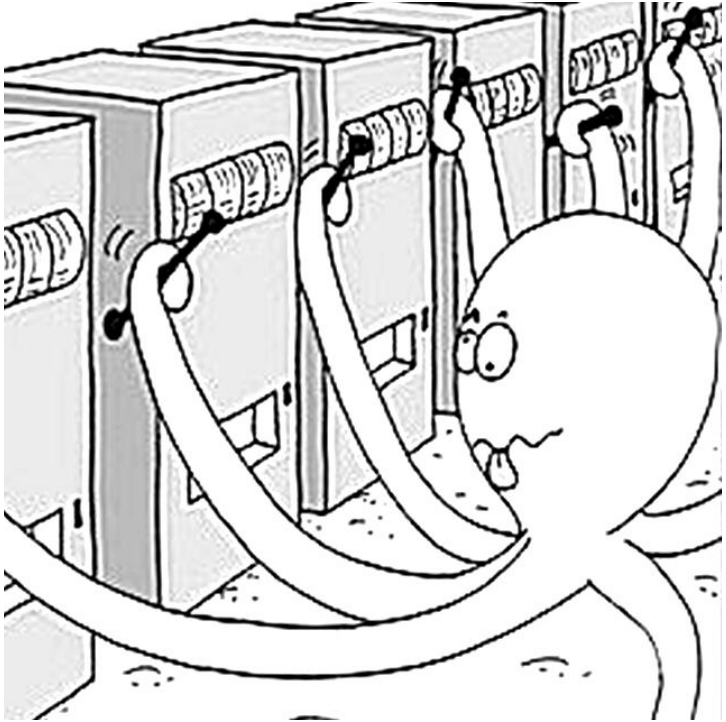
Multiarmed bandits

Thompson Sampling team

МФТИ x

General statement of the problem of multi-armed bandits

- The agent interacts with the environment by choosing actions (hands) from a set of available ones
- Each action brings a random reward with an unknown distribution
- The agent's goal is to maximize the total reward over a certain number of steps, balancing between exploration (gathering information about distributions) and exploitation (using current knowledge to maximize reward)
- The main metric is regret (the difference between the reward of the optimal action and the chosen action)



Out-of-context bandits

Formally:

- At each step t, the agent chooses an action a_t from N options.
- The reward $r(t)$ is chosen from a distribution with mathematical expectation μ_{a_t}.
- Optimal action: $a^* = \arg\max_a \mu_a$.
- Regret at step t: $\Delta_t = \mu_{a^*} - \mu_{a_t}$.
- Goal: minimize $R(T) = \sum_{t=1}^{T} \Delta_t$.

Features:

- No additional information (context) before choosing an action
- Each action has a fixed but unknown reward distribution

Examples of algorithms:

- ϵ-greedy
- UCB (Upper Confidence Bound)
- Thompson Sampling

Contextual Bandits

Features:

- Before choosing an action, the agent receives context - a feature vector that affects the reward
- The reward distribution depends on the context (e.g., linearly)
- The context can be adaptive (depends on the agent's previous actions)

Formally:

- At step t:
  - The agent receives context vectors $\{\mathbf{b}_i(t)\} \in R^d$, $i \in [1, N]$ for each action
  - Expected reward of action i: $E[r_i(t)] = \mathbf{b}_i(t)^\top \mu$, where $\mu \in R^d$ is an unknown parameter
  - The agent chooses action a_t and receives reward $r_{a_t}(t)$
- Regret: $\Delta_t = \max_i(\mathbf{b}_i(t)^\top \mu) - \mathbf{b}_{a_t}(t)^\top \mu$
- Goal: minimize $R(T) = \sum_{t=1}^{T} \Delta_t$

Examples of algorithms:

- LinUCB (Linear Upper Confidence Bound)
- LinTS (Linear Thompson Sampling)

Comparison of algorithms

Epsilon-Greedy-Constant

Initialize, for $a = 1$ to k:

$Q(a) \leftarrow 0$

$N(a) \leftarrow 0$

Loop forever:

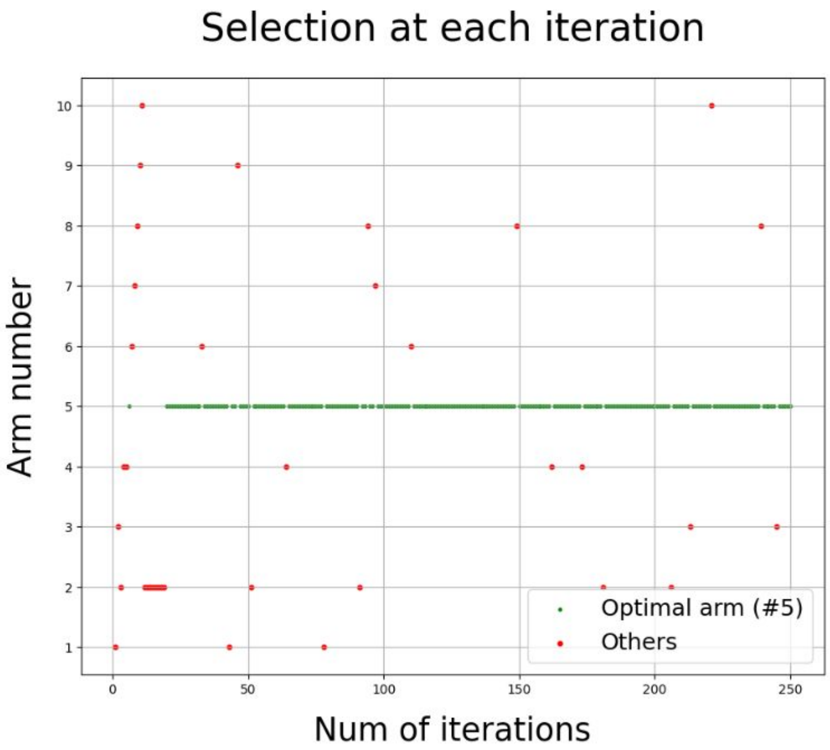
$$A \leftarrow \begin{cases} \arg\max_a Q(a) & \text{with probability } 1-\varepsilon \quad \text{(breaking ties randomly)} \\ \text{a random action} & \text{with probability } \varepsilon \end{cases}$$

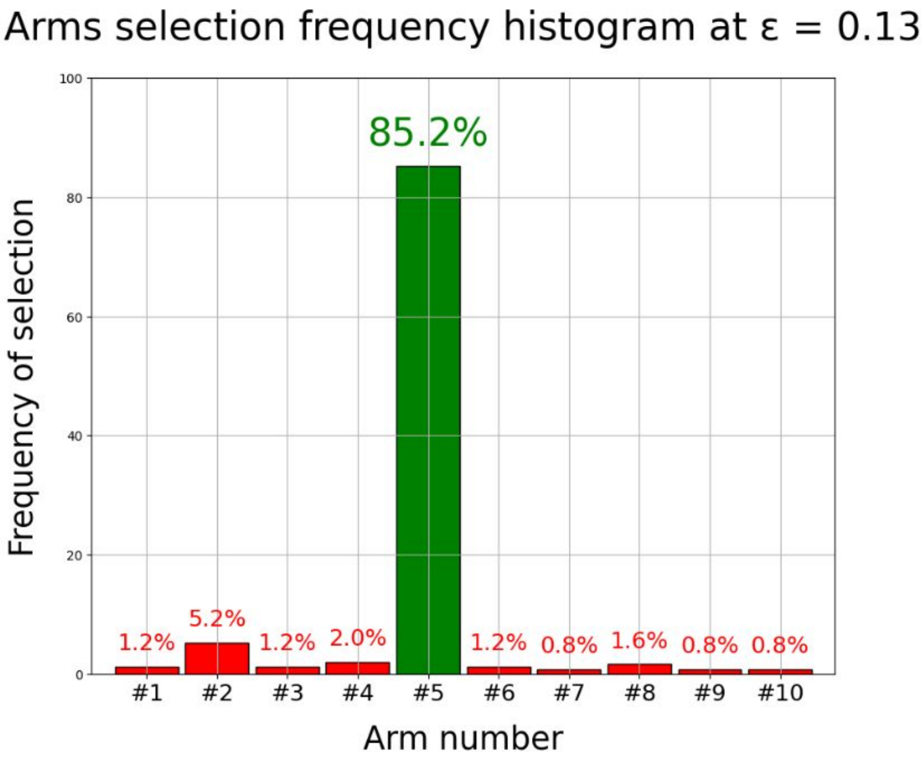
$R \leftarrow bandit(A)$

$N(A) \leftarrow N(A) + 1$

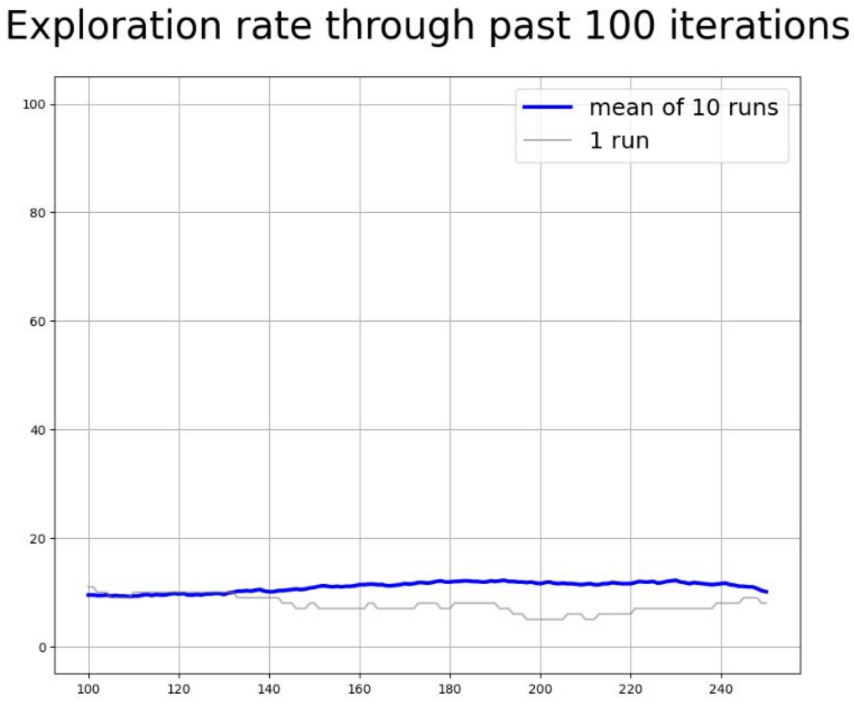
$Q(A) \leftarrow Q(A) + \frac{1}{N(A)}\left[R - Q(A)\right]$

