Thompson sampling and Ensemble sampling

Yingru Li

https://richardli.xyz/

Extracted from my remote talk at
Informs Optimiztion Society Conference, March, 2024

Outline

Sequential Decision-making under Uncertainty

Existing solutions and their limitations

Sequential decision-making

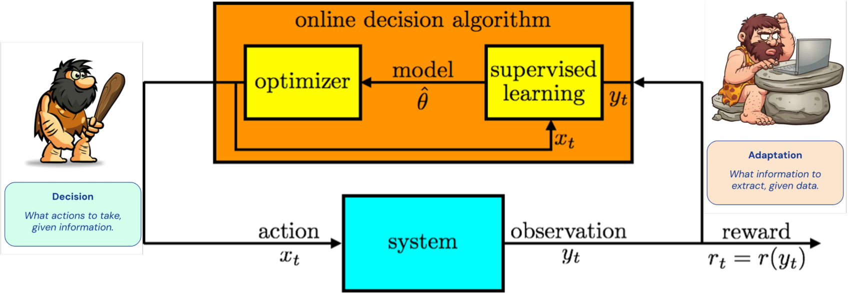


Figure: An **Agent (online decision algorithm)** interacts with **the environment (system)**.

- **Adaptation**: At time t, **the agent** extracts information from history data $D_{t-1} = (x_1, y_1, \ldots, x_{t-1}, y_{t-1})$. E.g., estimate model $\hat{\theta}$ for unknown system.
- **Decision**: Then, **the agent** selects action x_t accordingly and observes the outcome y_t.

Sequential decision-making

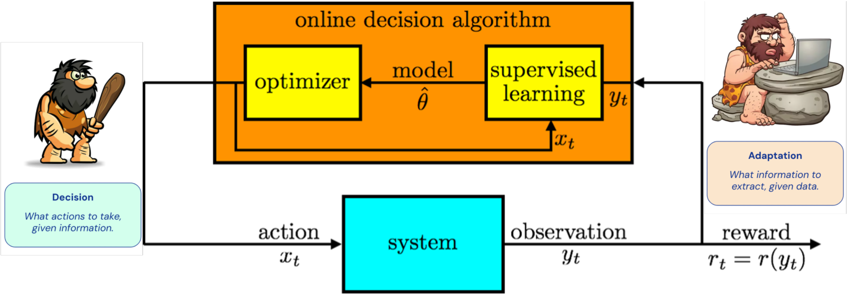


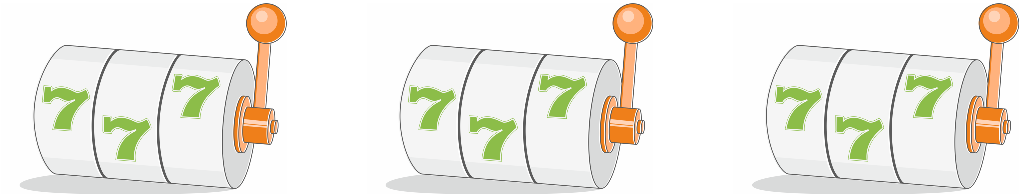
Figure: An **Agent (online decision algorithm)** interacts with **the environment (system)**.

- **Goal**: Select actions $(x_t)_{t \geqslant 1}$ to maximize total expected future reward $\mathbb{E}\left[\sum_t r(y_t)\right]$.

Exploration-Exploitation tradeoff.

May require balancing long term & immediate rewards.

A simple setup: Bernoulli bandits



(a) Action 1: $\theta_1^* = 0.6$ **(b)** Action 2: $\theta_2^* = 0.4$ **(c)** Action 3: $\theta_3^* = 0.7$

- 3 actions with mean rewards $\theta^* = \{\theta_1^* = 0.6, \theta_2^* = 0.4, \theta_3^* = 0.7\}$, unknown to the Agent but fixed.
- Each time t, an action $x_t = k$ is selected and the observation

$$y_t \sim \text{Bernoulli}(\theta_k^*)$$

is revealed, resulting the reward $r_t = y_t$.

