration

- Which distributions are $\sigma$-subgaussian? **Gaussian, Bernoulli, bounded support.**
- And not: **exponential, power law**
- Comparing Chebyshev’s w. subgaussian bound:
  
  \[
  \text{Chebyshev’s: } \sqrt{\frac{\sigma^2}{n\delta}} \quad \text{Subgaussian: } \sqrt{\frac{2\sigma^2 \log(1/\delta)}{n}}
  \]
  
- Typically $\delta \ll 1/n$ in our use-cases

The results that follow hold when the distribution associated with each arm is $1$-subgaussian
Analyzing Explore-Then-Commit

- **Standard convention** Assume $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_K$
- Algorithms are symmetric and do not exploit this fact
- Means that first arm is optimal
- Remember, Explore-Then-Commit chooses each arm $m$ times
- Then commits to the arm with the largest payoff
- We consider only $K = 2$
Analysing Explore-Then-Commit

Step 1  Let $\hat{\mu}_i$ be the average reward after exploring

The algorithm commits to the wrong arm if

$$\hat{\mu}_2 \geq \hat{\mu}_1 \iff \hat{\mu}_2 - \mu_2 + \mu_1 - \hat{\mu}_1 \geq \Delta$$

Observation  $\hat{\mu}_1 - \mu_1 + \mu_2 - \hat{\mu}_2$ is $\sqrt{2/m}$-subgaussian

Step 2  The regret is

$$R_n = \mathbb{E} \left[ \sum_{t=1}^{n} \Delta_{A_t} \right] = \mathbb{E} \left[ \sum_{t=1}^{2m} \Delta_{A_t} \right] + \mathbb{E} \left[ \sum_{t=2m+1}^{n} \Delta_{A_t} \right]$$

$$= m\Delta + (n - 2m)\Delta \mathbb{P} \left( \text{commit to the wrong arm} \right)$$

$$= m\Delta + (n - 2m)\Delta \mathbb{P} \left( \hat{\mu}_2 - \mu_2 + \mu_1 - \hat{\mu}_1 \geq \Delta \right)$$

$$\leq m\Delta + n\Delta \exp \left( -\frac{m\Delta^2}{4} \right)$$
Analysing Explore-Then-Commit

\[ R_n \leq m\Delta + n\Delta \exp\left(-m\Delta^2/4\right) \]

(A) is monotone increasing in \( m \) while (B) is monotone decreasing in \( m \)

**Exploration/Exploitation dilemma**  Exploring too much (\( m \) large) then (A) is big, while exploring too little makes (B) large

Bound minimised by \( m = \left\lceil \frac{4}{\Delta^2} \log\left(\frac{n\Delta^2}{4}\right) \right\rceil \) leading to

\[ R_n \leq \Delta + \frac{4}{\Delta} \log\left(\frac{n\Delta^2}{4}\right) + \frac{4}{\Delta} \]
Analysing Explore-Then-Commit

Last slide: \( R_n \leq \Delta + \frac{4}{\Delta} \log \left( \frac{n\Delta^2}{4} \right) + \frac{4}{\Delta} \)

What happens when \( \Delta \) is very small?

\[
R_n \leq \min \left\{ n\Delta, \Delta + \frac{4}{\Delta} \log \left( \frac{n\Delta^2}{4} \right) + \frac{4}{\Delta} \right\}
\]
Does this figure make sense? Why is the regret largest when $\Delta$ is small, but not too small?

$$R_n \leq \min \left\{ n\Delta, \Delta + \frac{4}{\Delta} \log \left( \frac{n\Delta^2}{4} \right) + \frac{4}{\Delta} \right\}$$

Small $\Delta$ makes identification hard, but cost of failure is low.

Large $\Delta$ makes the cost of failure high, but identification easy.

Worst case is when $\Delta \approx \sqrt{1/n}$ with $R_n \approx \sqrt{n}$.
Limitations of Explore-Then-Commit

- Need advance knowledge of the horizon $n$
- Optimal tuning depends on $\Delta$
- Does not behave well with $K > 2$
- Issues can be overcome by using data to adapt the commitment time
- All variants of Explore-Then-Commit are at least a factor of 2 from being optimal
- Better approaches now exist, but Explore-Then-Commit is often a good place to start when analysing a bandit problem
Optimism principle

Optimism its the best
Way to see life
Visiting a new region

Shall I try local cuisine?