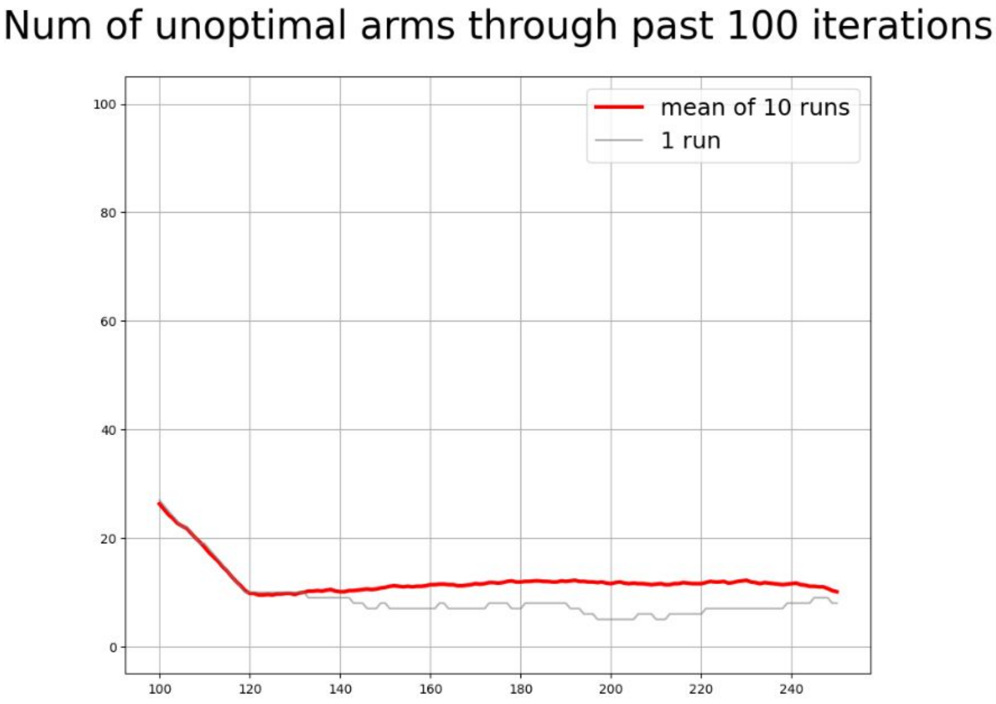
Epsilon-Greedy-Constant



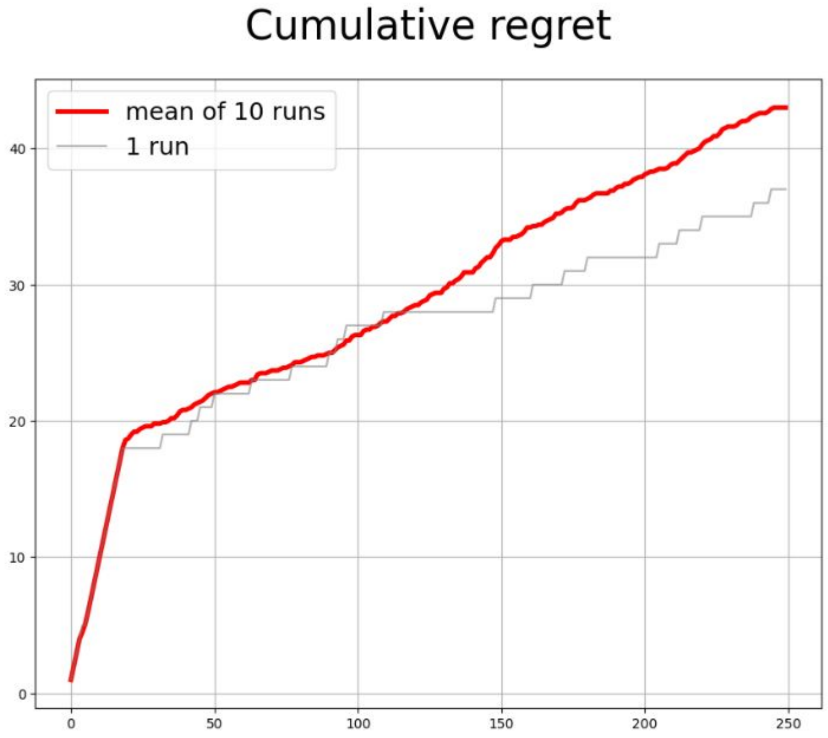


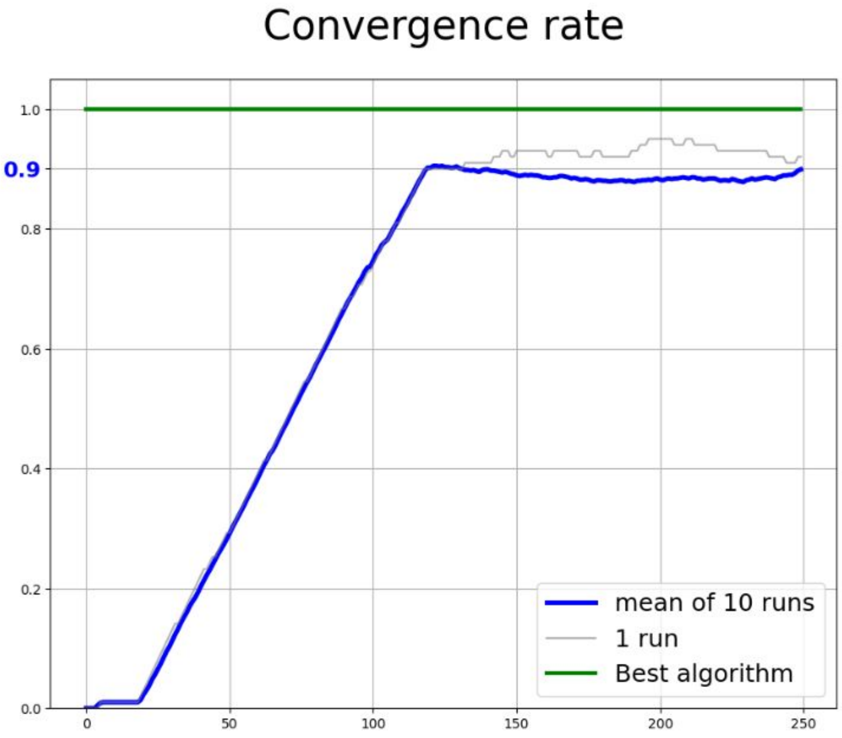
Epsilon-Greedy-Constant





Epsilon-Greedy-Constant





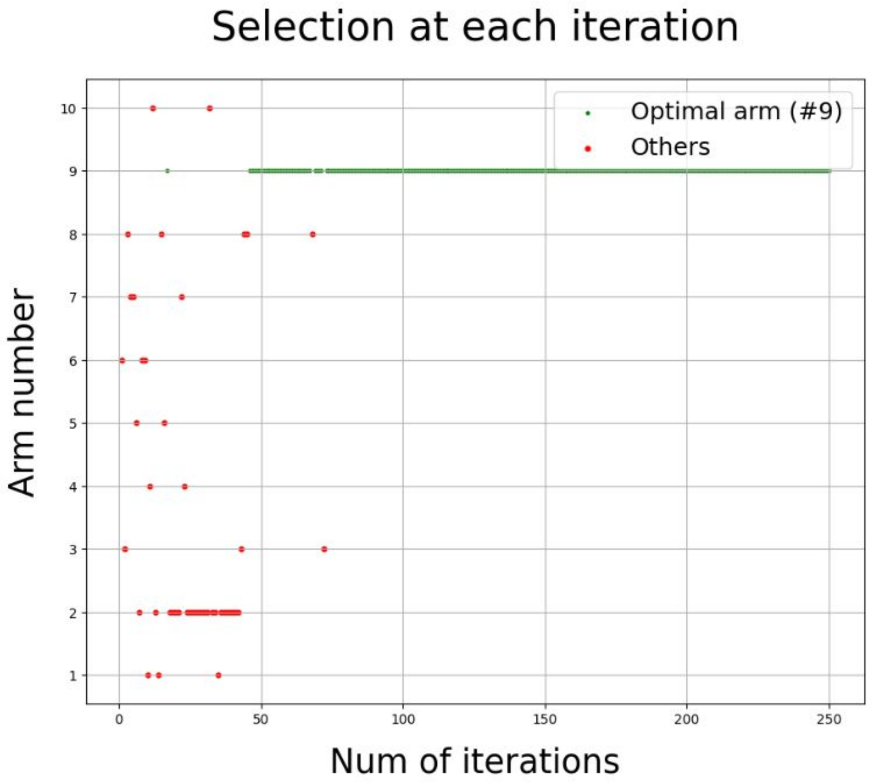
Value-Difference-Based-Epsilon

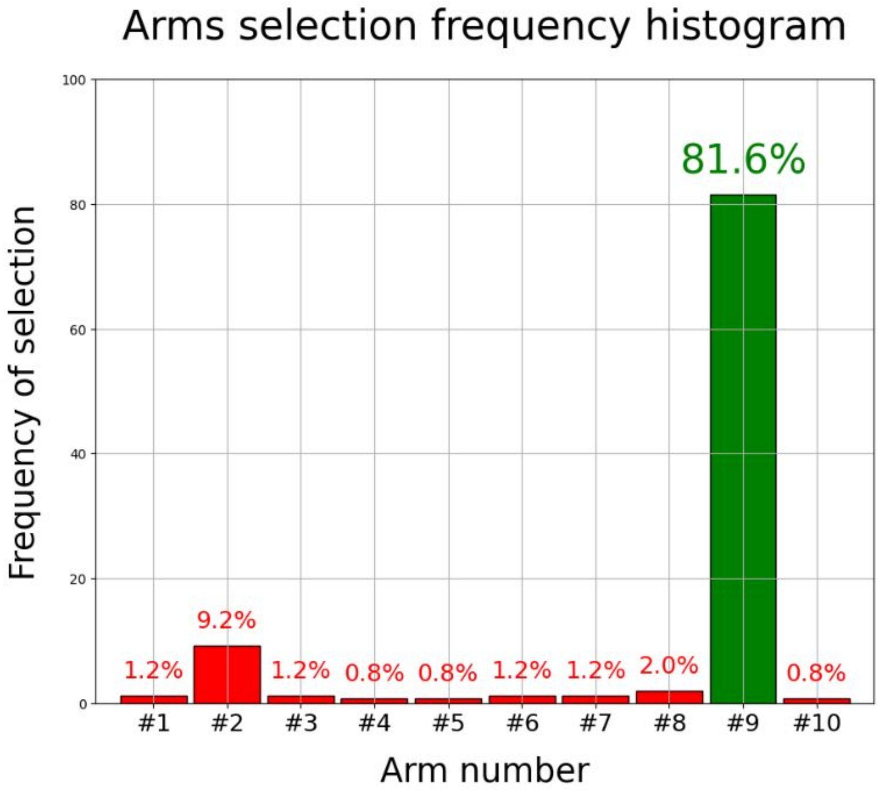
$$\pi(s) = \begin{cases} \text{random action from } \mathcal{A}(s) & \text{if } \xi < \varepsilon \\ \operatorname{argmax}_{a \in \mathcal{A}(s)} Q(s,a) & \text{otherwise,} \end{cases}$$

$$\begin{aligned} f(s,a,\sigma) &= \left| \frac{e^{\frac{Q_t(s,a)}{\sigma}}}{e^{\frac{Q_t(s,a)}{\sigma}} + e^{\frac{Q_{t+1}(s,a)}{\sigma}}} - \frac{e^{\frac{Q_{t+1}(s,a)}{\sigma}}}{e^{\frac{Q_t(s,a)}{\sigma}} + e^{\frac{Q_{t+1}(s,a)}{\sigma}}} \right| \\ &= \frac{1 - e^{\frac{-|Q_{t+1}(s,a) - Q_t(s,a)|}{\sigma}}}{1 + e^{\frac{-|Q_{t+1}(s,a) - Q_t(s,a)|}{\sigma}}} \\ &= \frac{1 - e^{\frac{-|\alpha \cdot \mathrm{TD-Error}|}{\sigma}}}{1 + e^{\frac{-|\alpha \cdot \mathrm{TD-Error}|}{\sigma}}} \end{aligned}$$

$$\varepsilon_{t+1}(s) = \delta \cdot f(s_t, a_t, \sigma) + (1 - \delta) \cdot \varepsilon_t(s) \quad ,$$

