Thompson Sampling

Jason Hartford

“randomly take action according to the probability you believe it is the optimal action” **- Thompson 1933**

History

Thompson was interested in the problem of **assigning treatments** to individuals…

Need to **explore** which of the two treatments is more successful, but also want to **minimize** the number of times you give patients the **suboptimal** treatment.

Likelihood that One Unknown Probability exceeds Another - Jstor
https://www.jstor.org/stable/pdf/2332286.pdf
by WR Thompson - 1933 - Cited by 1322 - Related articles
ON THE LIKELIHOOD THAT ONE UNKNOWN. PROBABILITY EXCEEDS ANOTHER IN VIEW. OF THE **EVIDENCE OF TWO SAMPLES.** BY WILLIAM R.

Fewer citations than a typical GAN paper 😐

ON THE LIKELIHOOD THAT ONE UNKNOWN PROBABILITY EXCEEDS ANOTHER IN VIEW OF THE EVIDENCE OF TWO SAMPLES.

By WILLIAM R. THOMPSON. From the Department of Pathology, Yale University.

Section 1.

In elaborating the relations of the present communication interest was not centred upon the interpretation of particular data, but grew out of a general interest in problems of research planning. From this point of view there can be no objection to the use of data, however meagre, as a guide to action required before more can be collected; although serious objection can otherwise be raised to argument based upon a small number of observations. Indeed, the fact that such objection can never be eliminated entirely—no matter how great the number of observations—suggested the possible value of seeking other modes of operation than that of taking a large number of observations before analysis or any attempt to direct our course. This problem is more general than that treated in *Section* 2, and is directly concerned with any case where probability criteria may be established by means of which we judge whether one mode of operation is *better* than another in some given sense or not.

History

Not much interest for 80 years until a number of authors noticed it performed well empirically [e.g. Graepel et al. 2010].

The first frequentist regret bounds that matched lower bounds up to a log factor were due to **Agrawal and Goyal** [2012, 2013a].

Web-Scale Bayesian Click-Through Rate Prediction for Sponsored Search Advertising in Microsoft's Bing Search Engine

Thore Graepel THOREG@MICROSOFT.COM
Joaquin Quiñonero Candela JOAQUINC@MICROSOFT.COM
Thomas Borchert TBORCHER@MICROSOFT.COM
Ralf Herbrich RHERB@MICROSOFT.COM
Microsoft Research Ltd., 7 J J Thomson Avenue, Cambridge CB3 0FB, UK

Abstract

We describe a new Bayesian click-through rate (CTR) prediction algorithm used for Sponsored Search in Microsoft's Bing search engine. The algorithm is based on a probit regression model that maps discrete or real-valued input features to probabilities. It maintains Gaussian beliefs over weights of the model and performs Gaussian online updates derived from approximate message passing. Scalability of the algorithm is ensured through a principled weight pruning procedure and an approximate parallel implementation. We discuss the challenges arising from evaluating and tuning the predictor as part of the complex system of sponsored search where the predictions made by the algorithm decide about future training sample composition. Finally, we show experimental results from the production system and compare to a calibrated Naïve Bayes algorithm.

of CTR prediction is absolutely crucial to Sponsored Search advertising because it impacts user experience, profitability of advertising and search engine revenue.

Recognising the importance of CTR estimation for online advertising, management at Bing/adCenter decided to run a competition to entice people across the company to develop the most accurate and scalable CTR predictor. The algorithm described in this publication tied for first place in the first competition and won the subsequent competition based on prediction accuracy. As a consequence, it was chosen to replace Bing's previous CTR prediction algorithm, a transition that was completed in the summer of 2009.

The paper makes three major contributions. First, it describes the Sponsored Search application scenario, the key role of CTR prediction in general, and the particular constraints derived from the task, including accuracy, calibration, scalability, dynamics, and exploration. Second, it describes a new Bayesian online learning algorithm for binary prediction, subsequently referred to as *adPredictor*. The algorithm is based on a generalised

Baselines have changed a lot since 2010…

Thompson Sampling Algorithm

Very intuitive algorithm which has been reinvented multiple times.

- Start with prior over parameters. *Think: a prior over the possible explanations for way the environment works.*
- Sample a particular set of parameters from the prior. *Think: pick one of those explanations.*
- Select arm $=$ arg max$_i$ reward$_i \,|\,$ parameters. *Think: maximize reward given your choice of explanation.*
- Observe reward and update posterior. *Think: update your model of the world*

Beta - Bernoulli Example

Algorithm 1: Thompson Sampling using Beta priors

For each arm $i = 1, \ldots, N$ set $S_i = 0, F_i = 0$.

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

  For each arm $i = 1, \ldots, N$, sample $\theta_i(t)$ from the Beta$(S_i + 1, F_i + 1)$ distribution.

  Play arm $i(t) := \arg\max_i \theta_i(t)$ and observe reward r_t.

  If $r_t = 1$, then $S_{i(t)} = S_{i(t)} + 1$, else $F_{i(t)} = F_{i(t)} + 1$.

end

Recall:

Prior: $\mathrm{Beta}(\alpha, \beta)$

Posterior:

Head: $\mathrm{Beta}(\alpha, \beta + 1)$

Tail: $\mathrm{Beta}(\alpha + 1, \beta)$

Regret bounds [Agrawal & Goyal 2013]

Theorem 1. *For the N-armed stochastic bandit problem, TS algorithm, using Beta priors has expected regret*

$$\mathbb{E}[\mathcal{R}(T)] \leq (1+\epsilon)\sum_{i=2}^{N}\frac{\ln T}{d(\mu_i,\mu_1)}\Delta_i + O(\frac{N}{\epsilon^2})$$

in time T, where $d(\mu_i,\mu_1) = \mu_i \log\frac{\mu_i}{\mu_1} + (1-\mu_i)\log\frac{(1-\mu_i)}{(1-\mu_1)}$. The big-Oh notation assumes $\mu_i, \Delta_i, i = 1,\ldots,N$ to be constants.

Regret bounds [Agrawal & Goyal 2013]

Theorem 1. *For the N-armed stochastic bandit problem, TS algorithm, using Beta priors has expected regret*

$$\mathbb{E}[\mathcal{R}(T)] \leq (1+\epsilon)\sum_{i=2}^{N}\frac{\ln T}{d(\mu_i,\mu_1)}\Delta_i + O(\frac{N}{\epsilon^2})$$