Source of uncertainty: unknown environments and insufficient data

- $\theta^* = \{\theta_1^* = 0.6, \theta_2^* = 0.4, \theta_3^* = 0.7\}$ unknown.
- The agent begin with an independent uniform prior belief over each θ_k^*.
- The agent's beliefs in any given time period about these mean rewards can be expressed in terms of posterior distributions.
  - **Posterior** $\propto$ **Prior** $\times$ **Data likelihoods**
  - More Data $\Rightarrow$ Posterior concentrates!
  - Less Data $\Rightarrow$ Posterior spreads!

Epistemic Uncertainty due to **insufficient** data.

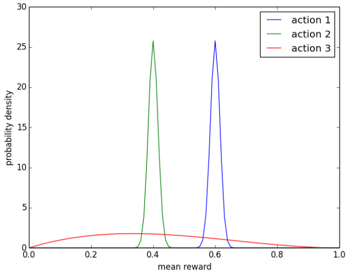


Figure: Posterior p.d.f. over mean rewards after the agent tries **actions 1 and 2 one thousand times** each, **action 3 three times**, receives cumulative rewards of 600, 400, and 1.

Why agent needs to track the degree of uncertainty - Greed is no good

- **Greedy** algorithm (maximize expected mean reward with current belief) will always select action 1.
- Under current belief: Reasonable to avoid action 2, since it is extremely unlikely $\theta_2^* > \theta_1^*$.
- Because of high uncertainty in θ_3^*, there is some chance $\theta_3^* > \theta_1^*$. In the long run, the agent should try action 3.

Greedy algorithm fails to account for uncertainty information in θ_3^*, causing suboptimal decision.

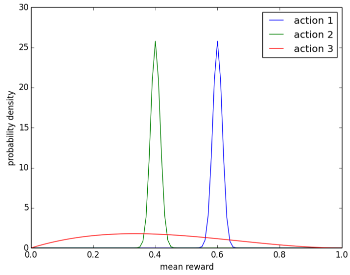


Figure: Posterior p.d.f. over mean rewards after the agent tries **actions 1 and 2 one thousand times each, action 3 three times**, receives cumulative rewards of 600, 400, and 1. Ground truth $\{\theta_1^* = 0.6, \theta_2^* = 0.4, \theta_3^* = 0.7\}$.

Why agent needs to track the degree of uncertainty - Thompson sampling

Algorithm: Thompson sampling (TS)

- Given prior distribution $p_0(\theta^*)$ over model θ^*. Set initial dataset $D_0 = \emptyset$.
- For $t = 1, \ldots, T$,
  - **Sample** $\tilde{\theta}_t \sim p(\theta^* \mid D_{t-1})$ from posterior
  - **Select** $x_t = \arg\max_{x \in \mathcal{A}} \mathbb{E}[r(y_t) \mid x_t = x, \theta^* = \tilde{\theta}_t]$ and **observe** y_t and $r_t = r(y_t)$
  - **Update** the history dataset $D_t = D_{t-1} \cup \{(x_t, y_t)\}$

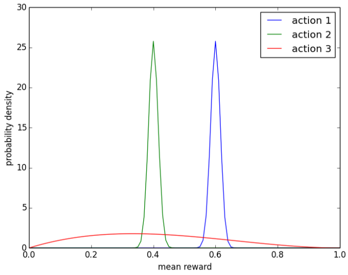


Figure: Posterior p.d.f. over mean rewards after the agent tries **actions 1 and 2 one thousand times each, action 3 three times**, receives cumulative rewards of 600, 400, and 1. Ground truth $\{\theta_1^* = 0.6, \theta_2^* = 0.4, \theta_3^* = 0.7\}$.

- TS would sample actions 1, 2, or 3, with prob. $\approx$ 0.82, 0, and 0.18, respectively.
- TS explores θ_3^* to solve its uncertainty and finally identifies the optimal action

Why agent needs to track the degree of uncertainty

Definition 1 (Performance metric: Regret).

$$\textit{Regret}(T) = \sum_{t=1}^{T} \mathbb{E}[\max_x \mathbb{E}[r(y) \mid x, \theta^*] - r(y_t)]$$

In previous bernoulli bandit example, $\theta_1^* = 0.6, \theta_2^* = 0.4, \theta_3^* = 0.7$ and

$$\max_x \mathbb{E}[r(y) \mid x, \theta^*] = \theta_3^*.$$

Therefore, Regret$(T) = T\theta_3^* - \mathbb{E}[\sum_{t=1}^{T} r(y_t)]$.