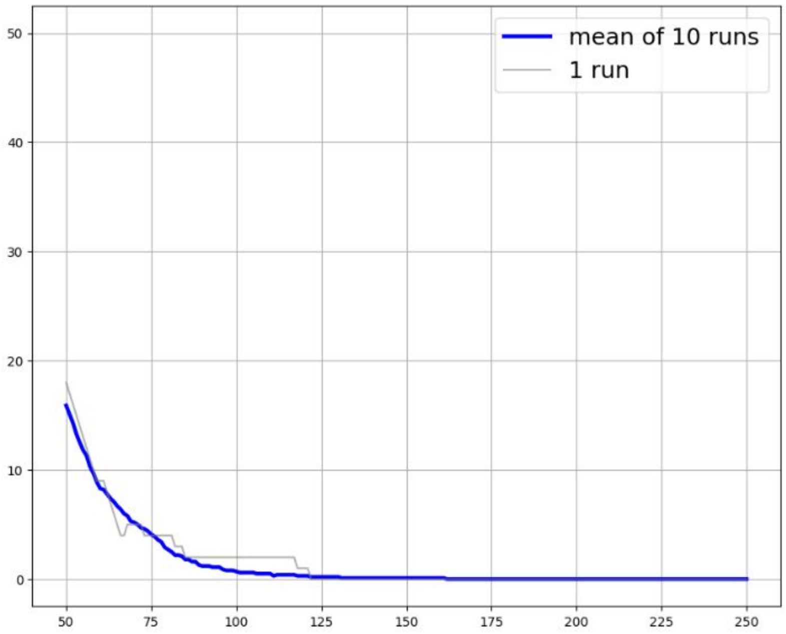
Value-Difference-Based-Epsilon



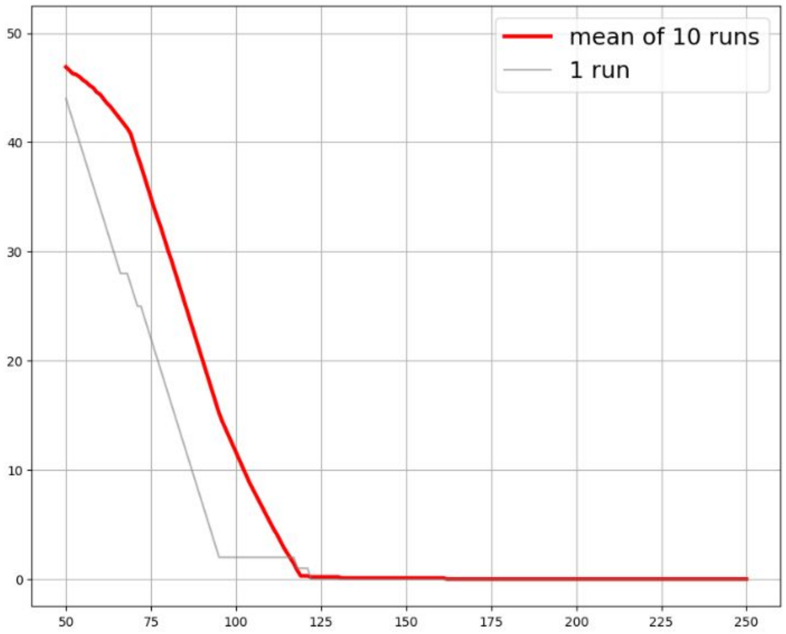


Value-Difference-Based-Epsilon

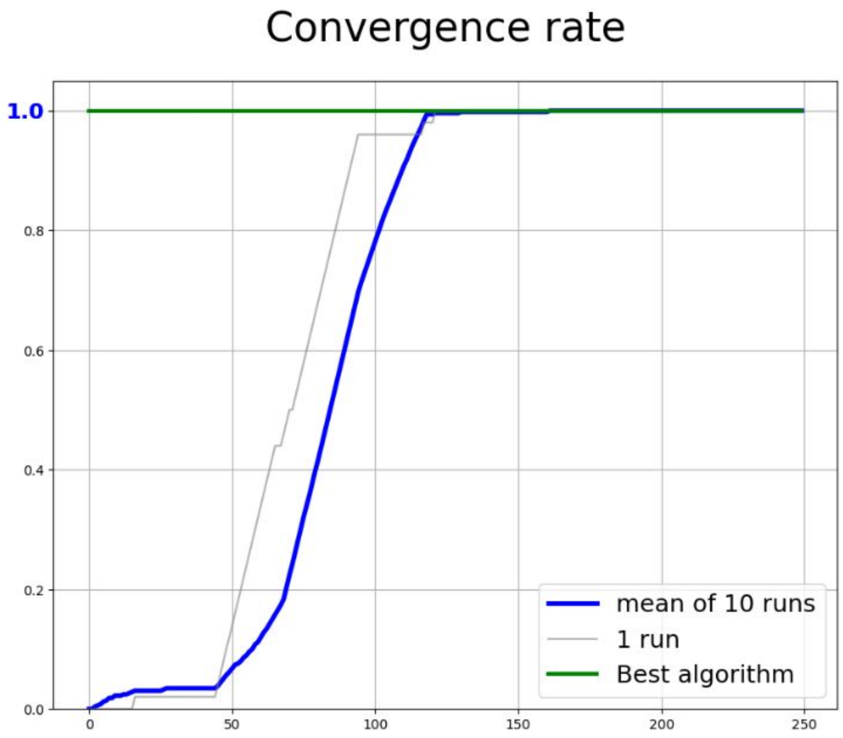
Exploration rate through past 50 iterations



Num of unoptimal arms through past 50 iterations



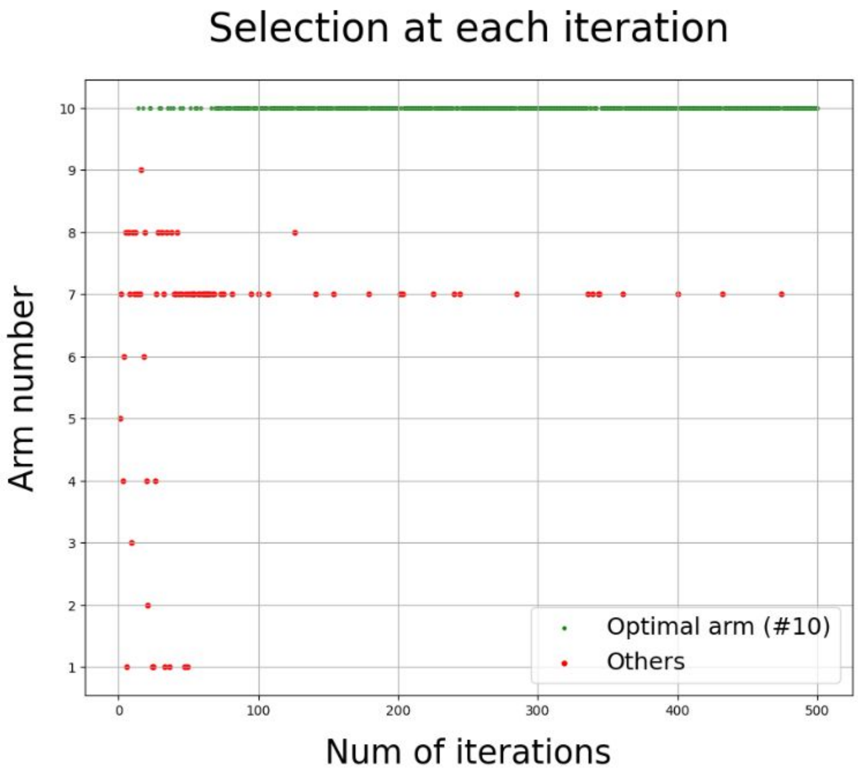
Value-Difference-Based-Epsilon



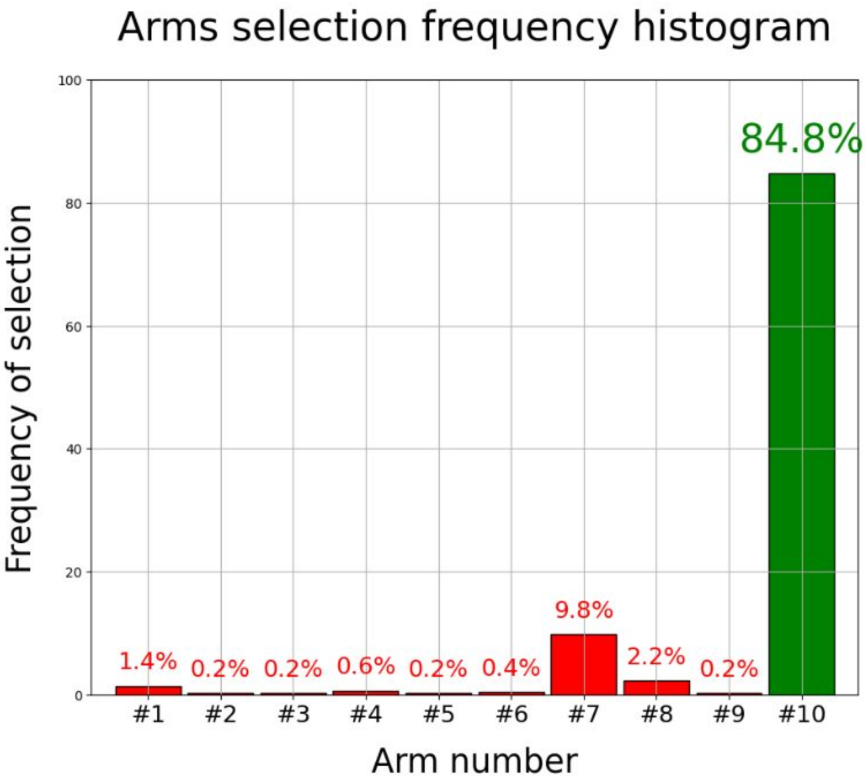
SoftMax

$$\pi(a|s) = Pr\{a_t = a|s_t = s\} = \frac{e^{\frac{Q(s,a)}{\tau}}}{\sum_b e^{\frac{Q(s,b)}{\tau}}}$$

SoftMax



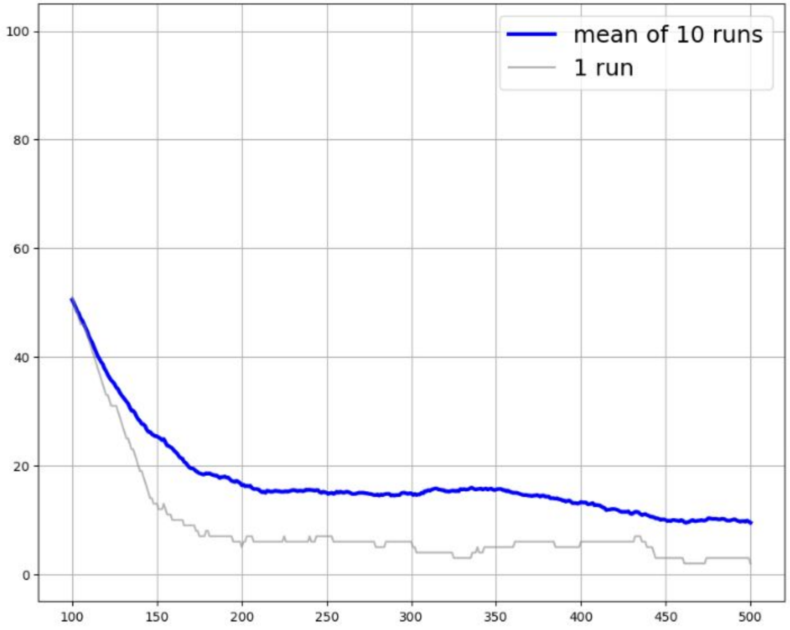
Selection at each iteration
Arm number
Num of iterations
Optimal arm (#10)
Others



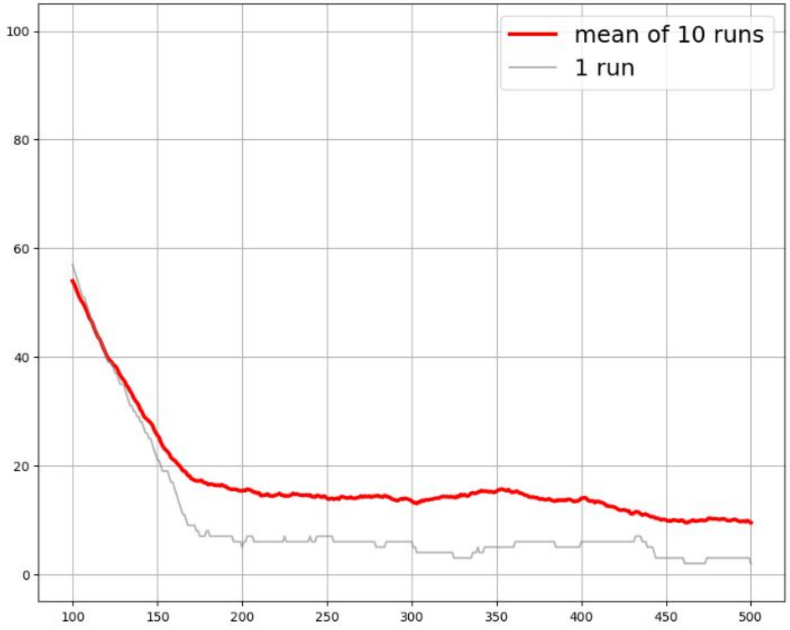
Arms selection frequency histogram
Frequency of selection
Arm number
84.8%
9.8%
1.4%
0.2%
0.2%
0.6%
0.2%
0.4%
2.2%
0.2%
#1
#2
#3
#4
#5
#6
#7
#8
#9
#10

SoftMax

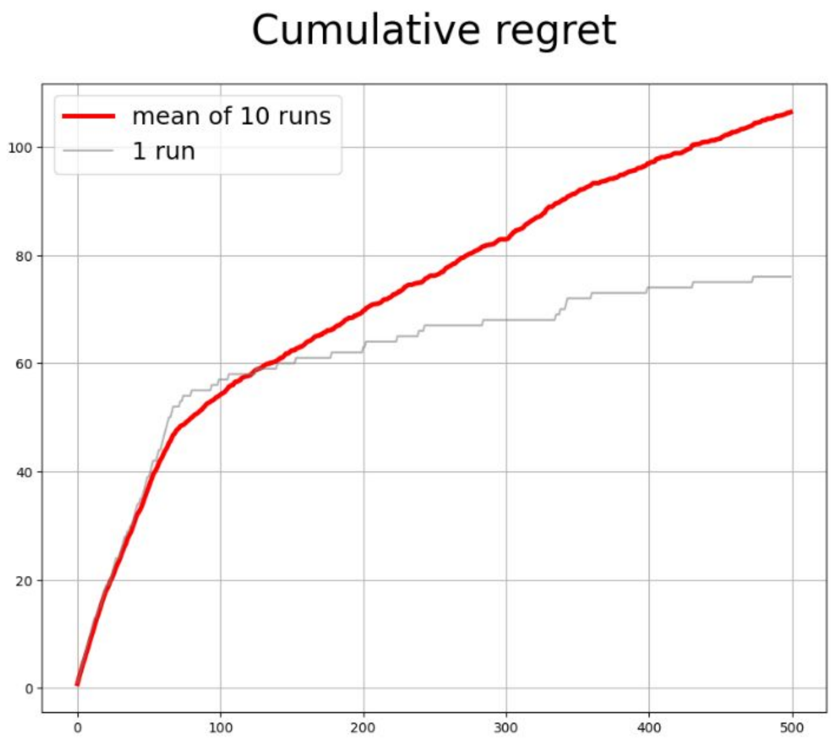
Exploration rate through past 100 iterations