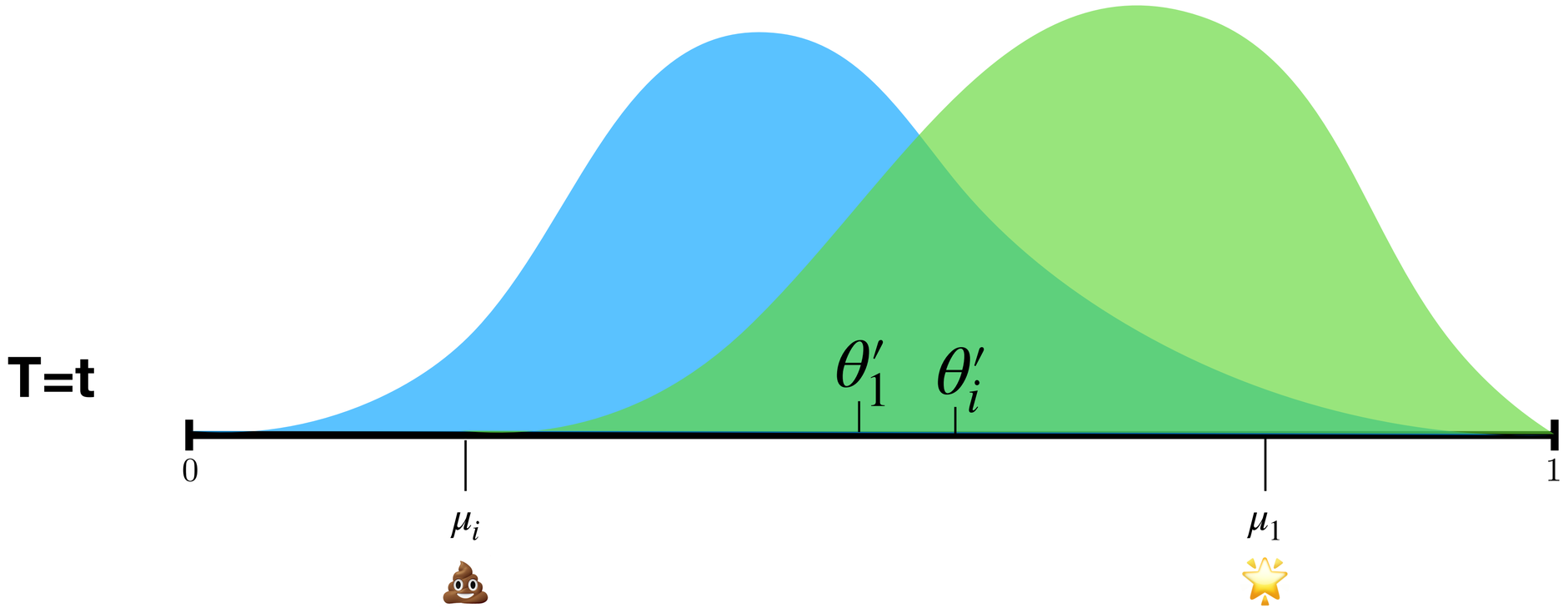
Theorem. Let $\Delta_i := \mu^* - \mu_i$ be suboptimality of arm i. If we choose $\delta \sim 1/T^2$:

UCB from last week

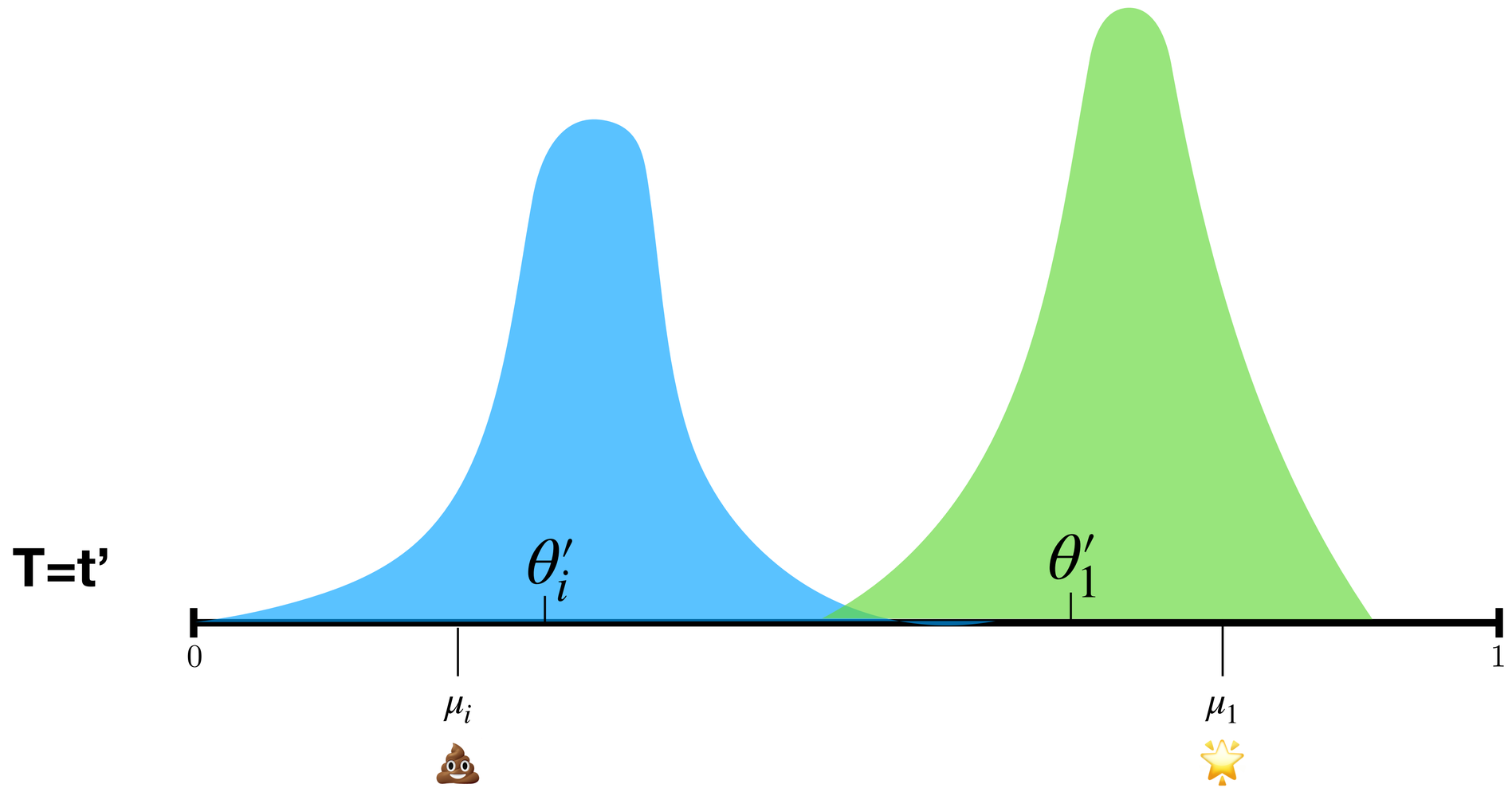
$$Reg(T) \leq C\sum_{i\in[K]}\Delta_i + \sum_{i:\Delta_i>0}\frac{C\log(T)}{\Delta_i}$$

Why might this be hard to prove?

Why might this be hard to prove?



Why might this be hard to prove?