Why agent needs to track the degree of uncertainty - Thompson sampling

Algorithm: Thompson sampling (TS)

- Given prior distribution $p_0(\theta^*)$ over model θ^*. Set initial dataset $D_0 = \emptyset$.
- For $t = 1, \ldots, T$,
  - **Sample** $\tilde{\theta}_t \sim p(\theta^* \mid D_{t-1})$ from posterior
  - **Select** $x_t = \arg\max_{x \in \mathcal{A}} \mathbb{E}[r(y_t) \mid x_t = x, \theta^* = \tilde{\theta}_t]$ and **observe** y_t and $r_t = r(y_t)$
  - **Update** the history dataset $D_t = D_{t-1} \cup \{(x_t, y_t)\}$

Theorem 1 (Thompson sampling for K-armed bandit [RVRK+18]).

K actions with mean parameter $\{\theta_1^*, \ldots, \theta_K^*\}$, and when played, any action yields the observation $y_t \sim \mathrm{Bernoulli}(\theta_k^*)$ and resulting the reward $r_t = r(y_t)$. The regret ***lower bound*** *is $\Omega(\sqrt{KT})$. Thompson sampling achieves near-optimal regret up to a $\log K$ factor,*

$$\textit{Regret}(T) = O(\sqrt{KT \log K}).$$

How to track the degree of uncertainty? Bayesian inference

- Given data $D_T = \{(x_t, y_t), t = 1, \dots, T\}$, compute the posterior of θ^* via the **Bayes rule**

$$p(\theta^* \mid D_T) \propto p(D_T \mid \theta^*) p_0(\theta^*)$$

Example: Beta-Bernoulli model

- Prior: $\theta^* \in \mathbb{R}^K$ each $\theta_k \sim p_0 : \mathrm{Beta}(\alpha_k, \beta_k)$
- $y_t \sim \mathrm{Bernoulli}(\theta_{x_t})$
- Posterior over $\theta_k \mid D_T$ still Beta with parameters

$$\left(\alpha_k + \sum_{t=1}^{T} y_t \mathbb{I}_{x_t = k}, \beta_k + \sum_{t=1}^{T} (1 - y_t) \mathbb{I}_{x_t = k}\right)$$

Example: Linear-Gaussian model

- Prior: $\theta^* \in \mathbb{R}^d \sim p_0 : N(\mu_0, \boldsymbol{\Sigma}_0)$
- $y_t = \langle \theta^*, x_t \rangle + \omega_t^*$ and $\omega_t^* \sim N(0, \sigma^2)$
- Gaussian Posterior $\theta^* \mid D_T \sim N(\mu_T, \boldsymbol{\Sigma}_T)$

$$\boldsymbol{\Sigma}_T = \left(\frac{1}{\sigma^2} \sum_{t=1}^{T} x_t x_t^\top + \Sigma_0^{-1}\right)^{-1},$$

$$\mu_T = \boldsymbol{\Sigma}_T \left(\frac{1}{\sigma^2} \sum_{t=1}^{T} x_t y_t + \Sigma_0^{-1} \mu_0\right).$$

Conjugacy allows for incremental update on Bayesian posterior

- Update $\boldsymbol{\Sigma}_t$ by Sherman-Morrison formula

$$\boldsymbol{\Sigma}_t = \left(\boldsymbol{\Sigma}_{t-1}^{-1} + \frac{1}{\sigma^2} x_t x_t^\top\right)^{-1} = \boldsymbol{\Sigma}_{t-1} - \frac{\boldsymbol{\Sigma}_{t-1} x_t x_t^\top \boldsymbol{\Sigma}_{t-1}}{\sigma^2 + x_t^\top \boldsymbol{\Sigma}_{t-1} x_t}$$

- (Incrementally) Update $p_t := \Sigma_t^{-1} \mu_t$ with

$$\underbrace{\boldsymbol{\Sigma}_t^{-1} \mu_t}_{p_t} = \underbrace{\boldsymbol{\Sigma}_{t-1}^{-1} \mu_{t-1}}_{p_{t-1}} + \frac{1}{\sigma^2} x_t y_t \tag{1}$$

- Compute μ_t

$$\mu_t = \boldsymbol{\Sigma}_t p_t$$

Research question

Fact:

Without conjugacy properties, exact Bayesian posterior inference is intractable.