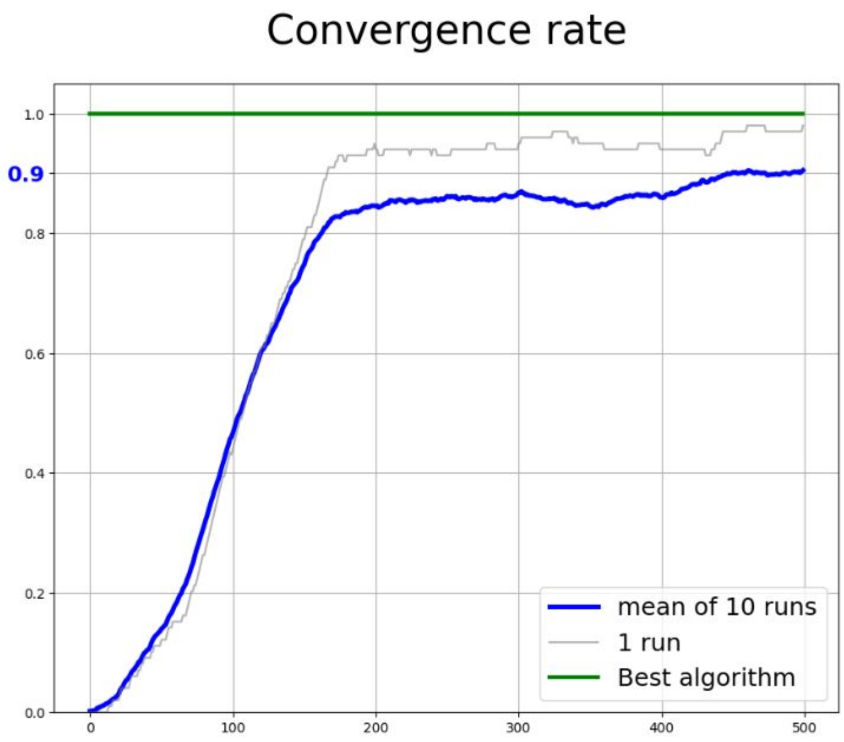


Num of unoptimal arms through past 100 iterations



SoftMax

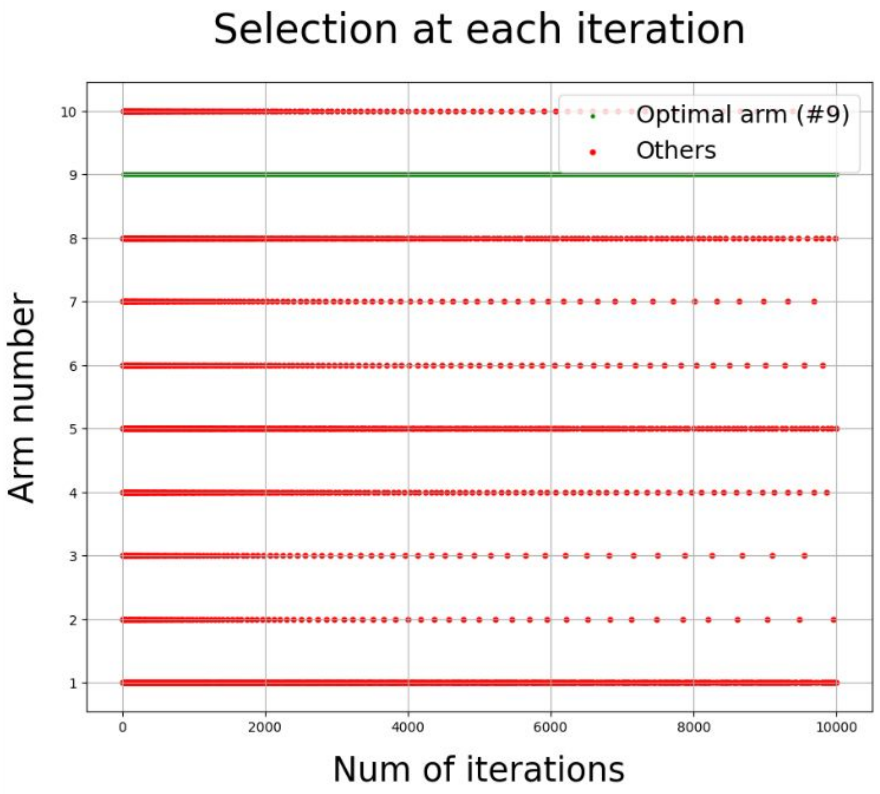

Cumulative regret
mean of 10 runs
1 run
0
20
40
60
80
100
0
100
200
300
400
500

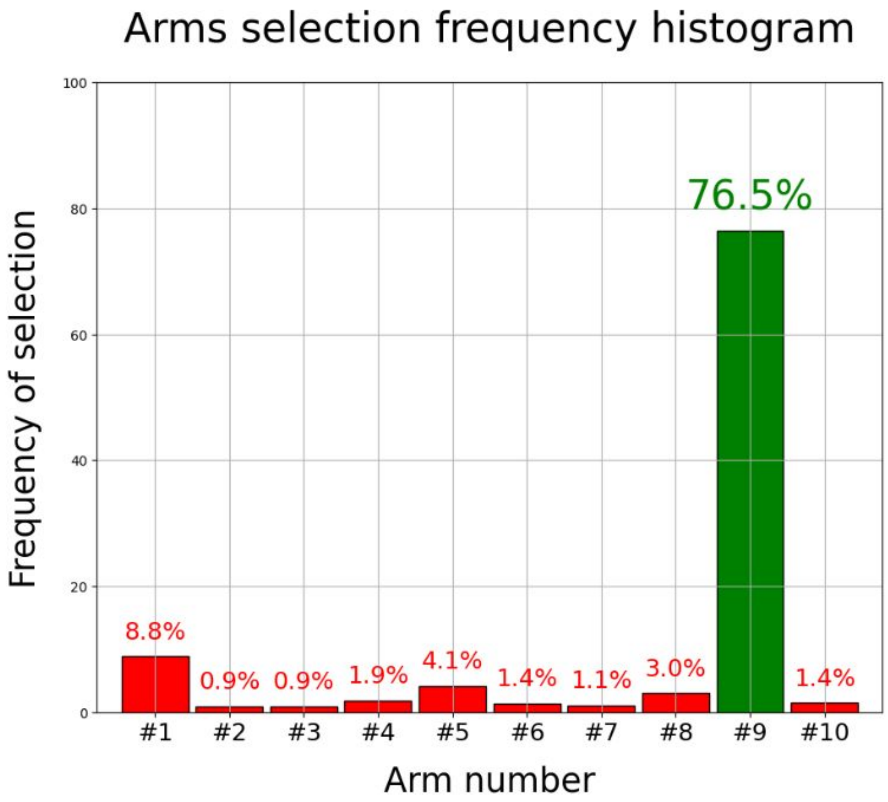

Convergence rate
mean of 10 runs
1 run
Best algorithm
0.0
0.2
0.4
0.6
0.8
0.9
1.0
0
100
200
300
400
500

Upper Confidence Bound

$$A_t \doteq \underset{a}{\arg\max}\left[Q_t(a) + c\sqrt{\frac{\ln t}{N_t(a)}}\right]$$