Proof idea

Recall from last week: $R(T) = \sum_{i:\Delta_i>0} \Delta_i \mathbb{E}\left[k_i(T)\right]$

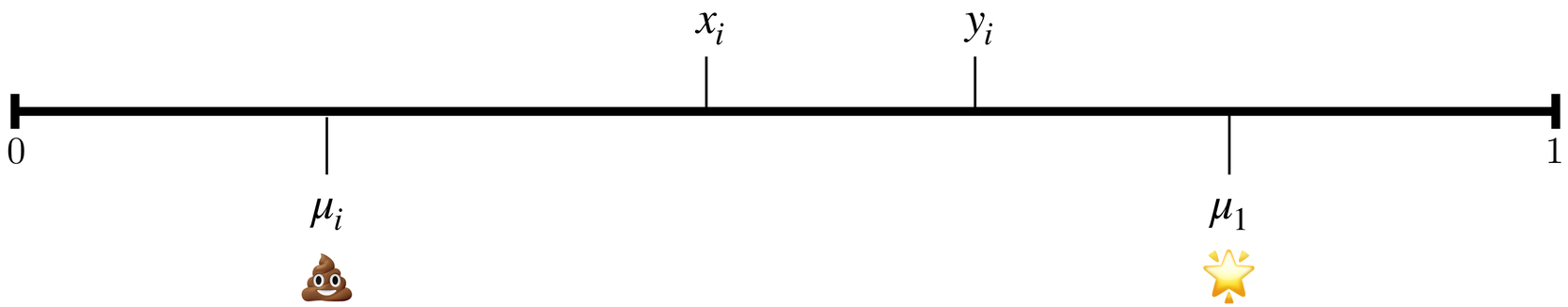
So we need to bound $\mathbb{E}\left[k_i(T)\right]$: the expected number of times each suboptimal arm is played.

Use two events to split up the expectation:

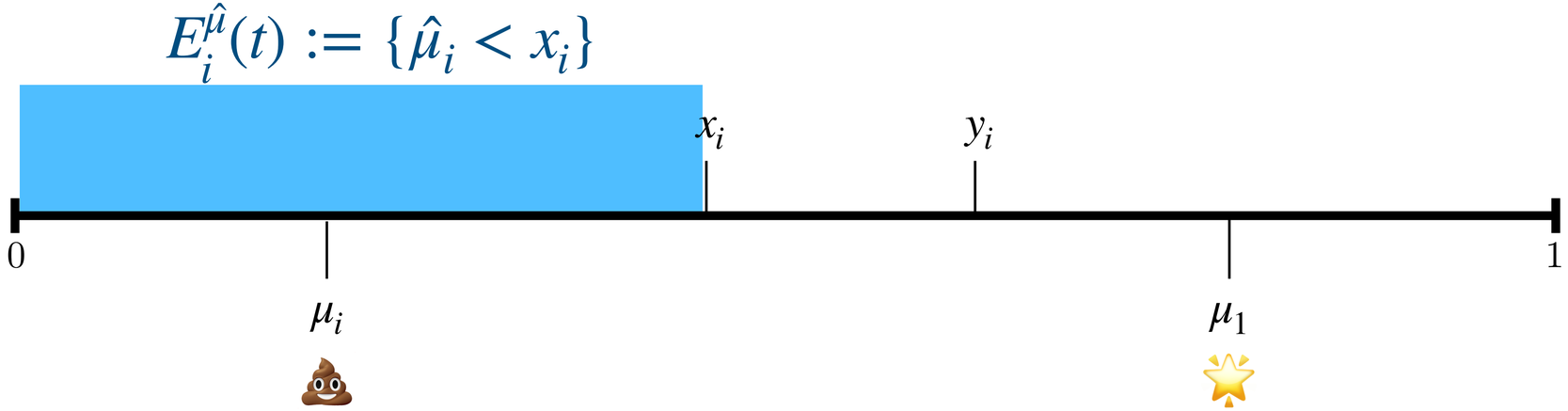
- $E_i^{\theta}(t)$ - the event that the sampled parameter is far from μ_i
- $E_i^{\hat{\mu}}(t)$ - the event that the estimated mean $\hat{\mu}_i$ is from from μ_i

