Bandit algorithms for demand-side management

WPI-Workshop on Stochastics, Statistics, Machine Learning and their Applications of Sustainable Finance and Energy Markets

Margaux Brégère - September 2023

LPSM
LABORATOIRE DE PROBABILITÉS STATISTIQUE & MODÉLISATION

Bandits algorithms

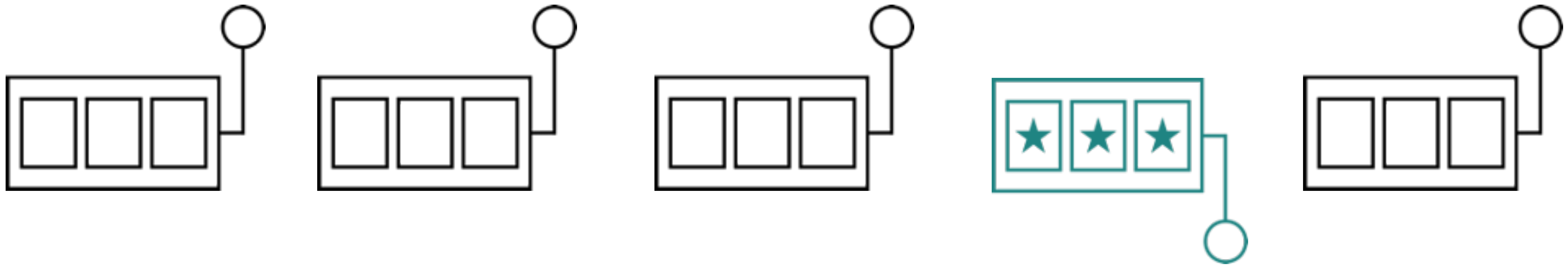
- Stochastic multi-armed bandit
- UCB algorithm
- Thompson sampling algorithm

Applications

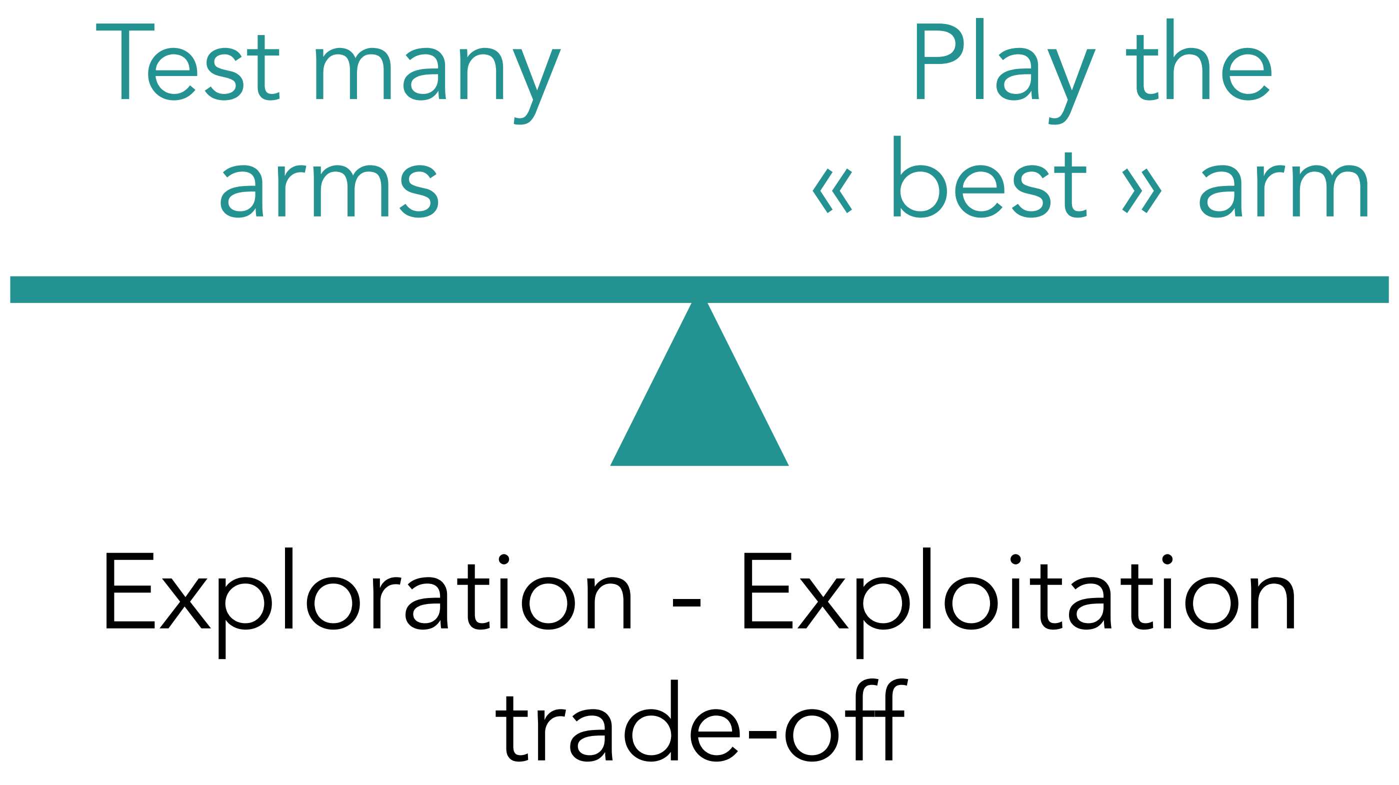
- Demand side management with incentive signals
- Control of flexible devices
- Hyper-parameter optimisation