Question:

How to perform posterior sampling **without** using conjugacy?

Outline

Sampling through pptimization with perturbed history

- For a history dataset $D_t = \{(x_s, y_s)_{s=1}^t\}$, perturb with algorithmic noise to generate a

Perturbed history $\tilde{D}_t = \{ \tilde{\theta}_0 \sim N(\mu_0, \Sigma_0), (x_s, y_s + \sigma z_s); z_s \sim N(0,1), s = 1, \ldots, t\}$

- Randomize Least Square (**RLS**) via Perturbed History (**PH**) [OAC18, OVRRW19]

$$\theta_t = \arg\min_{\theta} \ell(\theta; \tilde{D}_t) := \frac{1}{\sigma^2} \sum_{s=1}^{t} (g_\theta(x_s) - y_s - \sigma z_s)^2 + \theta^\top \Sigma_0^{-1} \theta \quad (2)$$

where $g_\theta(x)$ could be a generic nonlinear function.

Significance of Equation (2)

- **Sampling** through a **purely computational perspective**.
- **No** explicit posterior inference.
- **No** use of conjugacy properties.

Understanding RLS-PH under fixed history

Justification of Equation (2): **posterior sampling**

If the **fixed** history dataset D_t is generated from a **Linear-Gaussian model** w. prior $\theta^* \sim N(\mu_0, \Sigma_0)$ and $g_\theta(x) = \langle \theta + \tilde{\theta}_0, x \rangle$, then the optimal solution of Equation (2) is a posterior sample

$$\tilde{\theta}_t := (\theta_t + \tilde{\theta}_0) \overset{i.i.d.}{\sim} \theta^* \mid D_t.$$

$$\tilde{\theta}_t := \theta_t + \tilde{\theta}_0 = \boldsymbol{\Sigma}_t \left(\frac{1}{\sigma^2} \sum_{s=1}^{t} x_s (y_s + \sigma z_s) + \boldsymbol{\Sigma}_0^{-1} \tilde{\theta}_0 \right) \quad s.t.$$

$$\mathbb{E}[\tilde{\theta}_t \mid D_t] = \boldsymbol{\Sigma}_t \left(\frac{1}{\sigma^2} \sum_{s=1}^{t} x_s (y_s + \sigma \underbrace{\mathbb{E}[z_s \mid D_t]}_{=0}) + \boldsymbol{\Sigma}_0^{-1} \underbrace{\mathbb{E}[\tilde{\theta}_0 \mid D_t]}_{=0} \right) = \mu_t = \mathbb{E}[\theta^* \mid D_t],$$

$$\mathrm{Cov}[\tilde{\theta}_t \mid D_t] = \boldsymbol{\Sigma}_t \left(\frac{1}{\sigma^2} \sum_{s=1}^{t} x_s \underbrace{\mathbb{E}[z_s z_s^\top \mid D_t]}_{=\boldsymbol{I}} x_s^\top + \boldsymbol{\Sigma}_0^{-1} \underbrace{\mathrm{Cov}[\tilde{\theta}_0 \mid D_t]}_{=\boldsymbol{\Sigma}_0} \boldsymbol{\Sigma}_0^{-1} \right) \boldsymbol{\Sigma}_t = \boldsymbol{\Sigma}_t = \mathrm{Cov}[\theta^* \mid D_t].$$

A hypothetical algorithm for sequential-decision making without conjugacy

Incremental RLS for linear bandit w. prior: $\theta^* \in \mathbb{R}^d \sim p_0 : N(\mu_0, \boldsymbol{\Sigma}_0)$.

- Initialize prior perturbation $\tilde{\theta}_0 \sim N(\mu_0, \Sigma_0)$
- For $t = 1, \ldots, T$ do
  - — **Decision**: Select $\mathbf{x}_t = \arg\max_{x \in \mathcal{A}} \langle x, \tilde{\theta}_{t-1} \rangle$ and observe $y_t = \langle \theta^*, x_t \rangle + \omega_t^*$