Proof idea

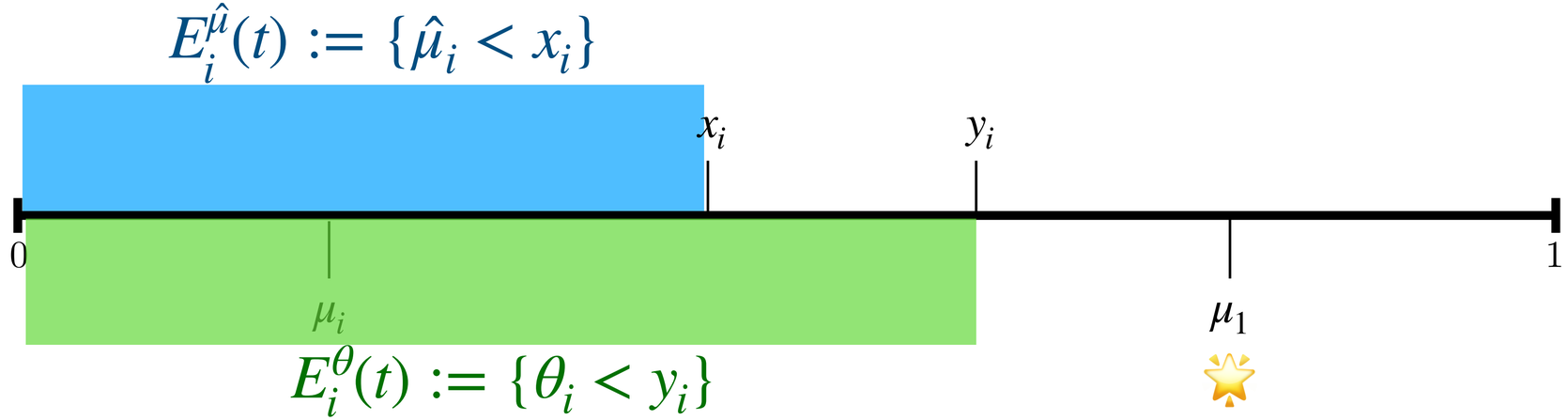
Proof idea



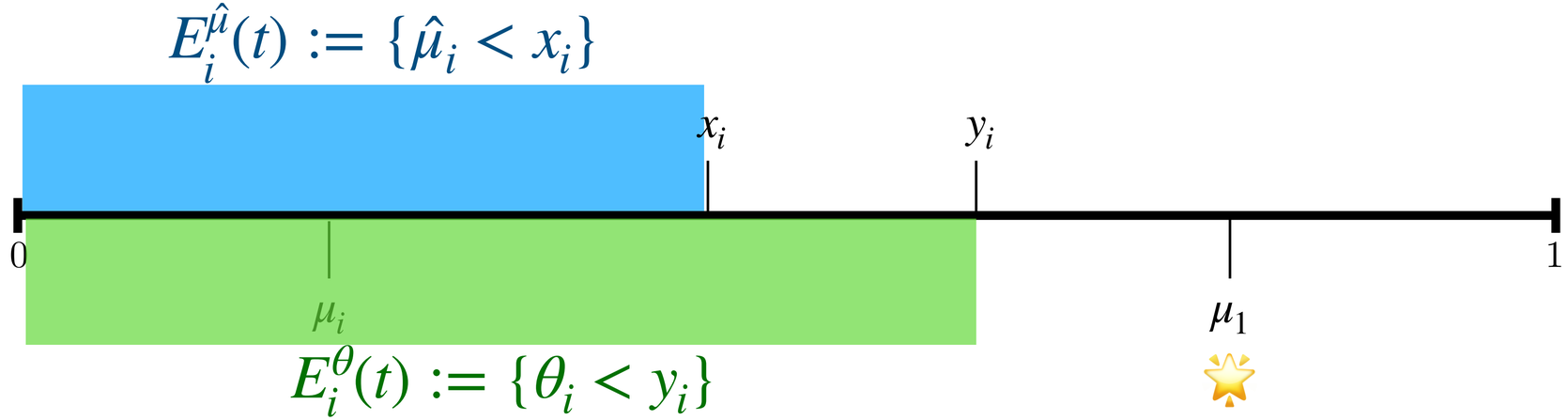
Proof idea



Proof idea



Proof idea



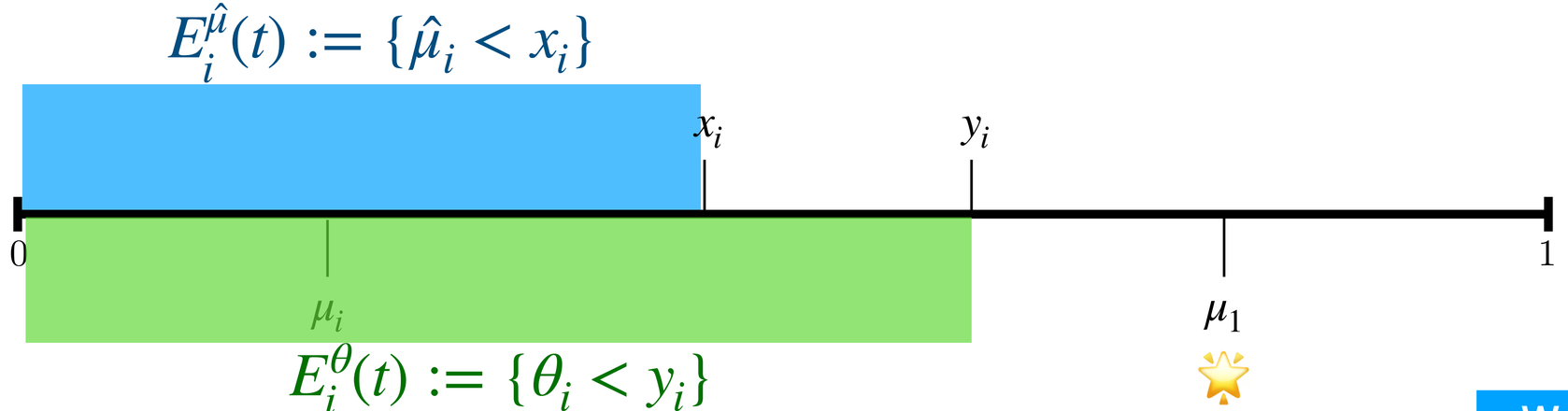
$$\mathbb{E}\left[k_i(T)\right] = \sum_{t=1}^{T} \Pr(i(t) = i) = \sum_{t=1}^{T} \Pr\left(i(t) = i, E_i^{\mu}(t), E_i^{\theta}(t)\right)$$

$$+ \sum_{t=1}^{T} \Pr\left(i(t) = i, E_i^{\mu}(t), \overline{E_i^{\theta}(t)}\right)$$

$$+ \sum_{t=1}^{T} \Pr\left(i(t) = i, \overline{E_i^{\mu}(t)}\right)$$

Number of times arm i is pulled

Proof idea



We'll show that…

$$\mathbb{E}\left[k_i(T)\right] = \sum_{t=1}^{T} \Pr(i(t) = i) = \sum_{t=1}^{T} \Pr\left(i(t) = i, E_i^{\mu}(t), E_i^{\theta}(t)\right)$$

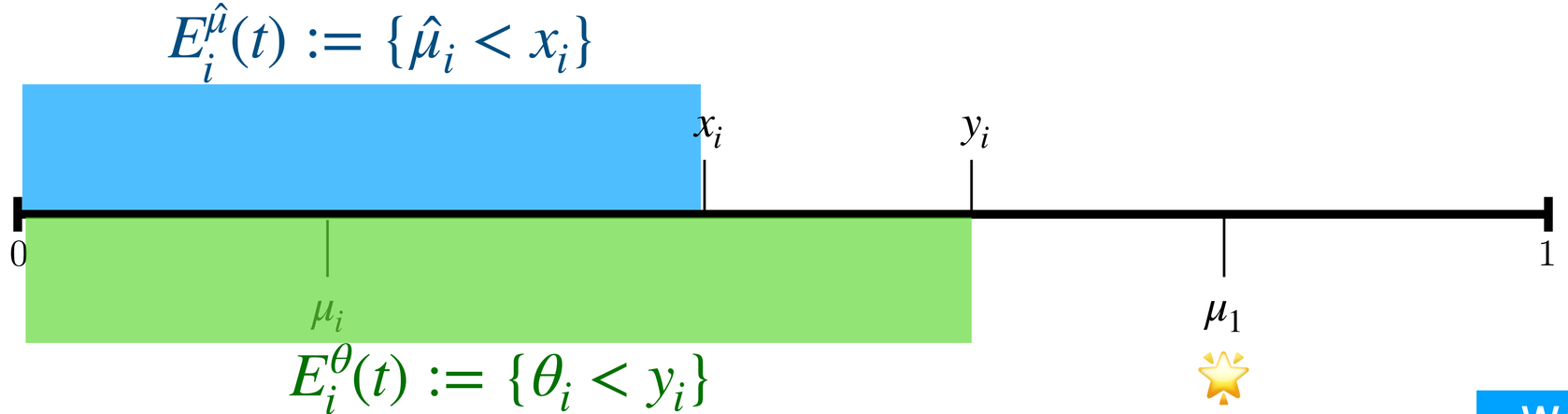
$$+ \sum_{t=1}^{T} \Pr\left(i(t) = i, E_i^{\mu}(t), \overline{E_i^{\theta}(t)}\right)$$

$$+ \sum_{t=1}^{T} \Pr\left(i(t) = i, \overline{E_i^{\mu}(t)}\right)$$

Number of times arm i is pulled

Bounded by linear function prob of playing 🌟

Proof idea



We'll show that…

$$\mathbb{E}[k_i(T)] = \sum_{t=1}^{T} \Pr(i(t) = i) = \sum_{t=1}^{T} \Pr\left(i(t) = i, E_i^{\mu}(t), E_i^{\theta}(t)\right)$$

Bounded by linear function prob of playing 🌟

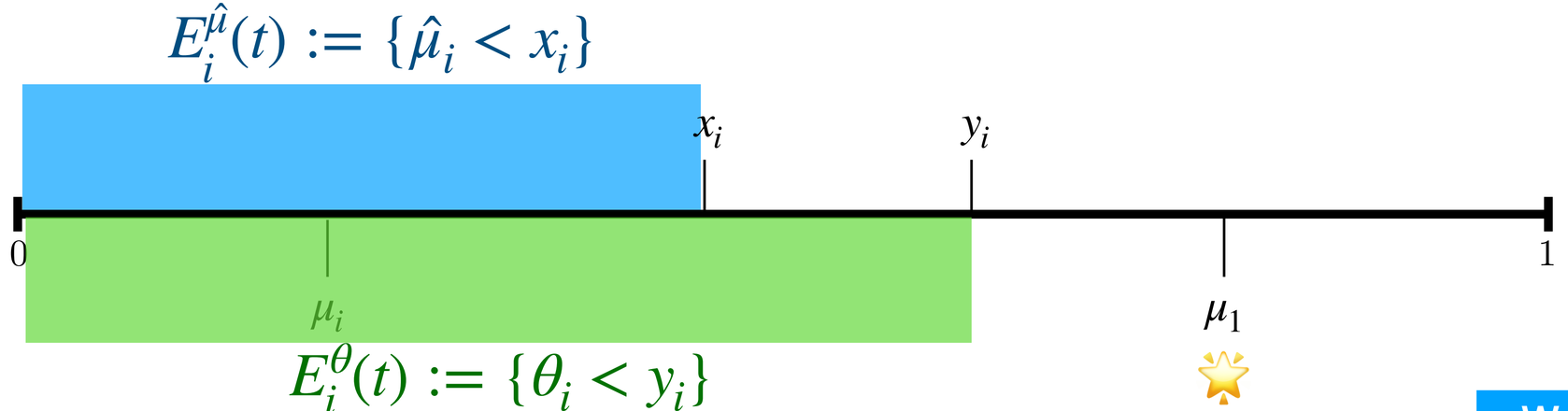
$$+ \sum_{t=1}^{T} \Pr\left(i(t) = i, E_i^{\mu}(t), \overline{E_i^{\theta}(t)}\right)$$