Bandits algorithms

Stochastic multi-armed bandit



In a multi-armed bandit problem, a gambler facing a row of K slot machines - also called « one-armed bandits » - has to decide which machines to play to maximise her reward



Stochastic multi-armed bandit

Each arm (slot machine) k is defined by an unknown probability distribution ν_k

At each round $t = 1,\ldots,T$

- Pick a machine $I_t \in \{1,\ldots,K\}$
- Receive a reward with $g_t \,|\, I_t = k \sim \nu_k$

With $\mu_k = \mathbb{E}\left[\nu_k\right]$, to maximise the cumulative reward, we aim to minimise the regret, which is the difference, in expectation, between the cumulative reward of the best strategy and that of ours:

$$R_T = T\mu_{k^\star} - \mathbb{E}\left[\sum_{t=1}^{T} \mu_{I_t}\right] \quad \text{with} \quad k^\star \in \arg\max_k \mu_k$$

A good bandit algorithm has a sub-linear regret: $\dfrac{R_T}{T} \to 0$

Upper Confidence Bound algorithm[1]

Initialisation: pick each arm once

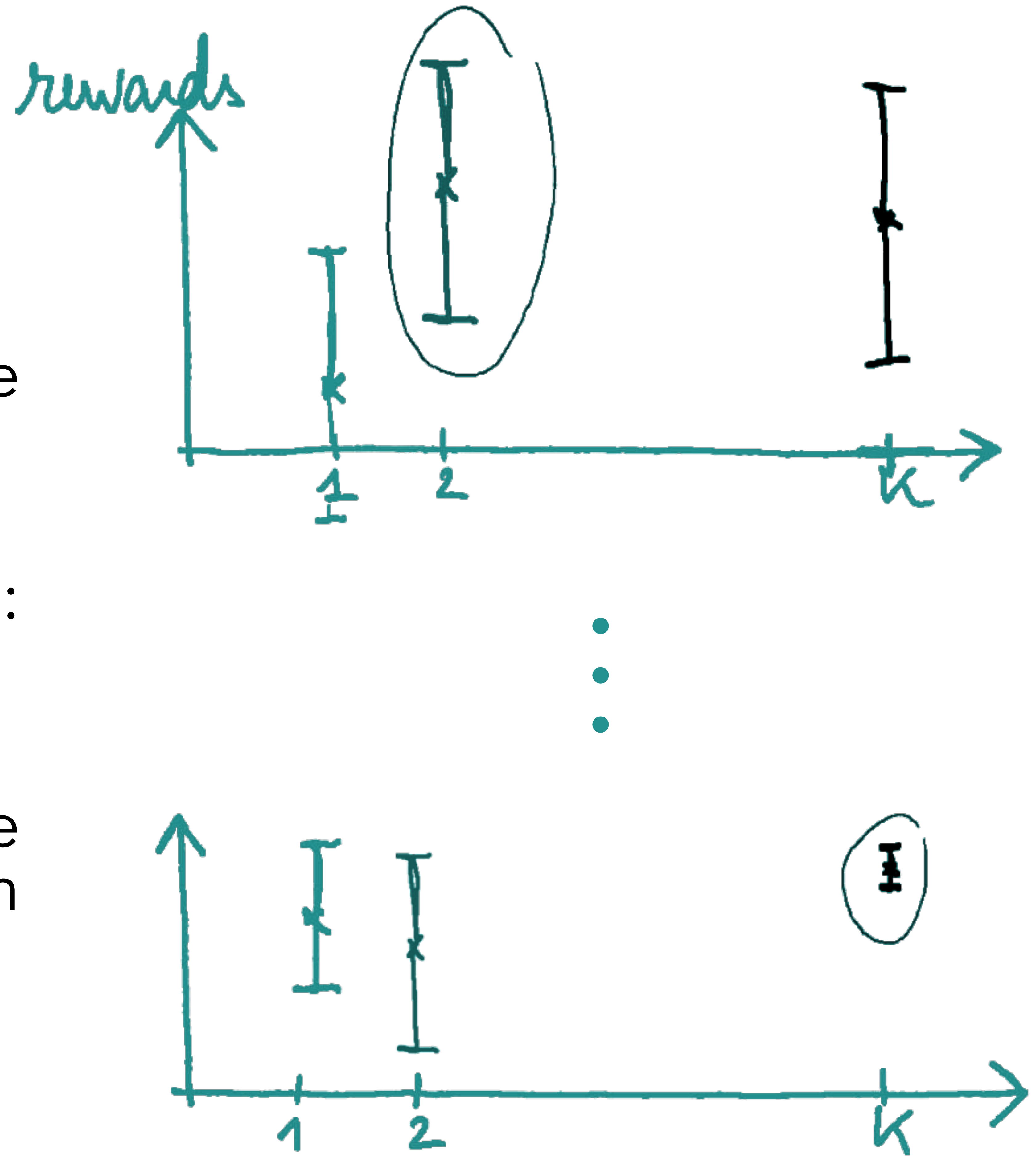
At each round $t = K + 1,\ldots,T$:

- Estimate the expected reward of each arm k with $\hat{\mu}_{k,t}$ the empirical mean of its past rewards

- Build some confidence intervals around these estimations: $\mu_k \in \left[\hat{\mu}_{k,t} - \alpha_{k,t}\,,\, \hat{\mu}_{k,t} + \alpha_{k,t}\right]$ with high probability

- Be optimistic and act as if the best possible probable reward was the true reward and choose the next arm accordingly

$$I_t \in \arg\max_k \left\{ \hat{\mu}_{k,t} + \alpha_{k,t} \right\}$$



[1] Auer et al. (2002) - *Finite-time analysis of the multiarmed bandit problem*

UCB regret bound

The empirical means based on past rewards are:

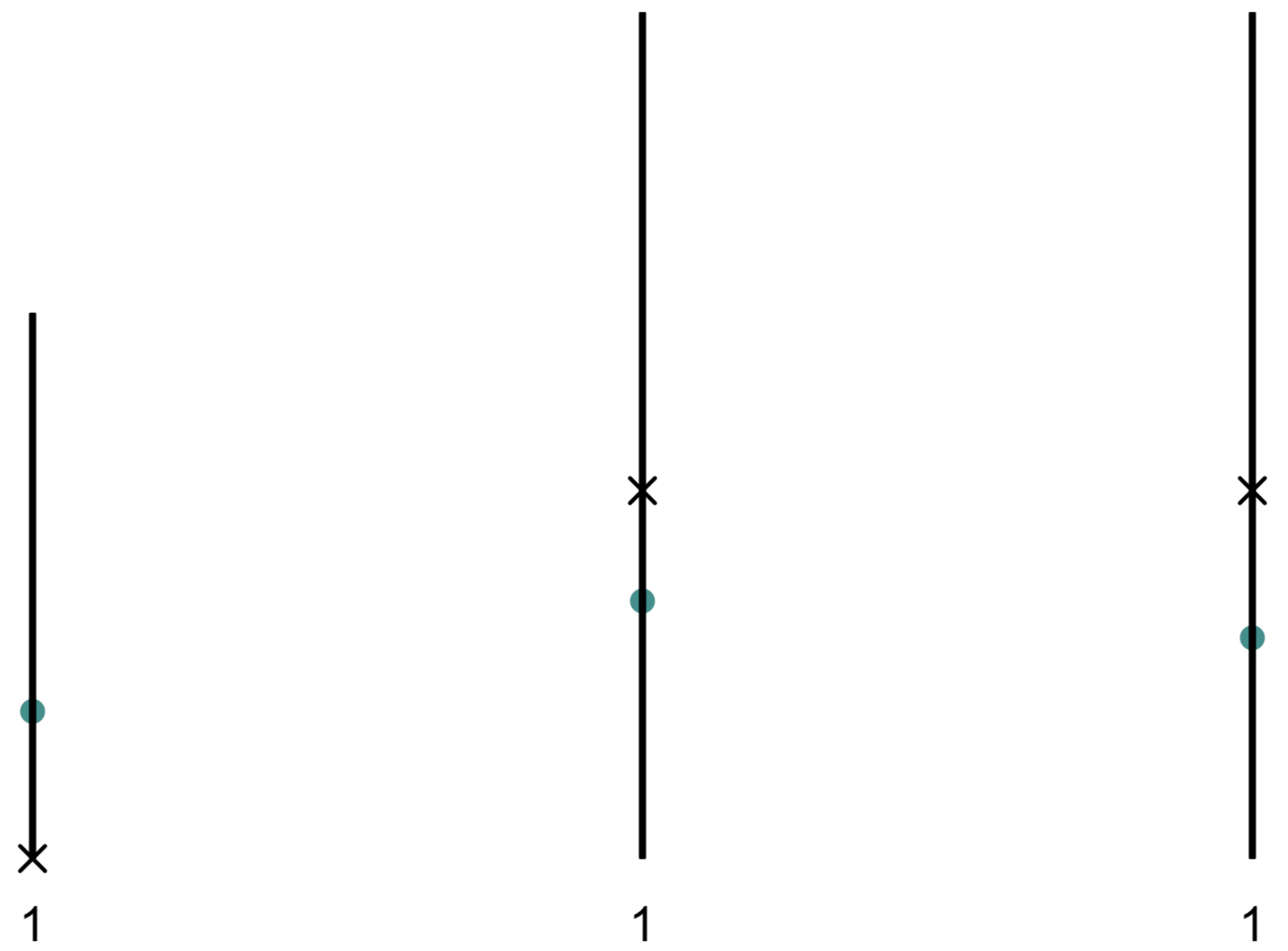
$$\hat{\mu}_{k,t} = \frac{1}{N_{k,t}} \sum_{s=1}^{t-1} g_s \mathbf{1}_{\{I_s = k\}} \text{ with } N_{k,t} = \sum_{s=1}^{t-1} \mathbf{1}_{\{I_s = k\}}$$

With Hoeffding-Azuma Inequality, we get

$$\mathbb{P}\left(\mu_k \in \left[\hat{\mu}_{k,t} - \alpha_{k,t} \,,\, \hat{\mu}_{k,t} + \alpha_{k,t}\right] \right) \geq 1 - t^{-3} \text{ with } \alpha_{k,t} = \sqrt{\frac{2 \log t}{N_{k,t}}}$$

And be optimistic ensures that

$$R_T \lesssim \sqrt{TK \log T}$$


1
1
1

Thompson sampling algorithm[2]