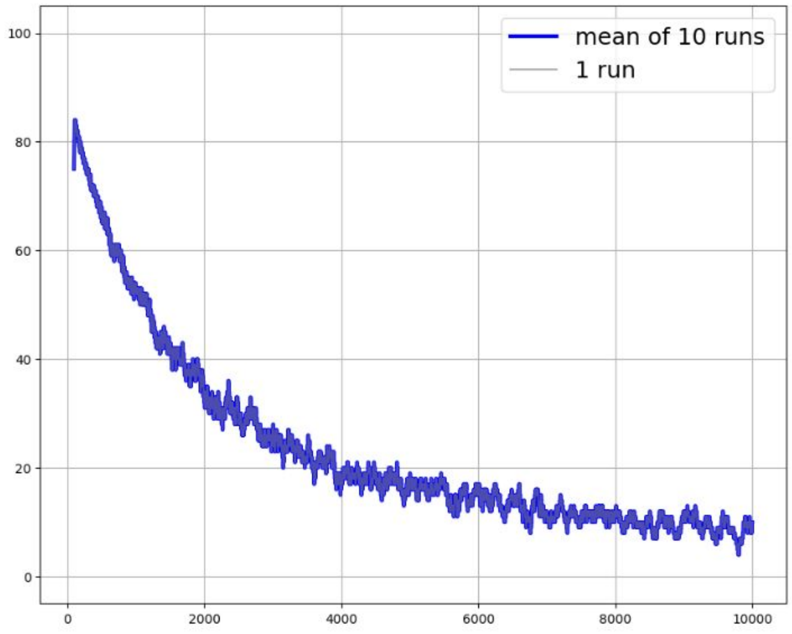
Upper Confidence Bound



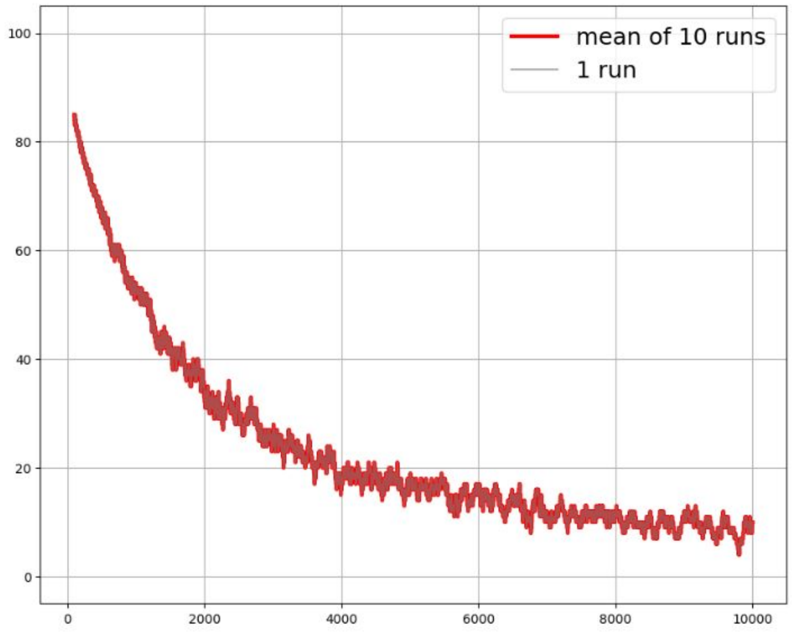


Upper Confidence Bound

Exploration rate through past 100 iterations

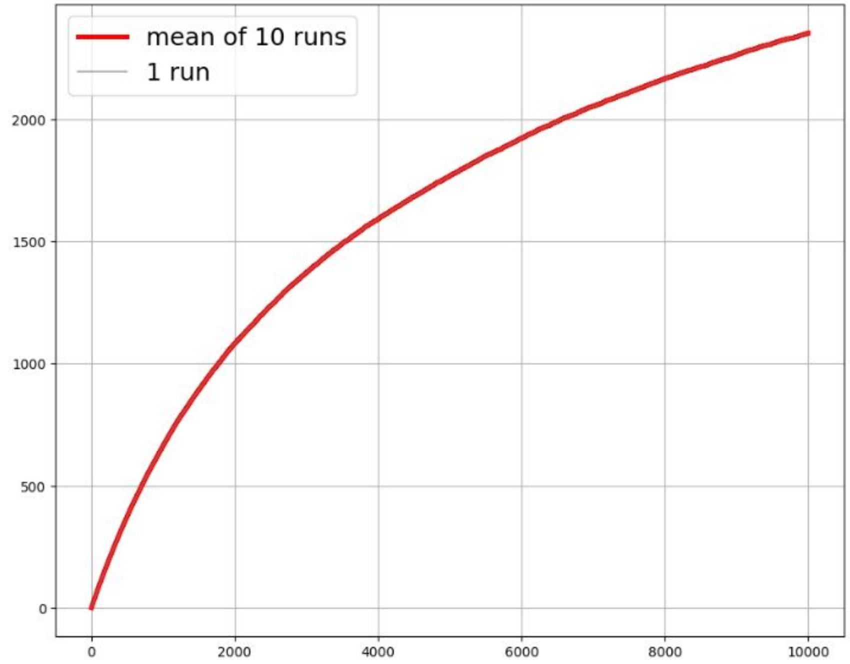


Num of unoptimal arms through past 100 iterations

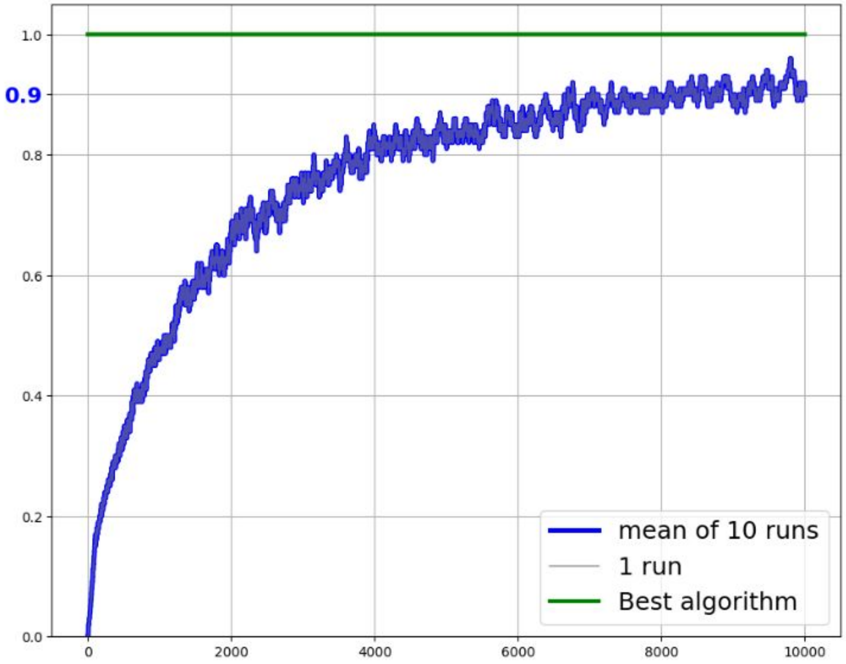


Upper Confidence Bound

Cumulative regret



Convergence rate



Thompson Sampling

Algorithm 1 Thompson Sampling for Bernoulli bandits

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

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

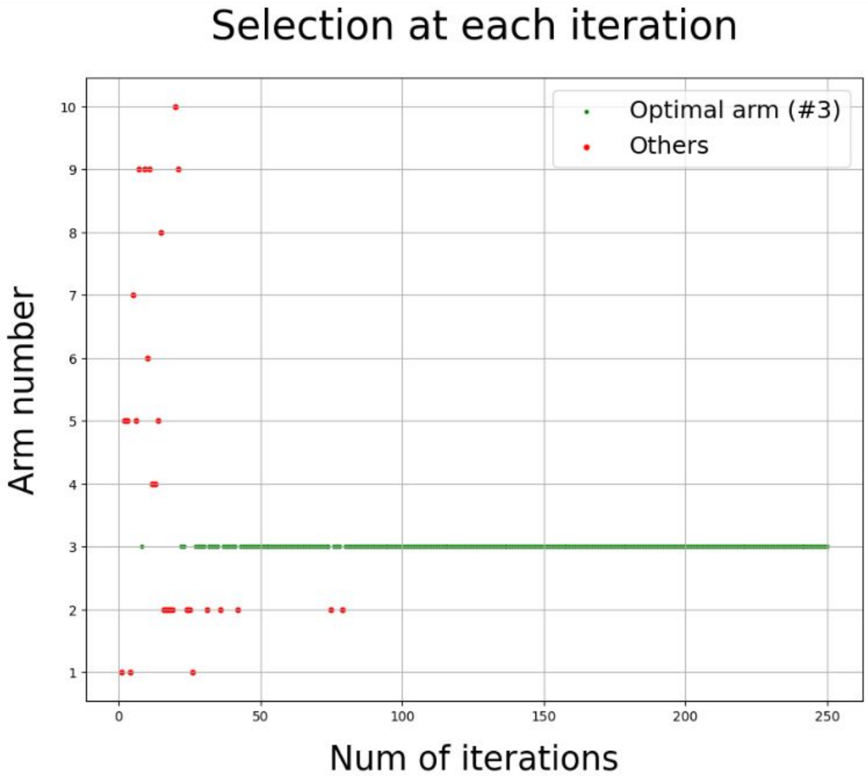
  For each arm $i = 1, \ldots, N$, sample $\theta_i(t)$ from the Beta$(S_i + 1, F_i + 1)$ distribution.

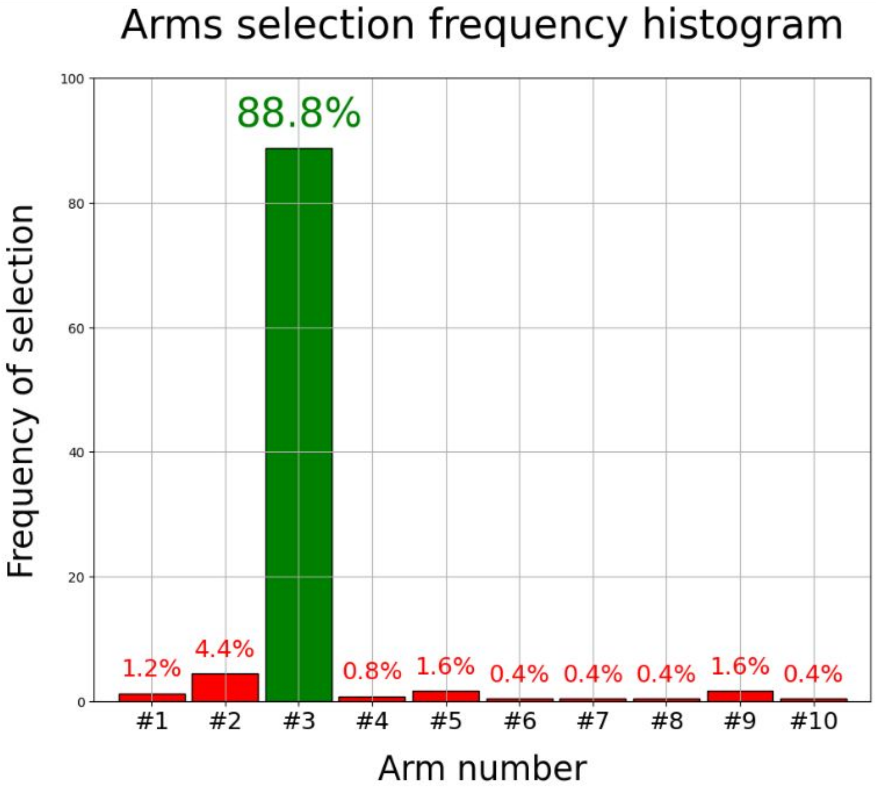
  Play arm $i(t) := \arg\max_i \theta_i(t)$ and observe reward r_t.