Rare once mean is concentrated

$$+ \sum_{t=1}^{T} \Pr\left(i(t) = i, \overline{E_i^{\mu}(t)}\right)$$

Number of times arm i is pulled

Proof idea



We'll show that…

$$\mathbb{E}[k_i(T)] = \sum_{t=1}^{T} \Pr(i(t) = i) = \sum_{t=1}^{T} \Pr\left(i(t) = i, E_i^{\mu}(t), E_i^{\theta}(t)\right)$$

Bounded by linear function prob of playing 🌟

$$+ \sum_{t=1}^{T} \Pr\left(i(t) = i, E_i^{\mu}(t), \overline{E_i^{\theta}(t)}\right)$$

Rare once mean is concentrated

$$+ \sum_{t=1}^{T} \Pr\left(i(t) = i, \overline{E_i^{\mu}(t)}\right)$$

Rare (using Chernoff)

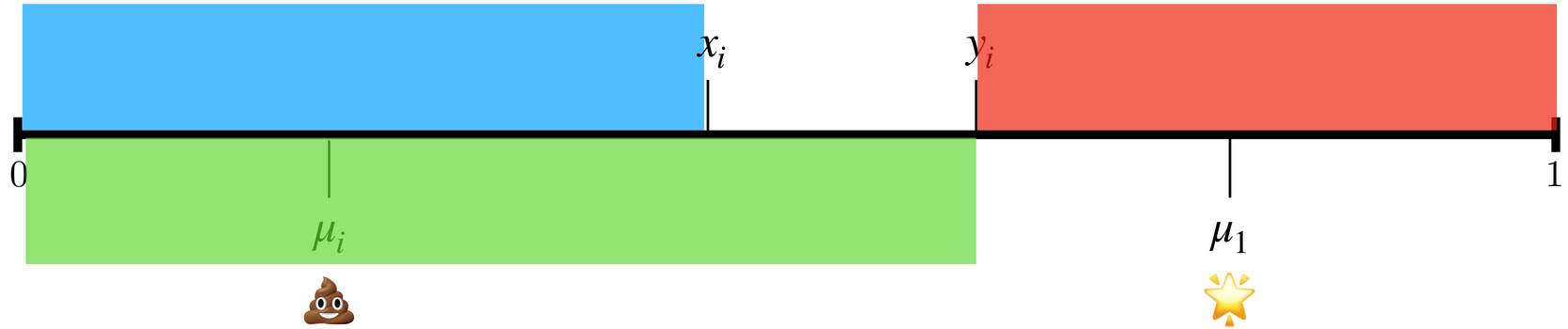
Number of times arm i is pulled

$$\sum_{t=1}^{T} \Pr\left(i(t) = i, E_i^{\mu}(t), E_i^{\theta}(t)\right)$$

$$+ \sum_{t=1}^{T} \Pr\left(i(t) = i, E_i^{\mu}(t), \overline{E_i^{\theta}(t)}\right)$$

$$+ \sum_{t=1}^{T} \Pr\left(i(t) = i, \overline{E_i^{\mu}(t)}\right)$$

Proof idea



Lemma 1. *For all* $t \in [1, T]$, *and* $i \neq 1$,

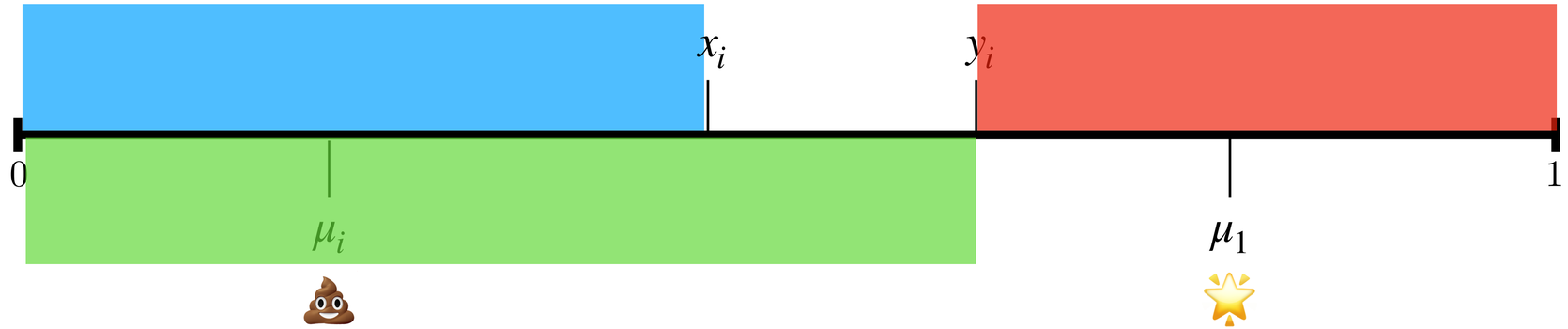
$$\Pr\left(i(t) = 💩, E_i^{\mu}(t), E_i^{\theta}(t) \,\middle|\, \mathcal{F}_{t-1}\right)$$

$$\leq \quad \frac{(1 - p_{i,t})}{p_{i,t}} \Pr\left(i(t) = 🌟, E_i^{\mu}(t), E_i^{\theta}(t) \,\middle|\, \mathcal{F}_{t-1}\right),$$

where $p_{i,t} = \Pr(\theta_1(t) > y_i | \mathcal{F}_{t-1})$.

$$\sum_{t=1}^{T} \Pr\left(i(t) = i, E_i^{\mu}(t), E_i^{\theta}(t)\right)$$
$$+ \sum_{t=1}^{T} \Pr\left(i(t) = i, E_i^{\mu}(t), \overline{E_i^{\theta}(t)}\right)$$
$$+ \sum_{t=1}^{T} \Pr\left(i(t) = i, \overline{E_i^{\mu}(t)}\right)$$