Initialisation: pick each arm once and set prior laws $\pi_{1,K}, \ldots, \pi_{K,K}$ on each arm

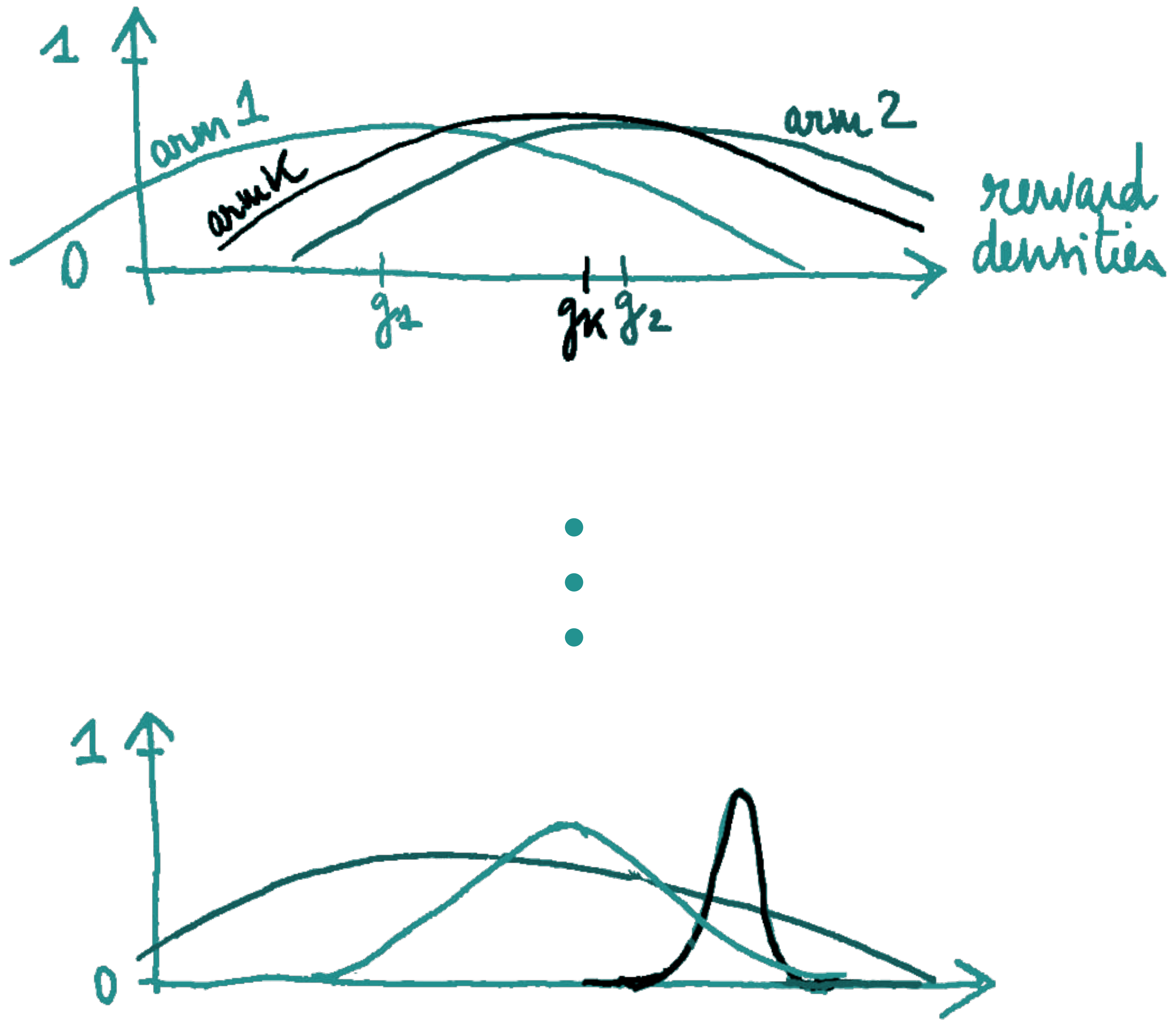
At each round $t = K+1, \ldots, T$

- Simulate reward $\hat{g}_{k,t} \sim \pi_{k,t}$

- Act as if the simulated rewards were the true rewards and choose the next arm accordingly