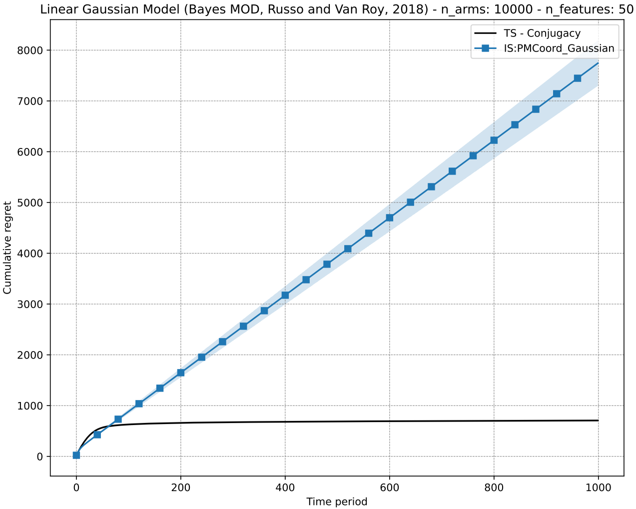
    where $\omega_t^* \sim N(0, \sigma^2)$ is the environmental noise
  - — **Adaptation**: Incrementally update model according to recursive LS

$$\tilde{\theta}_t = \boldsymbol{\Sigma}_t \left(\boldsymbol{\Sigma}_{t-1}^{-1} \tilde{\theta}_{t-1} + \frac{y_t + \sigma z_t}{\sigma^2} \mathbf{x}_t \right) \tag{3}$$

where each $z_t \sim N(0, 1)$ is an independent perturbation at each step t.

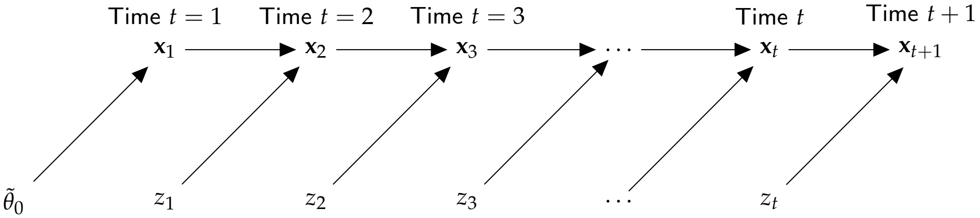
- Starting from this page, we use boldface $\mathbf{x}_t$ to emphasize it is a **history-dependent R.V.**.
- $D_t = \{(\mathbf{x}_s, y_s)_{s=1}^t\}$ is a **adaptively sampled dataset**.

Does incremental RLS work for sequential-decision making?



- Bayesian regret (avg 200 expes) in Linear-Gaussian bandit
- ✗ **Incremental RLS** (Blue) suffer **linear regret** (failure).
- ✓ **Thompson sampling** (Black) uses conjugacy for posterior update and then generates a sample from posterior. **Sublinear regret**.

Why incremental RLS does not work for sequential decision making?



Sequential Dependence due to **incremental update** alongside **sequential decision-making**.

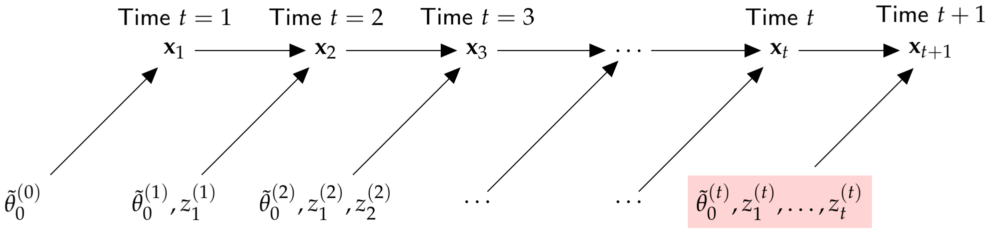
- Posterior mean **not** matching due to the **Sequential Dependence**. $D_t = \{(\mathbf{x}_s, y_s)_{s=1}^t\}$

$$\mathbb{E}[\tilde{\theta}_t \mid D_t] = \boldsymbol{\Sigma}_t \left(\boldsymbol{\Sigma}_0^{-1} \underbrace{\mathbb{E}[\tilde{\theta}_0 \mid D_t]}_{\neq \mu_0} + \sum_{s=1}^{t-1} \frac{\mathbf{x}_s}{\sigma^2} (y_s + \sigma \underbrace{\mathbb{E}[z_s \mid D_t]}_{\neq 0}) + \frac{\mathbf{x}_t}{\sigma^2} (y_t + \sigma \underbrace{\mathbb{E}[z_t \mid D_t]}_{=0}) \right) \neq \mathbb{E}[\theta^* \mid D_t]$$

- ✗ Incremental RLS produces biased posterior sample!