Proof idea



Lemma 1. *For all $t \in [1, T]$, and $i \neq 1$,*

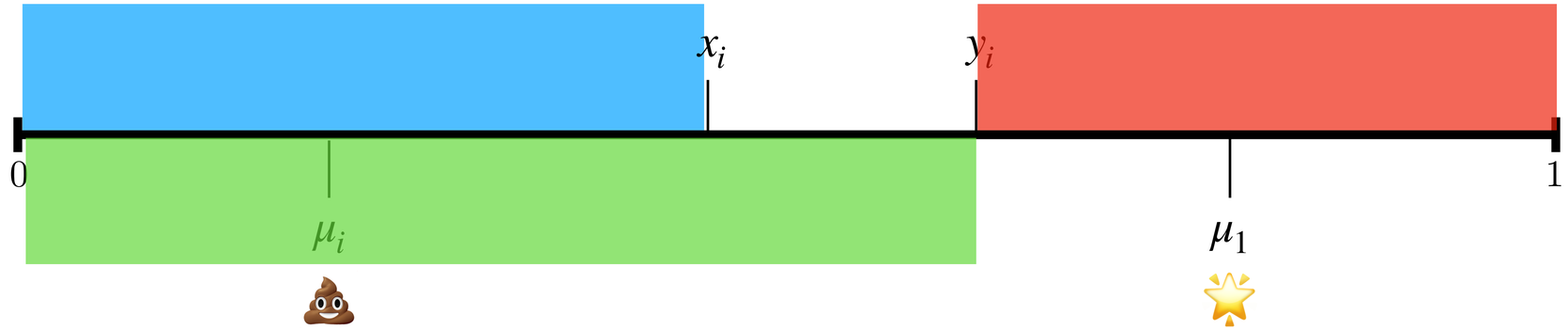
$$\Pr\left(i(t) = 💩, E_i^{\mu}(t), E_i^{\theta}(t) \mid \mathcal{F}_{t-1}\right)$$
$$\leq \frac{(1 - p_{i,t})}{p_{i,t}} \Pr\left(i(t) = 🌟, E_i^{\mu}(t), E_i^{\theta}(t) \mid \mathcal{F}_{t-1}\right),$$

where $p_{i,t} = \Pr(\theta_1(t) > y_i | \mathcal{F}_{t-1})$.

Probability of playing the optimal arm in the "nice" case

$$\sum_{t=1}^{T} \Pr\left(i(t) = i, E_i^{\mu}(t), E_i^{\theta}(t)\right)$$
$$+ \sum_{t=1}^{T} \Pr\left(i(t) = i, E_i^{\mu}(t), \overline{E_i^{\theta}(t)}\right)$$
$$+ \sum_{t=1}^{T} \Pr\left(i(t) = i, \overline{E_i^{\mu}(t)}\right)$$

Proof idea



Lemma 1. *For all $t \in [1, T]$, and $i \neq 1$,*

$$\Pr\left(i(t) = 💩, E_i^{\mu}(t), E_i^{\theta}(t) \mid \mathcal{F}_{t-1}\right)$$

$$\leq \frac{(1 - p_{i,t})}{p_{i,t}} \Pr\left(i(t) = 🌟, E_i^{\mu}(t), E_i^{\theta}(t) \mid \mathcal{F}_{t-1}\right),$$

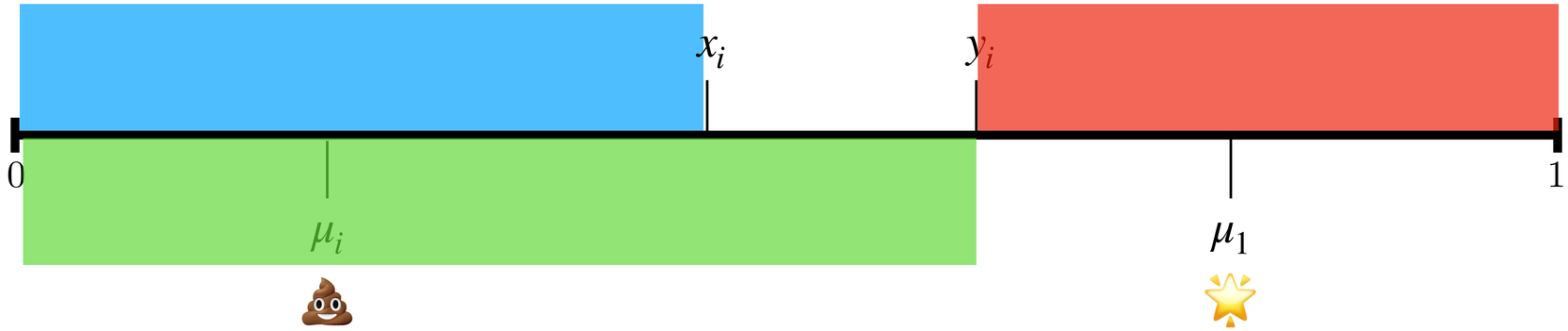
where $p_{i,t} = \Pr(\theta_1(t) > y_i | \mathcal{F}_{t-1})$.

Probability of playing the optimal arm in the “nice” case

Coefficient decreases exponentially fast with plays of opt

$$\sum_{t=1}^{T} \Pr\left(i(t) = i, E_i^{\mu}(t), E_i^{\theta}(t)\right)$$
$$+ \sum_{t=1}^{T} \Pr\left(i(t) = i, E_i^{\mu}(t), \overline{E_i^{\theta}(t)}\right)$$
$$+ \sum_{t=1}^{T} \Pr\left(i(t) = i, \overline{E_i^{\mu}(t)}\right)$$

Proof idea



Lemma 1. *For all $t \in [1, T]$, and $i \neq 1$,*

$$\Pr\left(i(t) = 💩, E_i^{\mu}(t), E_i^{\theta}(t) \mid \mathcal{F}_{t-1}\right)$$

$$\leq \frac{(1 - p_{i,t})}{p_{i,t}} \Pr\left(i(t) = 🌟, E_i^{\mu}(t), E_i^{\theta}(t) \mid \mathcal{F}_{t-1}\right),$$

where $p_{i,t} = \Pr(\theta_1(t) > y_i | \mathcal{F}_{t-1})$.

Probability of playing the optimal arm in the "nice" case

Coefficient decreases exponentially fast with plays of opt

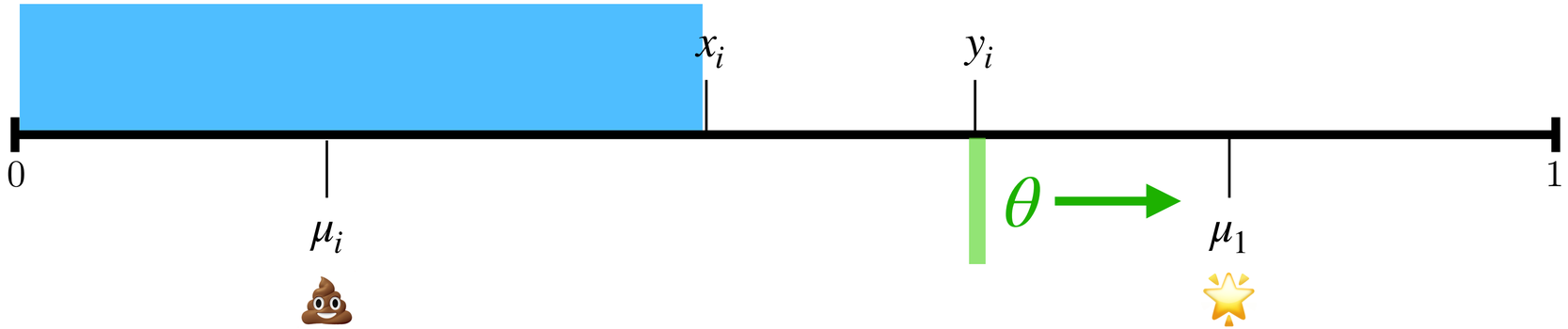
Acknowledgement. We thank Emil Jeřábek for telling us about his estimates of partial binomial sums. We also thank MathOverflow for connecting us with Emil.