$$I_t \in \arg\min_k \hat{g}_{k,t}$$

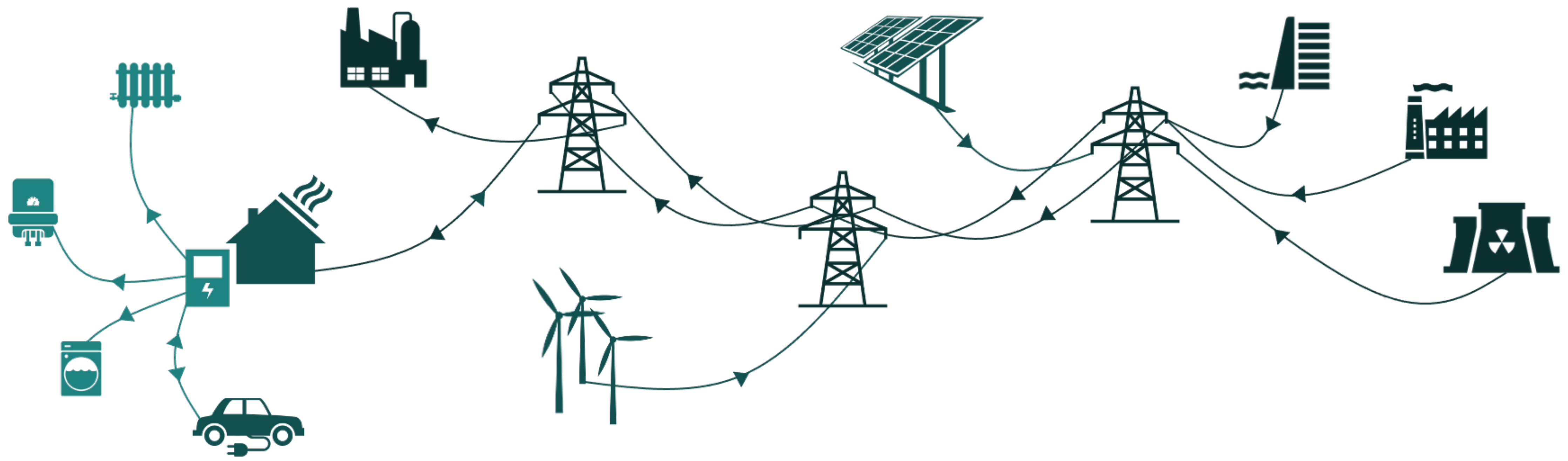
- Observe g_t and update prior law of arm I_t ($\pi_{i,t} = \pi_{i,t-1}$ if $i \neq I_t$)



[2] Thompson (1933) - *On the likelihood that one unknown probability exceeds another in view of the evidence of two samples*

Applications

Demand Side Management



As electricity is hard to store, balance between production and demand must be strictly maintained

Current solution: forecast demand and adapt production accordingly

- With the development of renewable energies, production becomes harder to adjust
- New (smart) meters provide access to data and instantaneous communication

Prospective solutions to manage demand response:

- Send incentive signals (electricity tariffs)
- Control flexible devices

With incentive signals (my PhD)

How to develop automatic solutions to chose incentive signals dynamically?