Optimist: Yes!

Pessimist: No!

Optimism leads to exploration, pessimism prevents it

Exploration is necessary, but how much?
Optimism Principle

- Let $\hat{\mu}_i(t) = \frac{1}{T_i(t)} \sum_{s=1}^{t} 1(A_s = i)X_s$
- Formalise the intuition using confidence intervals
- Optimistic estimate of the mean of arm = ‘largest value it could plausibly be’
- Suggests

$$\text{optimistic estimate} = \hat{\mu}_i(t - 1) + \sqrt{\frac{2 \log(1/\delta)}{T_i(t - 1)}}$$

- $\delta \in (0, 1)$ determines the level of optimism
Upper Confidence Bound Algorithm

1. Choose each action once

2. Choose the action maximising

\[ A_t = \arg\max_i \hat{\mu}_i(t - 1) + \sqrt{\frac{2 \log(t^3)}{T_i(t - 1)}} \]

3. Goto 2

Corresponds to \( \delta = 1/t^3 \)

This is quite a conservative choice. More on this later

Algorithm does not depend on horizon \( n \) (it is anytime)
Demonstration
Regret of UCB

**Theorem** The regret of UCB is at most

\[ R_n = O \left( \sum_{i : \Delta_i > 0} \left( \Delta_i + \frac{\log(n)}{\Delta_i} \right) \right) \]

Furthermore,

\[ R_n = O \left( \sqrt{Kn \log(n)} \right) \]

Bounds of the first kind are called **problem dependent** or **instance dependent**

Bounds like the second are called **distribution free** or **worst case**
Rewrite the regret \( R_n = \sum_{i=1}^{K} \Delta_i \mathbb{E}[T_i(n)] \)

Only need to show that \( \mathbb{E}[T_i(n)] \) is not too large for suboptimal arms
UCB Analysis

**Key insight** Arm $i$ is only played if its *index* is larger than the index of the optimal arm.

Need to show two things:

**(A)** The index of the optimal arm is larger than its actual mean with high probability.

**(B)** The index of suboptimal arms falls below the mean of the optimal arm after only a few plays.

\[
\gamma_i(t - 1) = \hat{\mu}_i(t - 1) + \sqrt{\frac{21 \log(t^3)}{T_i(t - 1)}}
\]

index of arm $i$ in round $t$
To make this intuition a reality we decompose the ‘pull-count’

\[ \mathbb{E}[T_i(n)] = \mathbb{E}\left[ \sum_{t=1}^{n} 1(A_t = i) \right] = \sum_{t=1}^{n} \mathbb{P}(A_t = i) \]

\[ = \sum_{t=1}^{n} \mathbb{P}(A_t = i \text{ and } (\gamma_1(t-1) \leq \mu_1 \text{ or } \gamma_i(t-1) \geq \mu_1)) \]

\[ \leq \sum_{t=1}^{n} \mathbb{P}(\gamma_1(t-1) \leq \mu_1) + \sum_{t=1}^{n} \mathbb{P}(A_t = i \text{ and } \gamma_i(t-1) \geq \mu_1) \]

- index of opt. arm too small?
- index of subopt. arm large?
UCB Analysis

We want to show that $\mathbb{P}(\gamma_1(t - 1) \leq \mu_1)$ is small.

Tempting to use the concentration theorem...

$$\mathbb{P}(\gamma_1(t - 1) \leq \mu_1) = \mathbb{P}\left(\hat{\mu}_1(t - 1) + \sqrt{\frac{2 \log(t^3)}{T_i(t - 1)}} \leq \mu_1\right) \leq \frac{1}{t^3}$$

What’s wrong with this? $T_i(t - 1)$ is a random variable!