Deal with issues due to Sequential Dependence? Solution 1: Resampling

- For each step t, **resample**, $\tilde{\theta}_0^{(t)} \sim N(\mu_0, \boldsymbol{\Sigma}_0), z_s^{(t)} \sim N(0,1)$ for $s = 1, \ldots, t$ independently and
- Form a new perturbed history $\tilde{D}_t^{(t)} = \{\tilde{\theta}_0^{(t)}, (\mathbf{x}_s, y_s + \sigma z_s^{(t)}); s = 1, \ldots, t\}$ for each step t,



- For each step t, **re-train** perturbed optimization problem from scratch, resulting $\tilde{\theta}_t(\tilde{D}_t^{(t)})$.
- ✓ **Posterior sampling**: $\tilde{\theta}_t(\tilde{D}_t^{(t)}) \sim \theta^* \mid D_t$ since $D_t \perp\!\!\!\perp (\tilde{\theta}_0^{(t)}, z_1^{(t)}, z_2^{(t)}, \ldots, z_t^{(t)})$. **Break the dependence!**
- ✗ Computational cost **growing unboundedly** as data accumulated. **No Incremental update**.

Deal with issues due to Sequential Dependence? Solution 2: Ensemble

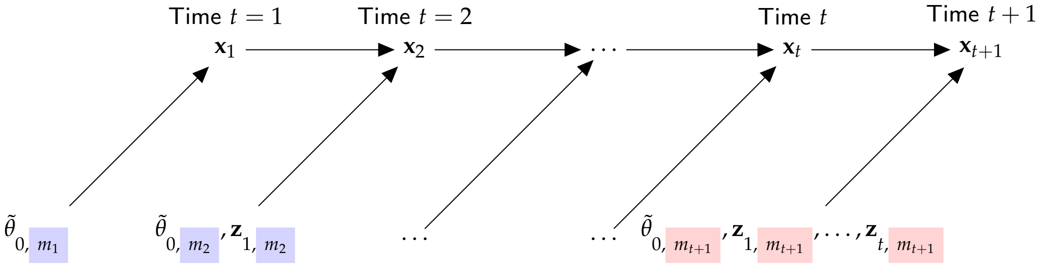
Ensemble sampling (ES) [OVRRW19, LVR17]

- Initialize each m-th model $\tilde{\theta}_{0,m} \sim N(\mu_0, \Sigma_0)$ independently for $m \in \{1, \ldots, M\}$
- For $t = 1, \ldots, T$ do
  - — **Decision**: Sample $m_t \sim \text{unif}\{1, \ldots, M\}$. Select $\mathbf{x}_t = \arg\max_{x \in \mathcal{A}} \langle x, \tilde{\theta}_{t-1, m_t} \rangle$ and observe y_t
  - — **Adaptation**: $\forall m \in [M]$, **Incrementally** update each m-th model according to

$$\tilde{\theta}_{t,m} = \boldsymbol{\Sigma}_t \left(\boldsymbol{\Sigma}_{t-1}^{-1} \tilde{\theta}_{t-1,m} + \frac{y_t + \sigma \mathbf{z}_{t,m}}{\sigma^2} \mathbf{x}_t \right) \tag{4}$$

where each $\mathbf{z}_t = (\mathbf{z}_{t,1}, \cdots, \mathbf{z}_{t,M})^\top \sim N(0, \boldsymbol{I}_M)$ is an independent perturbation at each step t.

Why ensemble sampling works? Intuition



- **Intuition**: breaking the dependence by large ensemble size
- If M sufficiently large, at time $t+1$, ES select an index $m_{t+1} \neq m, \forall m \in \{m_s\}_{s=1}^t$ w.h.p., then

$$\mathbb{E}[\tilde{\theta}_{t,m_{t+1}} \mid D_t] = \boldsymbol{\Sigma}_t \left(\boldsymbol{\Sigma}_0^{-1} \underbrace{\mathbb{E}[\tilde{\theta}_{0,m_{t+1}} \mid D_t]}_{=\mu_0} + \sum_{s=1}^{t} \frac{\mathbf{x}_s}{\sigma^2} (y_s + \sigma \underbrace{\mathbb{E}[\mathbf{z}_{s,m_{t+1}} \mid D_t]}_{=0}) \right) = \mathbb{E}[\theta^* \mid D_t]$$

and posterior covariance also matches as $D_t \perp\!\!\!\perp (\tilde{\theta}_{0,m_{t+1}}, (\mathbf{z}_{s,m_{t+1}})_{s=1}^t)$