$$\sum_{t=1}^{T} \Pr\left(i(t) = i, E_i^{\mu}(t), E_i^{\theta}(t)\right)$$

$$+ \sum_{t=1}^{T} \Pr\left(i(t) = i, E_i^{\mu}(t), \overline{E_i^{\theta}(t)}\right)$$

$$+ \sum_{t=1}^{T} \Pr\left(i(t) = i, \overline{E_i^{\mu}(t)}\right)$$

Proof idea



$$\sum_{t=1}^{T} \Pr\left(i(t) = i, \overline{E_i^{\theta}(t)}, E_i^{\mu}(t)\right) \leq \frac{\log(T)}{d(x_i, y_i)} + 1$$

Sketch: - given that $\hat{\mu}_i$ is less than x_i, we can only sample θ greater than y_i before the posterior concentrates around its mean.

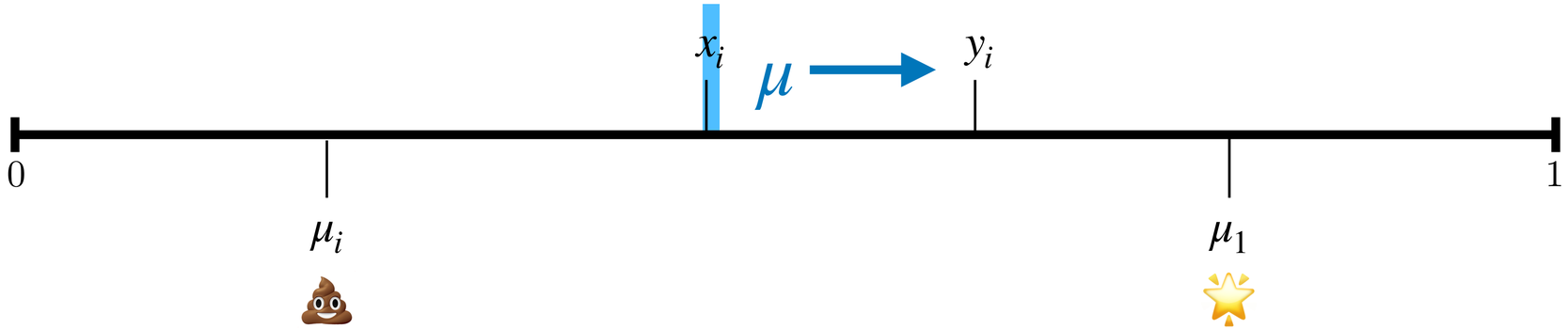
It takes at most $\frac{\log T}{d(x_i, y_i)}$ samples for this to happen and thereafter θ exceeds y with probability $\frac{1}{T}$.

$$\sum_{t=1}^{T} \Pr\left(i(t) = i, E_i^{\mu}(t), E_i^{\theta}(t)\right)$$

$$+ \sum_{t=1}^{T} \Pr\left(i(t) = i, E_i^{\mu}(t), \overline{E_i^{\theta}(t)}\right)$$

$$+ \sum_{t=1}^{T} \Pr\left(i(t) = i, \overline{E_i^{\mu}(t)}\right)$$

Proof idea



$$\sum_{t=1}^{T} \Pr\left(i(t) = i, \overline{E_i^{\mu}(t)}\right) \leq \frac{1}{d(x_i, \mu_i)} + 1.$$

For this one uses the fact that Chernoff implies this probability decreases exponentially in the number of times that the arm is played.

Putting it together

We set x_i and y_i to get appropriate bounds. For some $\epsilon \in [0,1]$ let $d(x_i, \mu_1) = d(\mu_i, \mu_1)/(1+\epsilon)$ and $d(x_i, y_i) = d(\mu_i, \mu_1)/(1+\epsilon)^2$.

With this, you get:

$$\mathbb{E}[k_i(T)] \leq O(1) + (1+\epsilon)^2 \log(T)/d(\mu_i, \mu_1) + O(1/\epsilon^2)$$

Putting it together

We set x_i and y_i to get appropriate bounds. For some $\epsilon \in [0,1]$ let $d(x_i, \mu_1) = d(\mu_i, \mu_1)/(1+\epsilon)$ and $d(x_i, y_i) = d(\mu_i, \mu_1)/(1+\epsilon)^2$.

With this, you get:

$$\mathbb{E}[k_i(T)] \leq O(1) + (1+\epsilon)^2 \log(T)/d(\mu_i, \mu_1) + O(1/\epsilon^2)$$

Ugly looking constant from the first sum

Putting it together

We set x_i and y_i to get appropriate bounds. For some $\epsilon \in [0,1]$ let $d(x_i, \mu_1) = d(\mu_i, \mu_1)/(1+\epsilon)$ and $d(x_i, y_i) = d(\mu_i, \mu_1)/(1+\epsilon)^2$.

With this, you get:

$$\mathbb{E}[k_i(T)] \leq O(1) + (1+\epsilon)^2 \log(T)/d(\mu_i, \mu_1) + O(1/\epsilon^2)$$

Ugly looking constant from the first sum

Second sum: “cost of exploration”

Putting it together

We set x_i and y_i to get appropriate bounds. For some $\epsilon \in [0,1]$ let $d(x_i, \mu_1) = d(\mu_i, \mu_1)/(1+\epsilon)$ and $d(x_i, y_i) = d(\mu_i, \mu_1)/(1+\epsilon)^2$.

With this, you get:

$$\mathbb{E}[k_i(T)] \leq O(1) + (1+\epsilon)^2 \log(T)/d(\mu_i, \mu_1) + O(1/\epsilon^2)$$

Ugly looking constant from the first sum

Second sum: "cost of exploration"

Third sum

Application

Requirements.

How do we use these ideas in more complicated settings (e.g. contextual bandits, reinforcement learning)?

In order to apply Thompson Sampling, you need two things:

- Proper **uncertainty estimates** of the parameters of your model.
- A way of **updating posterior** given new data.

This is straightforward with conjugate models (e.g. GPs work well).

What about neural nets? Isn't **Bayesian Deep Learning** all the rage?

Deep reinforcement learning with Thompson Sampling ideas

There is a long history of research in Bayesian neural networks that never quite became mainstream practice [37, 43]. Recently, Bayesian deep learning has experienced a resurgence of interest with a myriad of approaches for uncertainty quantification in fixed datasets and also sequential decision problems [29, 11, 20, 47]. In this paper we highlight the surprising fact that many of these well-cited and popular methods for uncertainty estimation in deep learning can be poor choices for sequential decision problems. We show that this disconnect is more than a technical detail, but a serious shortcoming that can lead to arbitrarily poor performance. We support our claims by a series of simple lemmas for simple environments, together with experimental evidence in more complex settings.