$$\mathbb{P}\left(\hat{\mu}_1(t - 1) + \sqrt{\frac{2 \log(t^3)}{T_i(t - 1)}} \leq \mu_1\right) \leq \mathbb{P}\left(\exists s < t : \hat{\mu}_{1,s} + \sqrt{\frac{2 \log(t^3)}{s}} \leq \mu_1\right)$$

$$\leq \sum_{s=1}^{t-1} \mathbb{P}\left(\hat{\mu}_{1,s} + \sqrt{\frac{2 \log(t^3)}{s}} \leq \mu_1\right)$$

$$\leq \sum_{s=1}^{t-1} \frac{1}{t^3} \leq \frac{1}{t^2}.$$
\[ \sum_{t=1}^{n} \mathbb{P}(A_t = i \text{ and } \gamma_i(t - 1) \geq \mu_1) = \mathbb{E} \left[ \sum_{t=1}^{n} 1(A_t = i \text{ and } \gamma_i(t - 1) \geq \mu_1) \right] \]

\[ = \mathbb{E} \left[ \sum_{t=1}^{n} 1(A_t = i \text{ and } \hat{\mu}_i(t - 1) + \sqrt{\frac{6 \log(t)}{T_i(t - 1)}} \geq \mu_1) \right] \]

\[ \leq \mathbb{E} \left[ \sum_{t=1}^{n} 1(A_t = i \text{ and } \hat{\mu}_i(t - 1) + \sqrt{\frac{6 \log(n)}{T_i(t - 1)}} \geq \mu_1) \right] \]

\[ \leq \mathbb{E} \left[ \sum_{s=1}^{n} 1(\hat{\mu}_{i,s} + \sqrt{\frac{6 \log(n)}{s}} \geq \mu_1) \right] \]

\[ = \sum_{s=1}^{n} \mathbb{P} \left( \hat{\mu}_{i,s} + \sqrt{\frac{6 \log(n)}{s}} \geq \mu_1 \right) \]
Let $u = \frac{24 \log(n)}{\Delta_i^2}$. Then

$$\sum_{s=1}^{n} \mathbb{P} \left( \hat{\mu}_{i,s} + \sqrt{\frac{6 \log(n)}{s}} \geq \mu_1 \right) \leq u + \sum_{s=u+1}^{n} \mathbb{P} \left( \hat{\mu}_{i,s} + \sqrt{\frac{6 \log(n)}{s}} \geq \mu_1 \right)$$

$$\leq u + \sum_{s=u+1}^{n} \mathbb{P} \left( \hat{\mu}_{i,s} \geq \mu_i + \frac{\Delta_i}{2} \right)$$

$$\leq u + \sum_{s=u+1}^{\infty} \exp \left( -\frac{s\Delta_i^2}{8} \right)$$

$$\leq 1 + u + \frac{8}{\Delta_i^2}.$$
Combining the two parts we have

\[ \mathbb{E}[T_i(n)] \leq 3 + \frac{8}{\Delta_i^2} + \frac{24 \log(n)}{\Delta_i^2} \]

So the regret is bounded by

\[ R_n = \sum_{i: \Delta_i > 0} \Delta_i \mathbb{E}[T_i(n)] \leq \sum_{i: \Delta_i > 0} \left( 3\Delta_i + \frac{8}{\Delta_i} + \frac{24 \log(n)}{\Delta_i} \right) \]
Let $\Delta > 0$ be some constant to be chosen later

\[
R_n = \sum_{i: \Delta_i > 0} \Delta_i \mathbb{E}[T_i(n)] \leq n\Delta + \sum_{i: \Delta_i > \Delta} \Delta_i \mathbb{E}[T_i(n)] \\
\leq n\Delta + \sum_{i: \Delta_i > \Delta} \frac{\log(n)}{\Delta_i} \leq n\Delta + \frac{K \log(n)}{\Delta} \lesssim \sqrt{nK \log(n)}
\]

where in the last line we tuned $\Delta = \sqrt{K \log(n)/n}$
Improvements

• The constants in the algorithm/analysis can be improved quite significantly.

\[ A_t = \arg\max_i \hat{\mu}_i(t - 1) + \sqrt{\frac{2 \log(t)}{T_i(t - 1)}} \]

• With this choice:

\[ \lim_{n \to \infty} \frac{R_n}{\log(n)} = \sum_{i: \Delta_i > 0} \frac{2}{\Delta_i} \]

• The distribution-free regret is also improvable

\[ A_t = \arg\max_i \hat{\mu}_i(t - 1) + \sqrt{\frac{4}{T_i(t - 1)}} \log \left(1 + \frac{t}{K T_i(t - 1)}\right) \]