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

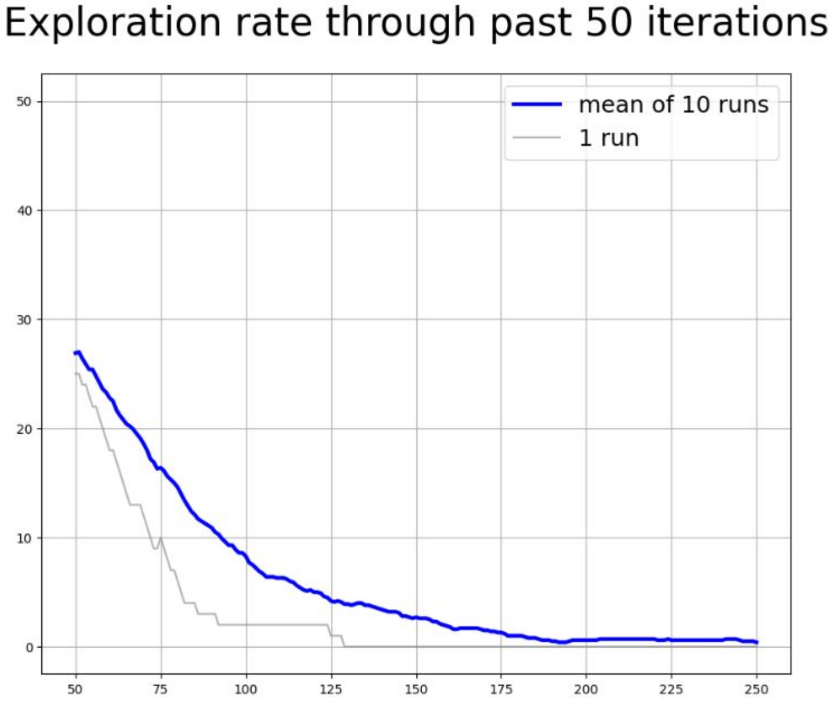
end

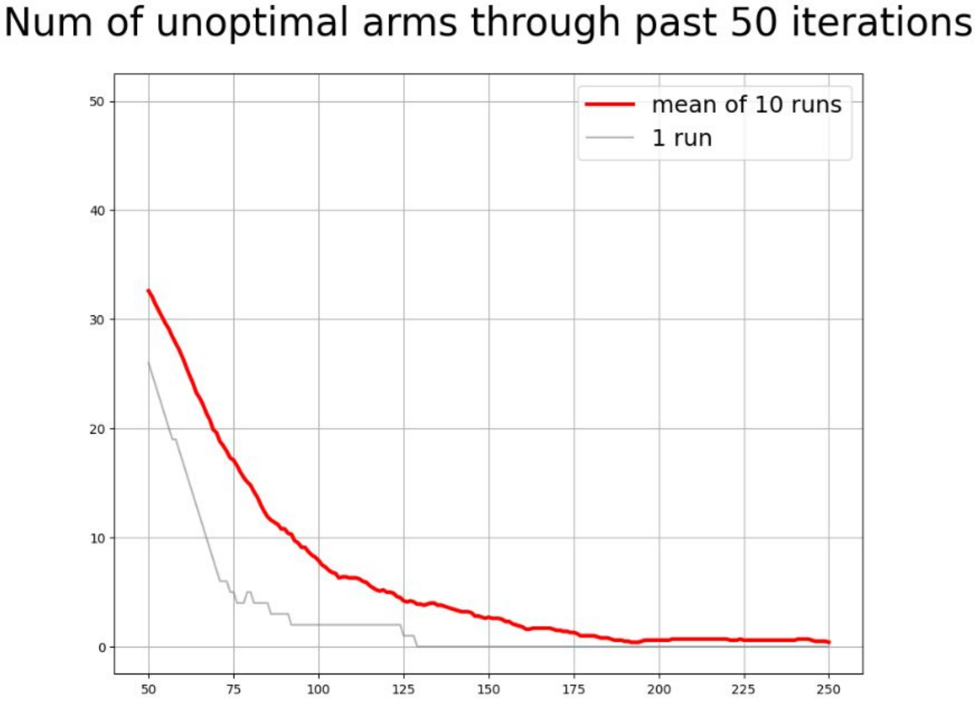
Thompson Sampling





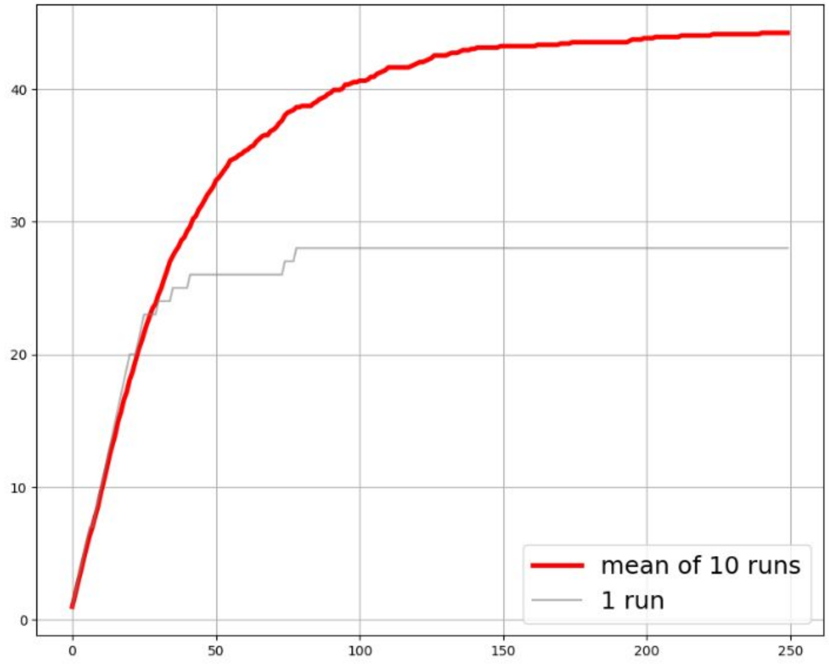
Thompson Sampling



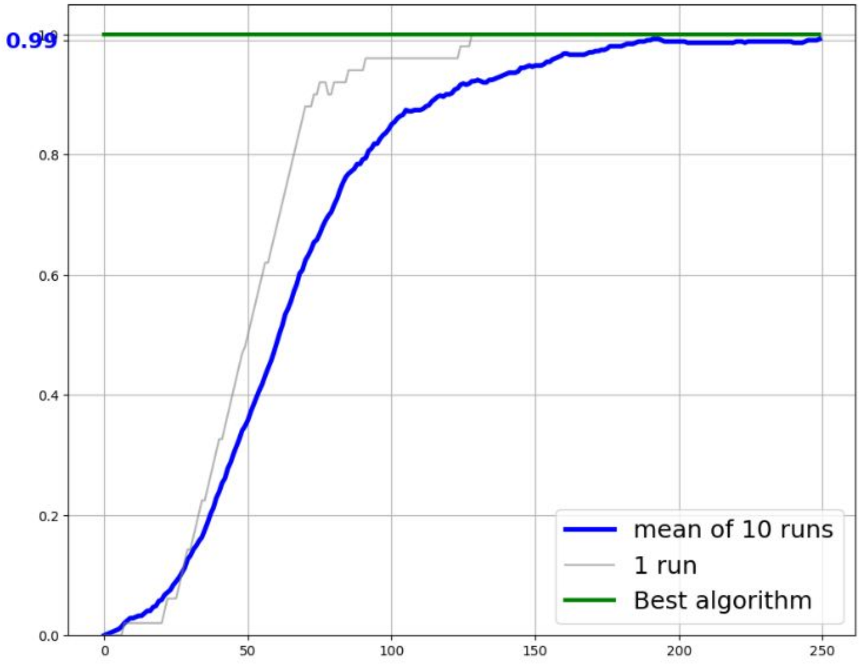


Thompson Sampling

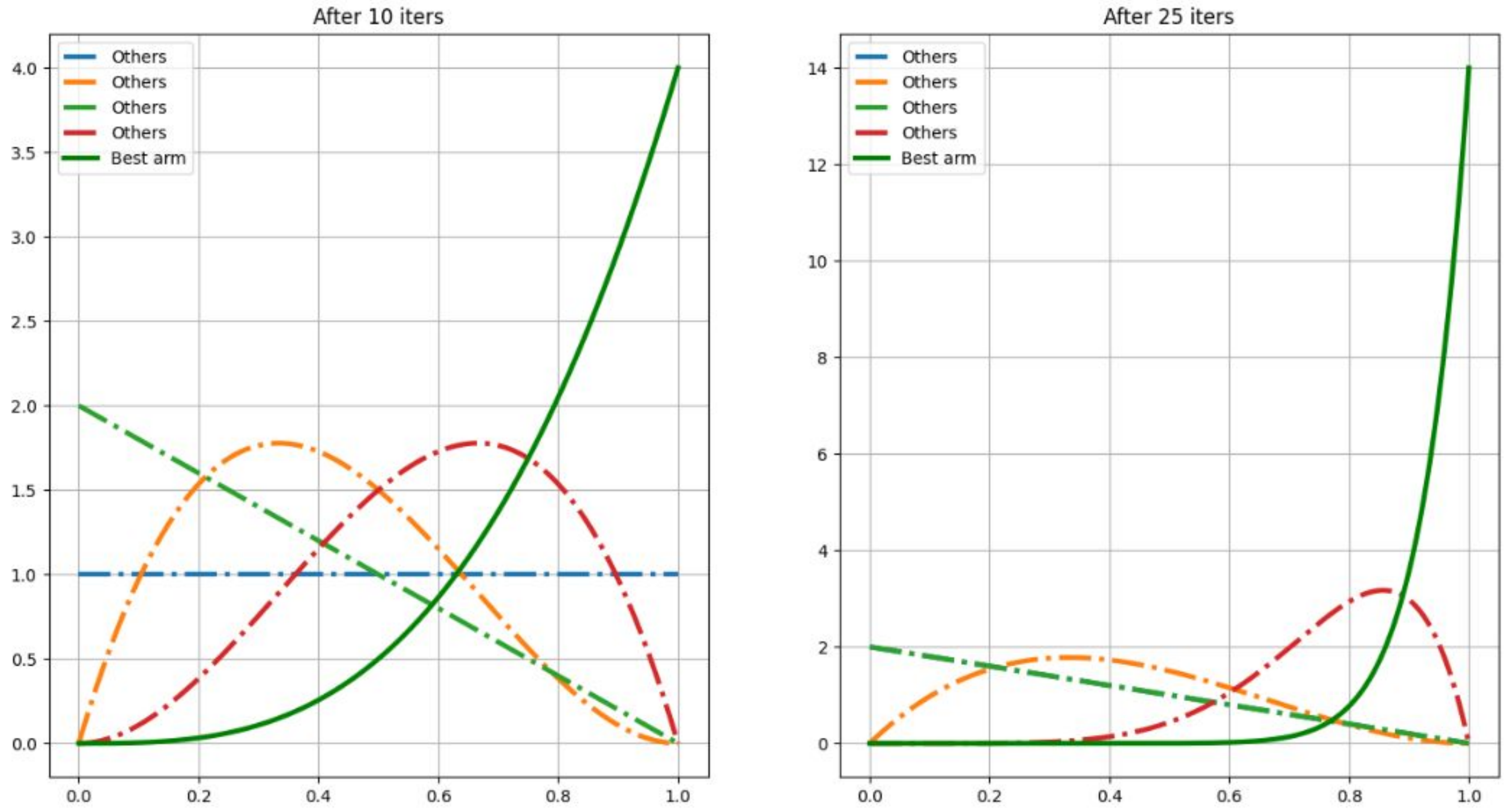
Cumulative regret



Convergence rate



Thompson Sampling



Modifications of TS

Algorithm 1 Online stochastic gradient descent with Thompson Sampling (SGD-TS)

Input: T, K, τ, α.

Randomly choose $a_t \in [K]$ and record X_t, Y_t for $t \in [\tau]$.
Calculate the maximum-likelihood estimator $\hat{\theta}_\tau$ by solving $\sum_{t=1}^{\tau}(Y_t - \mu(X_t^T\theta))X_t = 0$.
Maintain convex set $\mathcal{C} = \{\theta : \|\theta - \hat{\theta}_\tau\| \leq 2\}$.
$\tilde{\theta}_0 \leftarrow \hat{\theta}_\tau$.
for $t = \tau + 1$ **to** T **do**
  if $t\%\tau = 1$ **then**
    $j \leftarrow \lfloor (t-1)/\tau \rfloor$ and $\eta_j = \frac{1}{\alpha j}$.
    Calculate $\nabla l_{j,\tau}$ defined in Equation 3
    Update $\tilde{\theta}_j \leftarrow \prod_{\mathcal{C}}\left(\tilde{\theta}_{j-1} - \eta_j \nabla l_{j,\tau}(\tilde{\theta}_{j-1})\right)$.
    Compute $\bar{\theta}_j = \frac{1}{j}\sum_{q=1}^{j}\tilde{\theta}_q$.
    Compute A_j defined in Equation 5.
    Draw $\theta_j^{\mathrm{TS}} \sim \mathcal{N}\left(\bar{\theta}_j, A_j\right)$.
  end if
  Pull arm $a_t \leftarrow \mathrm{argmax}_{a\in[K]}\, \mu(x_{t,a}^T\theta_j^{\mathrm{TS}})$ and observe reward Y_t.
end for

Online Stochastic Gradient Descent and Thompson Sampling

Algorithm 1 BootstrapLinTS for partially observable delayed feedback

Input: $n_{\mathrm{prior}}, D_{\max}, T, d, K$.

Data $D_0 = ()$
for $n = 1, \ldots, T$ **do**
  Update data D_n with observed conversions
  for $j = 1, \ldots, n_{\mathrm{prior}}$ **do**
    Sample prior ϑ_j and x_j uniformly over $[0,1]^d$
    Normalise sampled ϑ_j and x_j
    Sample prior reward from Bernoulli$(\vartheta_j \cdot x_j)$
    Sample delays uniformly over $[0, D_{\max}]$
  end for
  Concatenate n_{prior} times and rewards with D_n
  Sample with replacement $n + n_{\mathrm{prior}}$ data points
  Estimate $\hat{S}(t, x)$ and $\hat{p}_1(x)$ via EM
  Observe current contexts x_A, $A = 1, \ldots, K$
  for $A = 1, \ldots, K$ **do**
    Calculate probability $(1 - \hat{S}(T, x_A))\hat{p}_1(x_A)$
  end for
  Select arm $\mathrm{argmax}_i(1 - \hat{S}(T, x_A))\hat{p}_1(x_A)$
end for

Bootstrapped Thompson Sampling

Linear Thompson Sampling (LinTS)

Algorithm 1 Thompson Sampling for Contextual bandits

Set $B = I_d, \hat{\mu} = 0_d, f = 0_d$.

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

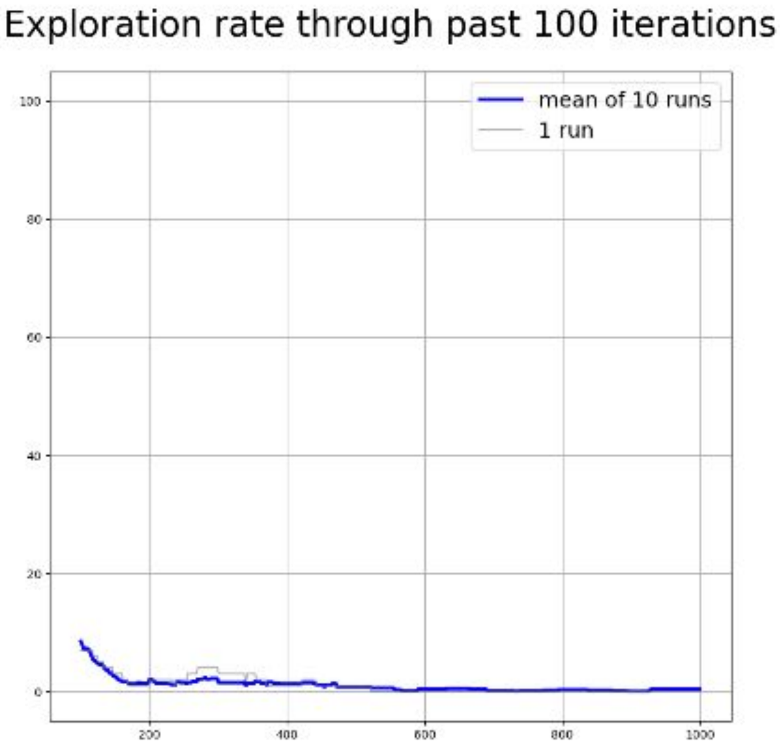
Sample $\tilde{\mu}(t)$ from distribution $\mathcal{N}(\hat{\mu}, v^2 B^{-1})$.

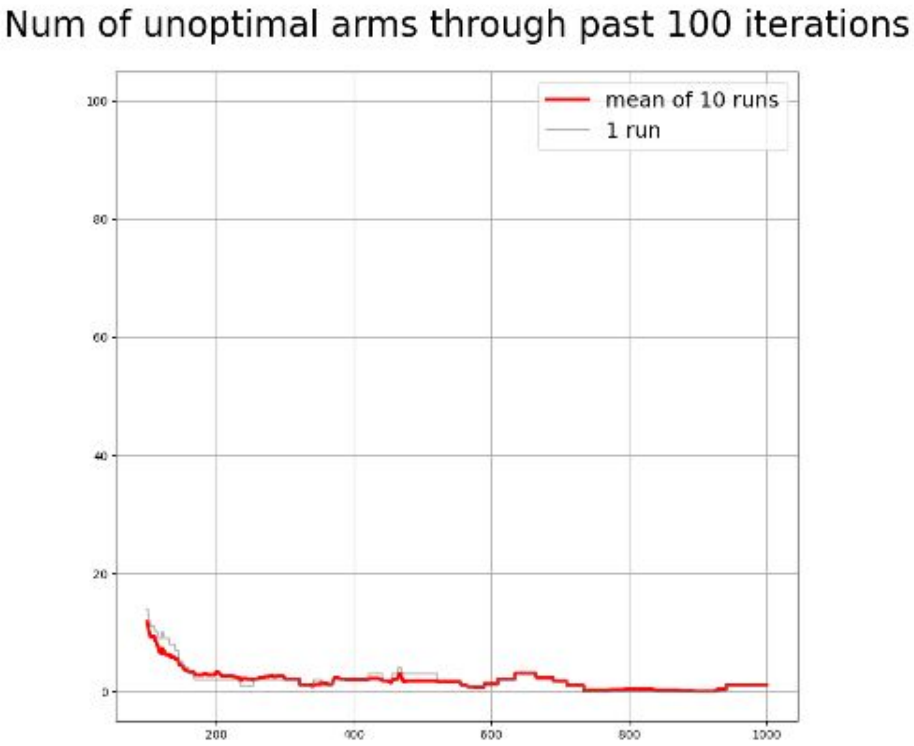
Play arm $a(t) := \arg\max_i b_i(t)^T \tilde{\mu}(t)$, and observe reward r_t.