Online incremental optimization formulation of ensembles

- For each $m \in [M]$, the m-th perturbed history dataset: Init $\tilde{D}_{0,m} = \{\tilde{\theta}_{0,m}\}$ and increment

$$\tilde{D}_{t,m} = \tilde{D}_{t-1,m} \cup \{\mathbf{x}_t, y_t, \mathbf{z}_{t,m}\}$$

- The m-th model

$$\tilde{\theta}_{t,m} = \underbrace{\theta_{t,m}}_{\text{learned model}} + \underbrace{\tilde{\theta}_{0,m}}_{\text{prior perturbation}}$$

- For each $m \in [M]$, the learned model $\theta_{t,m}$ is the solution of the **incremental RLS** updated from $\theta_{t-1,m}$ with new data $(\mathbf{x}_t, y_t)$:

$$\theta_{t,m} = \arg\min_{\theta} L(\theta; \tilde{D}_{t,m}) = \frac{1}{\sigma^2}(g_{\theta,m}(\mathbf{x}_t) - y_t - \sigma \mathbf{z}_{t,m})^2 + (\theta - \theta_{t-1,m})^\top \Sigma_{t-1}^{-1}(\theta - \theta_{t-1,m}) \quad (5)$$

- If $g_{\theta,m}(x) = \langle \theta + \tilde{\theta}_{0,m}, x \rangle$, Equation (5) reduces to Equation (4).
- **In general**, $g(\cdot)$ could be any function, including nonlinear mapping, e.g. neural networks.

Limitations of Ensmeble Sampling

- **Histogram effect**: Larger M, uniform distribution over M models $\mathcal{U}(\tilde{\theta}_1, \ldots, \tilde{\theta}_M)$ better approximate the true posterior distribution.
- **Sequential dependence issue**: inevitably introduced by the interleaving between **incremental update** and **sequential decision-making**. To solve this issue, we need large ensemble size to break the dependence.

Statistics v.s. Computation Trade-offs

- **Posterior approximation**: Requires a huge number of ensembles ($M > 100$) for **good** approximation and sequential decision-making. [LLZ+22, OWA+23, LXHL24]
- ✗ **Computationally expensive**: say, update > 100 neural networks for each time step.

References I

[LLZ+22] Ziniu Li, Yingru Li, Yushun Zhang, Tong Zhang, and Zhi-Quan Luo. HyperDQN: A randomized exploration method for deep reinforcement learning. In *International Conference on Learning Representations*, 2022.

[LVR17] Xiuyuan Lu and Benjamin Van Roy. Ensemble sampling. *Advances in neural information processing systems*, 30, 2017.

[LXHL24] Yingru Li, Jiawei Xu, Lei Han, and Zhi-Quan Luo. HyperAgent: A Simple, Scalable, Efficient and Provable Reinforcement Learning Framework for Complex Environments. In *Proceedings of the 41th International Conference on Machine Learning*, Proceedings of Machine Learning Research, 2024.

[OAC18] Ian Osband, John Aslanides, and Albin Cassirer. Randomized prior functions for deep reinforcement learning. *Advances in Neural Information Processing Systems*, 31, 2018.

[OVRRW19] Ian Osband, Benjamin Van Roy, Daniel J. Russo, and Zheng Wen. Deep exploration via randomized value functions. *Journal of Machine Learning Research*, 20(124):1–62, 2019.

References II

[OWA+23] Ian Osband, Zheng Wen, Seyed Mohammad Asghari, Vikranth Dwaracherla, Morteza Ibrahimi, Xiuyuan Lu, and Benjamin Van Roy. Epistemic neural networks. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[RVRK+18] Daniel J Russo, Benjamin Van Roy, Abbas Kazerouni, Ian Osband, Zheng Wen, et al. A tutorial on thompson sampling. *Foundations and Trends® in Machine Learning*, 11(1):1–96, 2018.