• With this index we save a log factor in the distribution free bound

\[ R_n = O(\sqrt{nK}) \]
Improvements

• **Warning**  Pushing the expected regret too hard results in high variance
Lower bounds

- Two kinds of lower bound: distribution free (worst case) and instance-dependent
- What could an instance-dependent lower bound look like?
- Algorithms that always choose a fixed action?
Worst case lower bound

**Theorem** For every algorithm and $n$ and $K \leq n$ there exists a $K$-armed Gaussian bandit such that $R_n \geq \sqrt{(K - 1)n/27}$

**Proof sketch**

- $\mu = (\Delta, 0, \ldots, 0)$
- $i = \arg\min_{i>1} \mathbb{E}[T_i(n)]$
- $\mathbb{E}[T_i(n)] \leq n/(K - 1)$
- $\mu' = (\Delta, 0, \ldots, 2\Delta, 0, \ldots, 0)$
- Envs. indistinguishable if $\Delta \approx \sqrt{K/n}$
- Suffers $n\Delta$ regret on one of them
Instance-dependent lower bounds

An algorithm is **consistent** on class of bandits $\mathcal{E}$ if $R_n = o(n)$ for all bandits in $\mathcal{E}$

**Theorem** If an algorithm is consistent for the class of Gaussian bandits, then

$$
\liminf_{n \to \infty} \frac{R_n}{\log(n)} \geq \sum_{i: \Delta_i > 0} \frac{2}{\Delta_i}
$$

- Consistency rules out stupid algorithms like the algorithm that always chooses a fixed action
- Consistency is asymptotic, so it is not surprising the lower bound we derive from it is asymptotic
- A non-asymptotic version of consistency leads to non-asymptotic lower bounds
What else is there?

- All kinds of variants of UCB for different noise models: Bernoulli, exponential families, heavy tails, Gaussian with unknown mean and variance,...
- A twist on UCB that replaces classical confidence bounds with Bayesian confidence bounds – offers empirical improvements
- Thompson sampling: each round sample mean from posterior for each arm, choose arm with largest
- All manner of twists on the setup: non-stationarity, delayed rewards, playing multiple arms each round, moving beyond expected regret (high probability bounds)
- **Different objectives**  Simple regret, measures of risk
The adversarial viewpoint

• Replace random rewards with an adversary
• At the start of the game the adversary secretly chooses losses $y_1, y_2, \ldots, y_n$ where $y_t \in [0, 1]^K$
• Learner chooses actions $A_t$ and suffers loss $y_tA_t$
• Regret is

$$R_n = \mathbb{E} \left[ \sum_{t=1}^{n} y_tA_t \right] - \min_{i} \sum_{t=1}^{n} y_{ti}$$

- learner’s loss
- loss of best arm

• Mission Make the regret small, regardless of the adversary
• There exists an algorithm such that

$$R_n \leq 2\sqrt{Kn}$$
The adversarial viewpoint

• The trick is in the definition of regret
• The adversary cannot be too mean

\[ R_n = \mathbb{E} \left[ \sum_{t=1}^{n} y_t A_t \right] - \min_i \sum_{t=1}^{n} y_{ti} \]

\( y = \begin{pmatrix} 1 & \cdots & 1 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 1 & \cdots & 1 \end{pmatrix} \)

• The following alternative objective is hopeless

\[ R'_n = \mathbb{E} \left[ \sum_{t=1}^{n} y_t A_t \right] - \sum_{t=1}^{n} \min_i y_{ti} \]

• Randomisation is crucial in adversarial bandits
Tackling the adversarial bandit

- Learner chooses distribution $P_t \in \Delta^K$ over the $K$ actions
- Samples $A_t \sim P_t$
- Observes $Y_t = y_t A_t$
- Expected regret is

$$R_n = \max_i \mathbb{E} \left[ \sum_{t=1}^{n} (y_t A_t - y_{ti}) \right] = \max_{p \in \Delta^K} \mathbb{E} \left[ \sum_{t=1}^{n} \langle P_t - p, y_t \rangle \right]$$

- This looks a lot like online linear optimisation on a simplex
- Only $y_t$ is not observed
- Idea is to find unbiased estimator $\hat{y}_t$
Tackling the adversarial bandit