Update $B = B + b_{a(t)}(t) b_{a(t)}(t)^T$, $f = f + b_{a(t)}(t) r_t$, $\hat{\mu} = B^{-1} f$.

end for

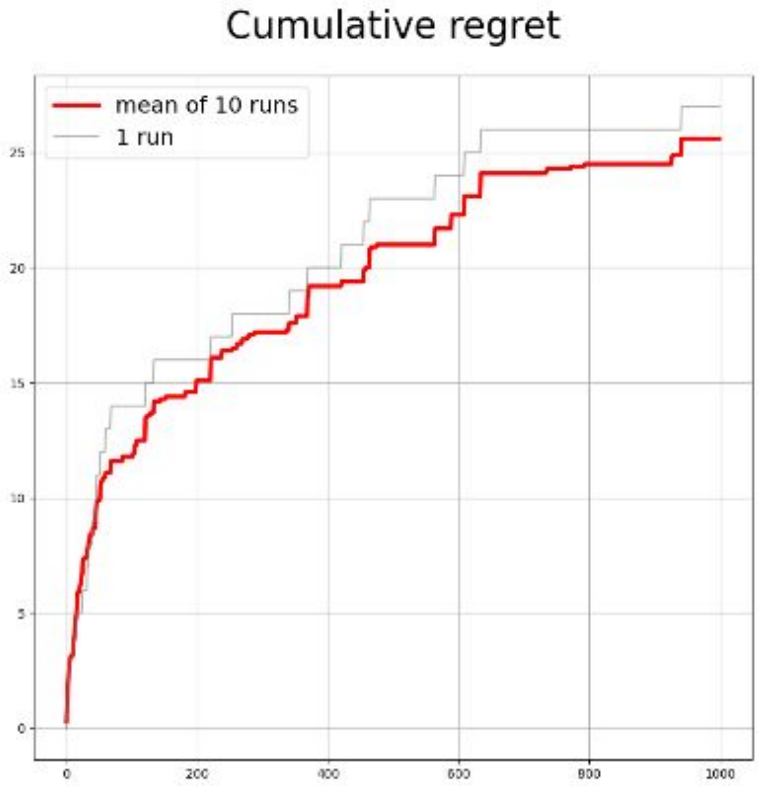
LinTS

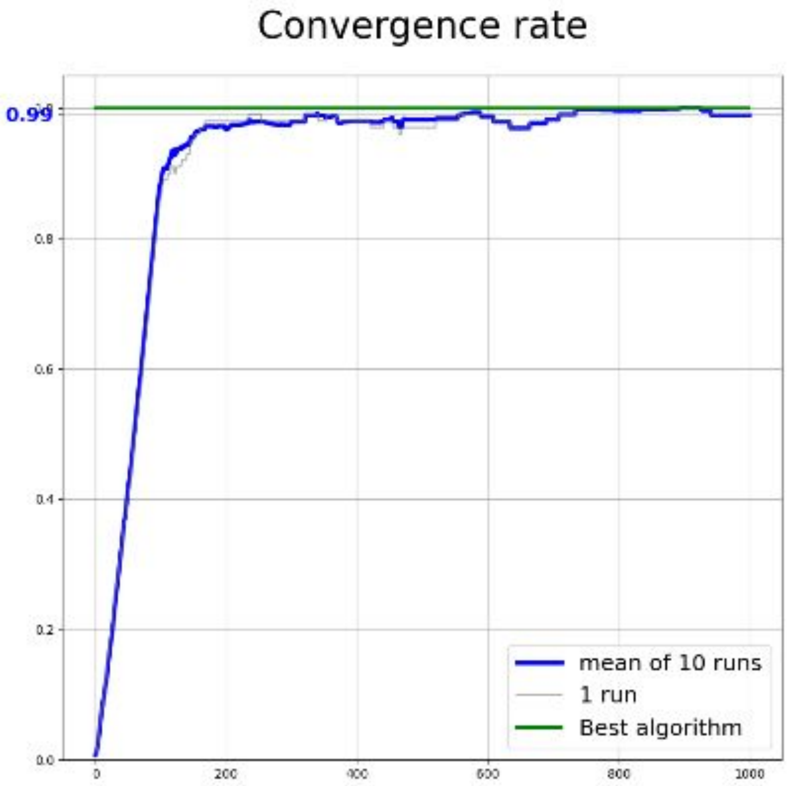




Mushroom dataset

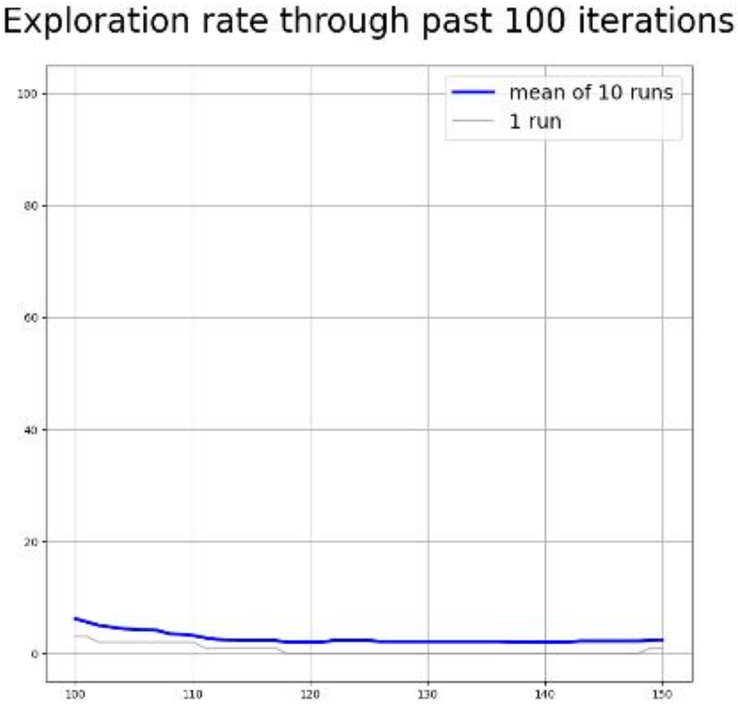
LinTS

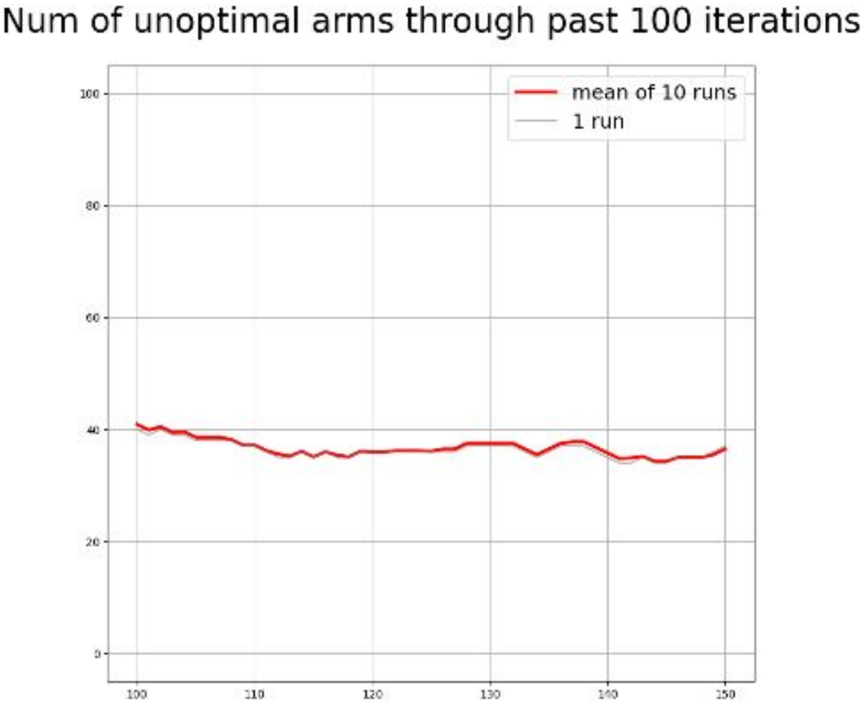




Mushroom dataset

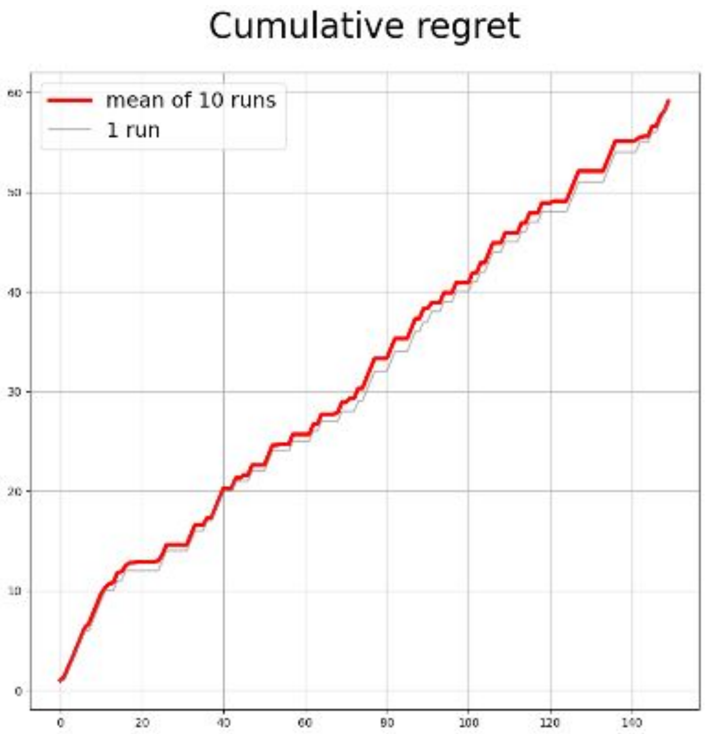
LinTS

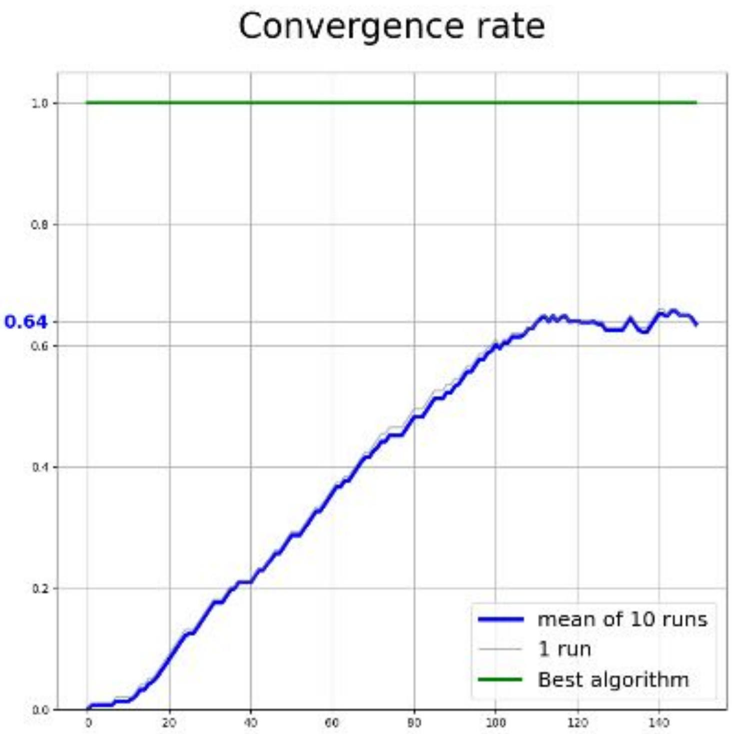




Iris dataset

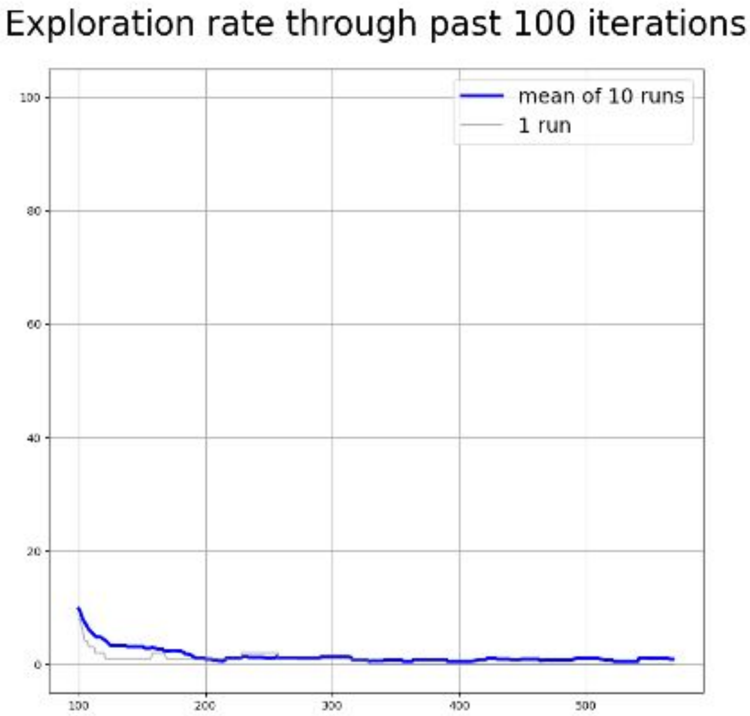
LinTS

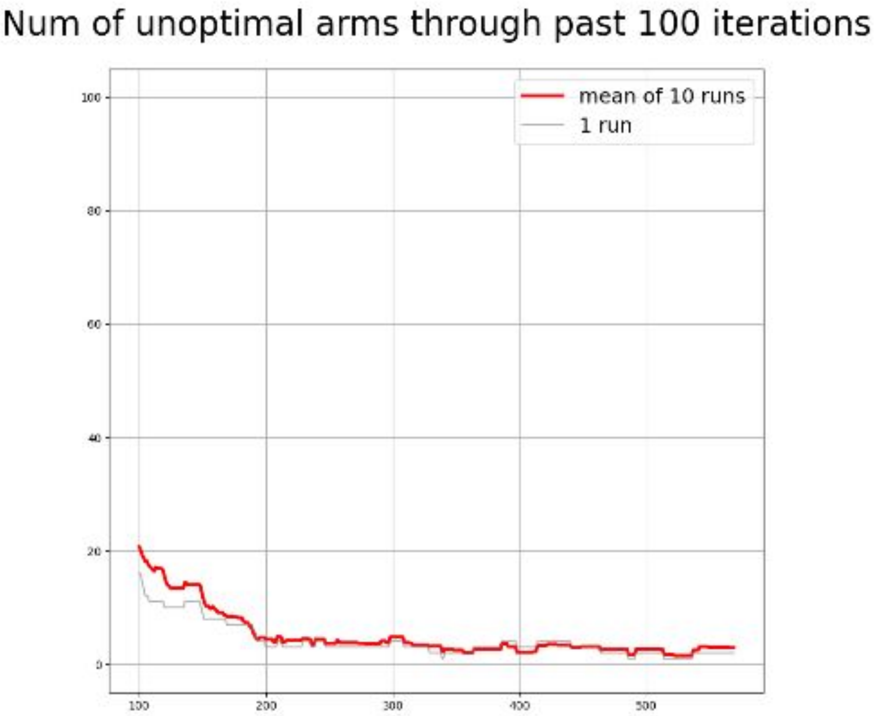




Iris dataset

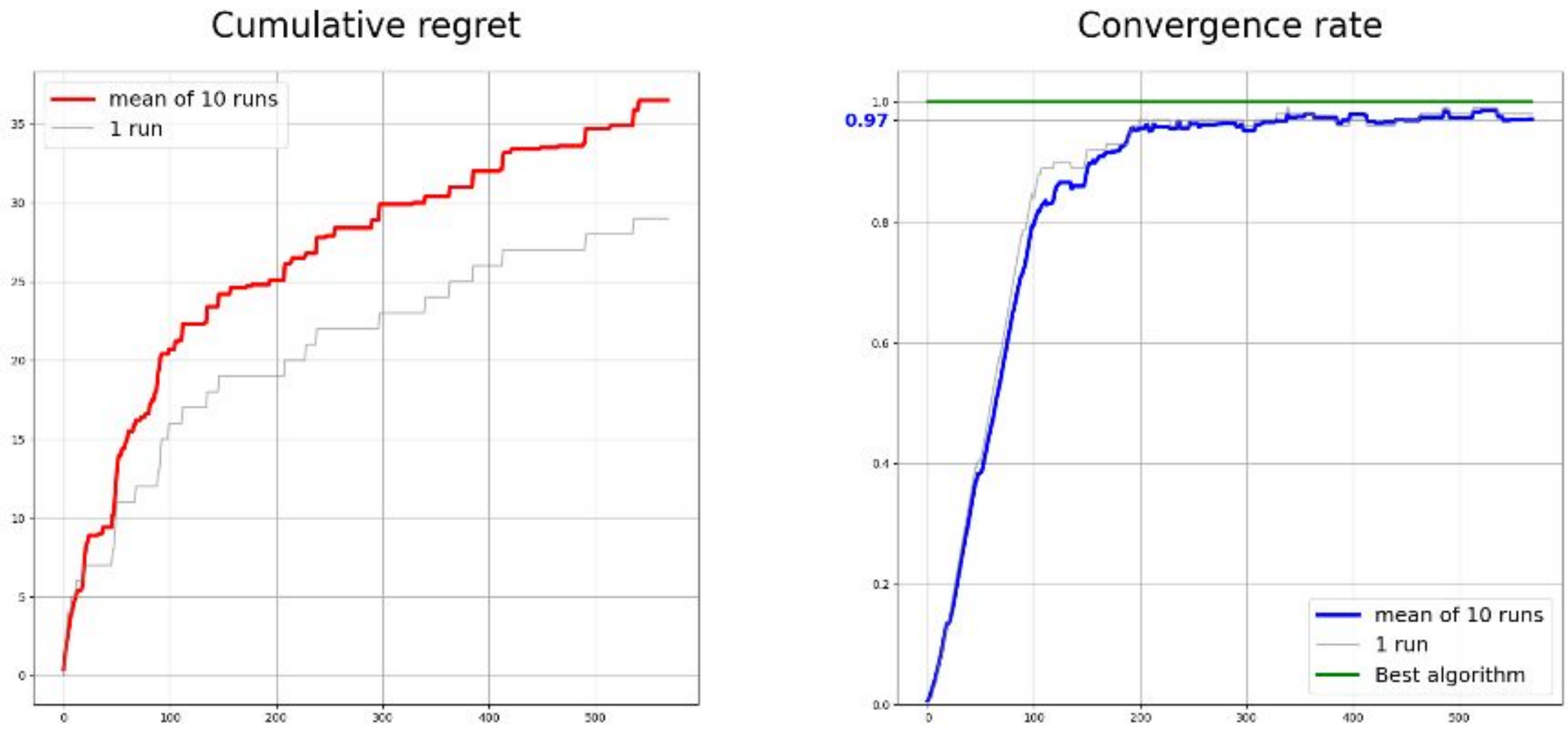
LinTS





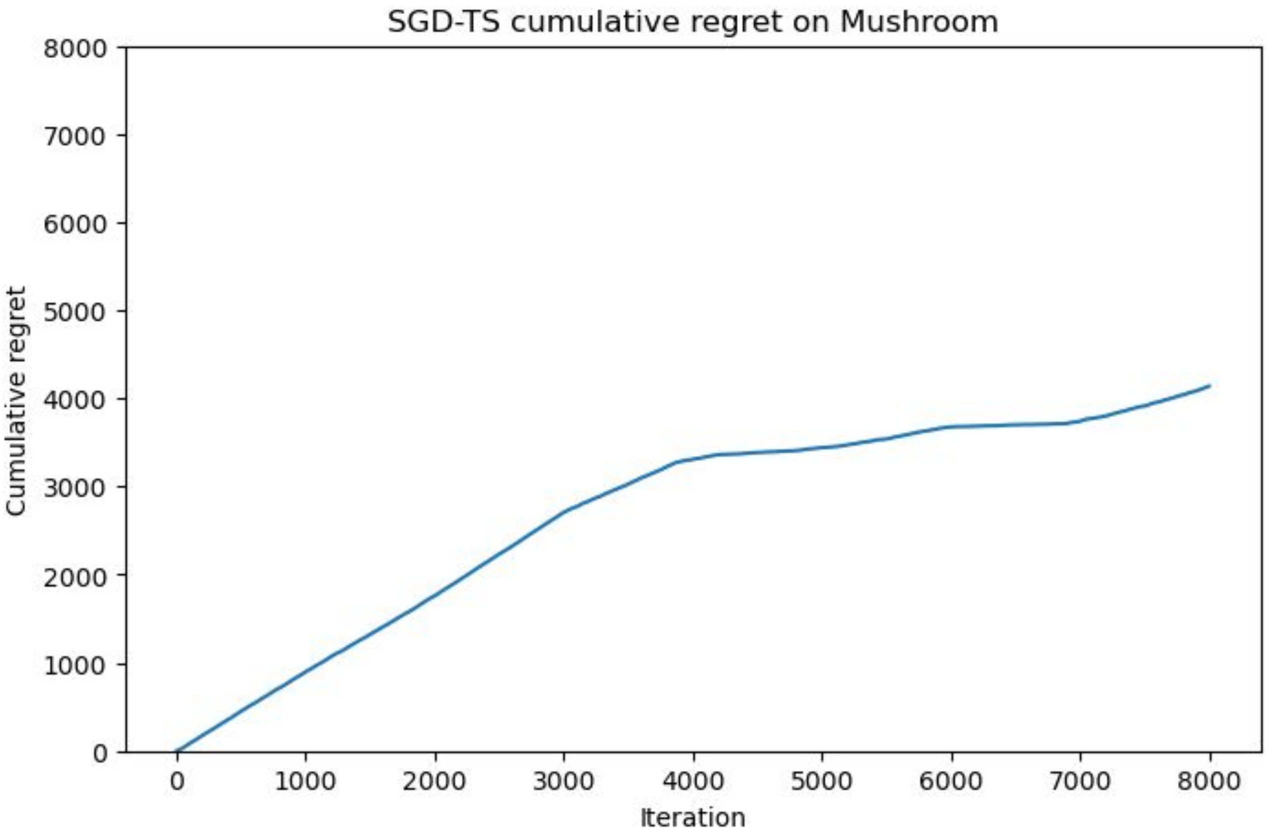
Breast cancer dataset

LinTS



Breast cancer dataset

SGD-TS


SGD-TS cumulative regret on Mushroom
Cumulative regret
Iteration
0
1000
2000
3000
4000
5000
6000
7000
8000

GTS

Algorithm 1 Generalized Thompson Sampling