Randomized Prior Functions for Deep Reinforcement Learning [Osband et al. NeurIPS 2018]

Paper makes two contributions:

1. Discusses the limitations of popular approaches to uncertainty quantification.

2. Proposes a bootstrap approach with a prior function.

Problems with popular Bayesian deep learning approaches

- **Dropout as posterior approximation**: dropout rate, p, doesn't depend on the data so posterior doesn't concentrate… improvements that make p data-dependent suffer from variational critique.
- **Variational inference and Bellman error:** using the wrong loss (MSE) does not propagate uncertainty correctly. *(Comment: weird critique)*
- **Distributional RL:** models distribution over rewards, $\mathbb{P}(r \mid \theta)$, not distribution over model parameters, θ.

Randomized prior functions for deep ensembles

- Standard bootstrap is easy to implement: train K neural networks in parallel random samples (with replacement) of your data. Use appropriately scaled variance of predictions as an estimate of confidence intervals.

- Doesn't give a way of introducing a prior. Proposed solution:

Algorithm 1 Randomized prior functions for ensemble posterior.

Require: Data $\mathcal{D} \subseteq \{(x,y)|x \in \mathcal{X}, y \in \mathcal{Y}\}$, loss function $\mathcal{L}$, neural model $f_\theta : \mathcal{X} \to \mathcal{Y}$,
Ensemble size $K \in \mathbb{N}$, noise procedure `data_noise`, distribution over priors $\mathcal{P} \subseteq \{\mathbb{P}(p)|p : \mathcal{X} \to \mathcal{Y}\}$.

for $k = 1, .., K$ **do**
  initialize $\theta_k \sim$ Glorot initialization [23].
  form $\mathcal{D}_k =$ `data_noise`$(\mathcal{D})$ (e.g. Gaussian noise or bootstrap sampling [50]).
  sample prior function $p_k \sim \mathcal{P}$.
  optimize $\nabla_{\theta|\theta=\theta_k} \mathcal{L}(f_\theta + p_k; \mathcal{D}_k)$ via ADAM [28].
return ensemble $\{f_{\theta_k} + p_k\}_{k=1}^K$.

https://sites.google.com/view/randomized-prior-nips-2018/

Results

Algorithm 2 `learn_bootstrapped_dqn_with_prior`

Agent:	$\theta_1, .., \theta_K$	trainable network parameters
	$p_1, .., p_K$	fixed prior functions
	$\mathcal{L}_\gamma(\theta{=}\cdot; \theta^-{=}\cdot, p{=}\cdot, \mathcal{D}{=}\cdot)$	TD error loss function
	`ensemble_buffer`	replay buffer of K-parallel perturbed data
Updates:	$\theta_1, .., \theta_K$	agent value function estimate

1: **for** k in $(1, \ldots, K)$ **do**
2: | Data $\mathcal{D}_k \leftarrow$ `ensemble_buffer[k].sample_minibatch()`
3: | optimize $\nabla_{\theta|\theta=\theta_k} \mathcal{L}(\theta; \theta_k, p_k, \mathcal{D}_k)$ via ADAM [28].

Figure 6: The prior network qualitatively changes behavior on Montezuma's revenge.

https://sites.google.com/view/randomized-prior-nips-2018/

Results

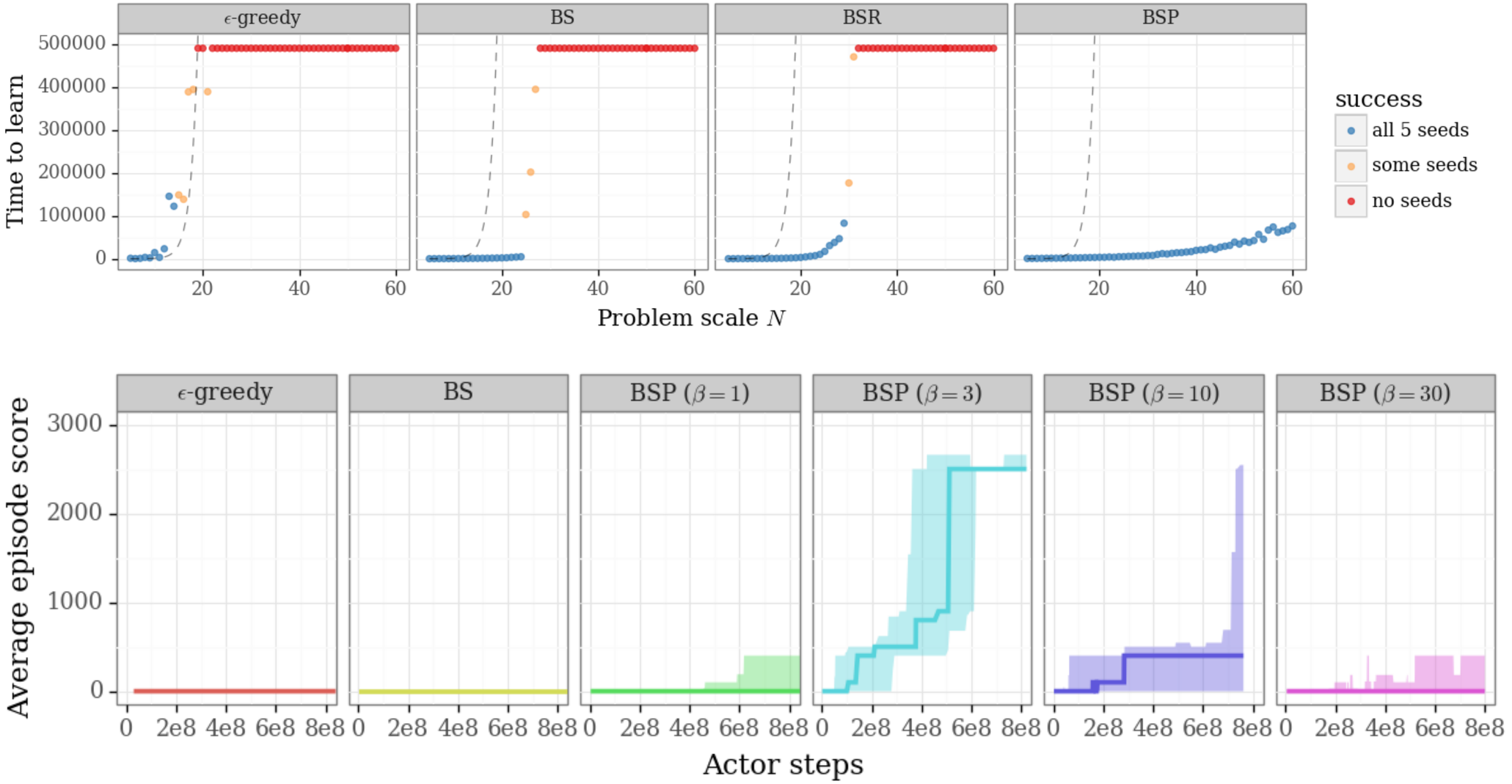


Figure 6: The prior network qualitatively changes behavior on Montezuma's revenge.

Things I didn’t get to…

- Bayesian regret vs frequentist regret. See https://tor-lattimore.com/downloads/book/book.pdf chapter 34 for details.

- Bayesian regret bounds - easier to prove and look very much like the UCB proof from last week. See https://tor-lattimore.com/downloads/book/book.pdf chapter 36 for details.

- The role of priors in Bayesian bandits. In supervised learning, priors are overwhelmed by the data eventually; in online learning a bad prior can mean you don’t explore a good action (not such an issue with typical choices of priors - e.g. uniform).