Simple estimator of $y_t$ is the **importance weighted estimator**

$$\hat{y}_{ti} = \frac{1(A_t = i)y_{ti}}{P_{ti}}$$

We can see that $\mathbb{E}[\hat{y}_{ti} | A_1, Y_1, \ldots, A_{t-1}, Y_{t-1}] = y_{ti}$

$$R_n = \max_{p \in \Delta^K} \mathbb{E} \left[ \sum_{t=1}^{n} \langle P_t - p, y_t \rangle \right] = \max_{p \in \Delta^K} \mathbb{E} \left[ \sum_{t=1}^{n} \langle P_t - p, \hat{y}_t \rangle \right]$$

Now we have an online linear optimisation problem!
Tackling the adversarial bandit

Classic algorithm: $P_t = \arg\min_p \eta \sum_{s=1}^{t-1} \langle p, \hat{y}_t \rangle + F(p)$

where $\eta > 0$ is called the **learning rate** and $F$ is the regulariser

**Theorem** if $F(p) = \sum_i p_i \log(p_i) - p_i$ is the **negentropy** regulariser, then

$$\sum_{t=1}^{n} \langle P_t - p, \hat{y}_t \rangle \leq \frac{\log(K)}{\eta} + \frac{\eta}{2} \sum_{t=1}^{n} \sum_{i=1}^{K} P_{ti} \hat{y}_t^2$$

Taking the expectation and using the def. of $\hat{y}_t$,

$$R_n \leq \frac{\log(K)}{\eta} + \frac{\eta}{2} \mathbb{E} \left[ \sum_{t=1}^{n} \sum_{i=1}^{K} P_{ti} \left( \frac{1(A_t = i)y_{ti}}{P_{ti}} \right)^2 \right]$$

$$\leq \frac{\log(K)}{\eta} + \frac{\eta}{2} \mathbb{E} \left[ \sum_{t=1}^{n} \sum_{i=1}^{K} \mathbb{E}_t[1(A_t = i)] \right]$$

$$= \frac{\log(K)}{\eta} + \frac{\eta nK}{2} = \sqrt{2nK \log(K)}$$
Adversarial bandits

- Instance-dependence?
- Moving beyond expected regret (high probability bounds)
- Why bother with stochastic bandits?
- Best of both worlds? Bubeck and Slivkins (2012); Seldin and Lugosi (2017); Auer and Chiang (2016)
- **Big myth** Adversarial bandits do not address nonstationarity
Resources

- Book by Bubeck and Cesa-Bianchi (2012)
- Book by Cesa-Bianchi and Lugosi (2006)
- The Bayesian books by Gittins et al. (2011) and Berry and Fristedt (1985). Both worth reading.
- Our online notes: http://banditalgs.com
- We will soon release a 450 page book (“Bandit Algorithms” to be published by Cambridge)
Historical notes

• First paper on bandits is by Thompson (1933). He proposed an algorithm for two-armed Bernoulli bandits and hand-runs some simulations (Thompson sampling)
• Popularised enormously by Robbins (1952)
• Confidence bounds first used by Lai and Robbins (1985) to derive asymptotically optimal algorithm
• Adversarial bandits: Auer et al. (1995)
• Minimax optimal algorithm by Audibert and Bubeck (2009)


Random concentration failure

Let $X_1, X_2, \ldots$ be a sequence of independent and identically distributed standard Gaussian. For any $n$ we have

$$\mathbb{P} \left( \sum_{t=1}^{n} X_t \geq \sqrt{2n \log(1/\delta)} \right) \leq \delta$$

Want to show this can fail if $n$ is replaced by random variable $T$

Law of the iterated logaritm says that

$$\lim sup_{n \to \infty} \frac{\sum_{t=1}^{n} X_t}{\sqrt{2n \log \log(n)}} = 1 \quad \text{almost surely}$$

Let $T = \min \{ n : \sum_{t=1}^{n} X_t \geq \sqrt{2n \log(1/\delta)} \}$. Then $\mathbb{P} (T < \infty) = 1$ and

$$\mathbb{P} \left( \sum_{t=1}^{T} X_t \geq \sqrt{2T \log(1/\delta)} \right) = 1.$$ 

Contradiction! (works if $T$ is independent of $X_1, X_2, \ldots$ though)