Input: $\eta > 0, \gamma > 0, \{\mathcal{E}_1, \ldots, \mathcal{E}_N\}$, and prior $\mathbf{p}$

Initialize posterior: $\mathbf{w}_1 \leftarrow \mathbf{p}$; $W_1 \leftarrow \|\mathbf{w}_1\|_1 = 1$

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

Receive context $x_t \in \mathcal{X}$

Select arm a_t according to the mixture probabilities: for each a

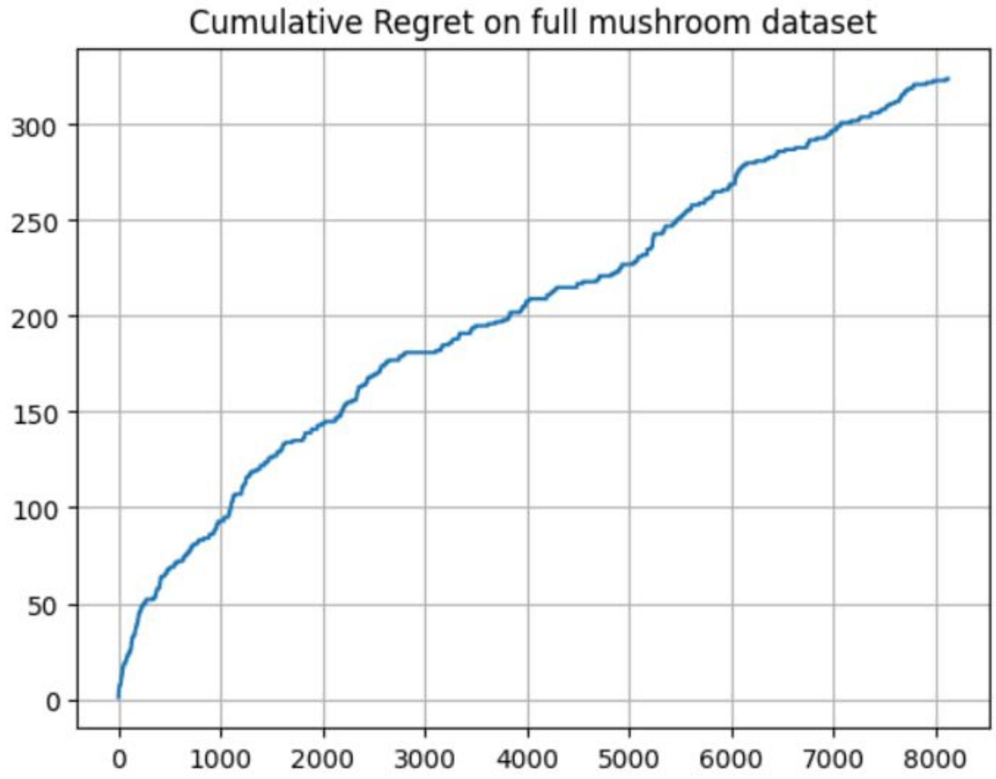
$$\Pr(a) = (1 - \gamma) \sum_{i=1}^{N} \frac{w_{i,t} \mathbb{I}(\mathcal{E}_i(x_t) = a)}{W_t} + \frac{\gamma}{K}$$

Observe reward r_t, and updates weights:

$$\forall i : w_{i,t+1} \leftarrow w_{i,t} \cdot \exp\left(-\eta \cdot \ell\left(f_i(x_t, a_t), r_t\right)\right); \qquad W_{t+1} \leftarrow \|\mathbf{w}_{t+1}\|_1 = \sum_i w_{i,t+1}$$

end for

GTS


Cumulative Regret on full mushroom dataset
300
250
200
150
100
50
0
0
1000
2000
3000
4000
5000
6000
7000
8000