Exploration: learn consumer behaviour

Exploitation: optimize signal sending

« Smart Meter Energy Consumption Data in London Households »

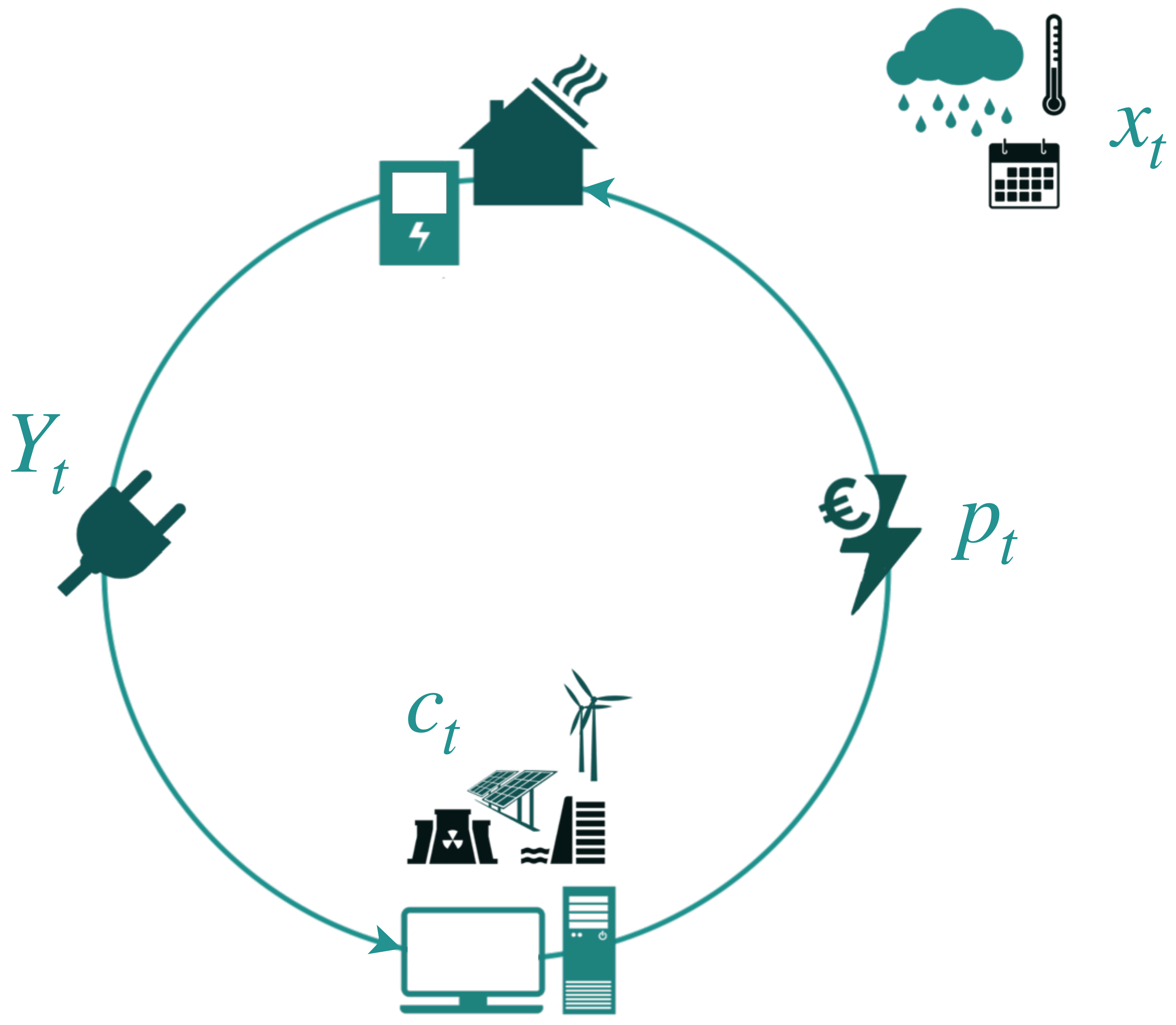
Demand side management protocol

At each round $t = 1, \dots, T$

- Observe a context x_t and a target c_t
- Choose price levels p_t
- Observe the resulting electricity demand

$$Y_t = f(x_t, p_t) + \mathrm{noise}(p_t)$$

and suffer the loss $\ell\left(Y_t, c_t\right)$



Assumptions:

- homogenous population, K tariffs, $p_t \in \Delta_K$
- $f\left(x_t, p_t\right) = \phi\left(x_t, p_t\right)^{\mathrm{T}}\theta$ with ϕ a known mapping function and θ an unknown vector to estimate
- $\mathrm{noise}(p_t) = p_t^{\mathrm{T}}\varepsilon_t$ with $\mathbb{V}\left[\varepsilon_t\right] = \Sigma$
- $\ell\left(Y_t, c_t\right) = \left(Y_t - c_t\right)^2$

Bandit algorithm for target tracking

Under these assumptions: $\mathbb{E}\left[\left(Y_t - c_t\right)^2 \middle| \text{past}, x_t, p_t\right] = \left(\phi(x_t, p_t)^\mathrm{T}\theta - c_t\right)^2 + p_t^\mathrm{T}\Sigma p_t$

☞ Estimate parameters θ and Σ to estimate losses and reach a bias-variance trade-off

Optimistic algorithm:

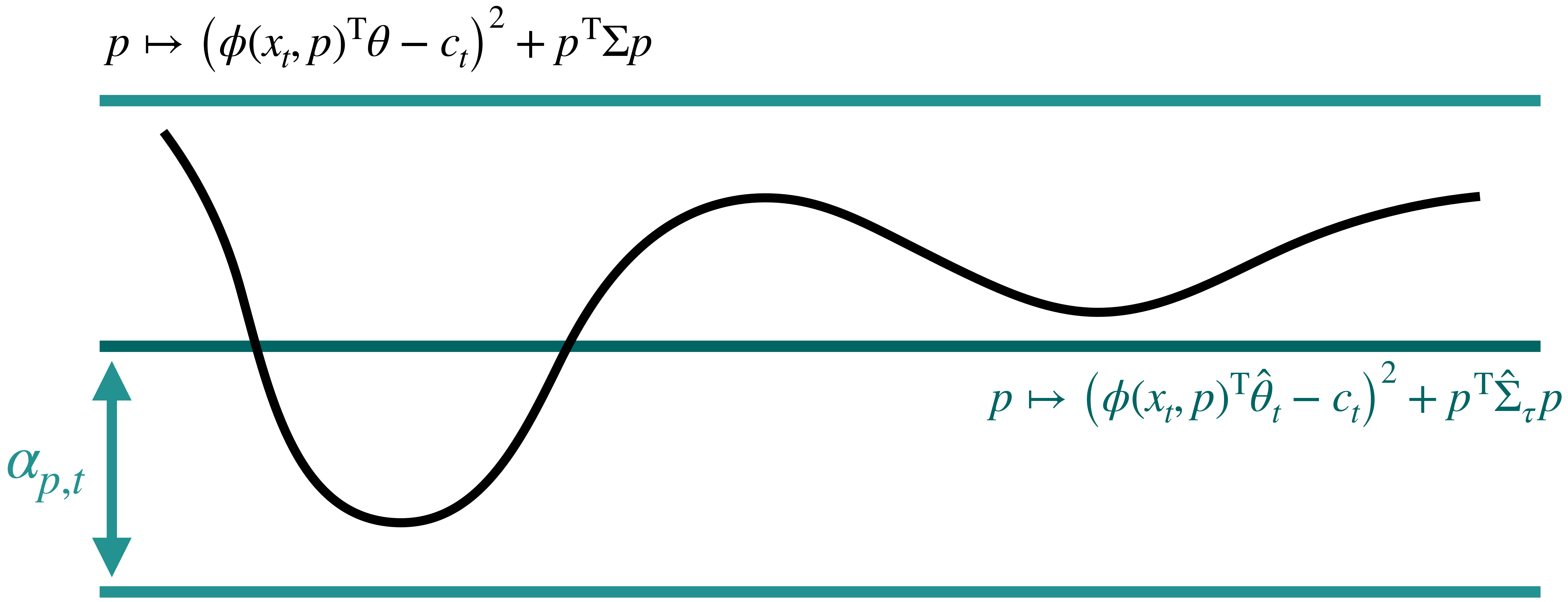
For $t = 1, \ldots, \tau$

- Select price levels deterministically to estimate Σ offline with $\hat{\Sigma}_\tau$

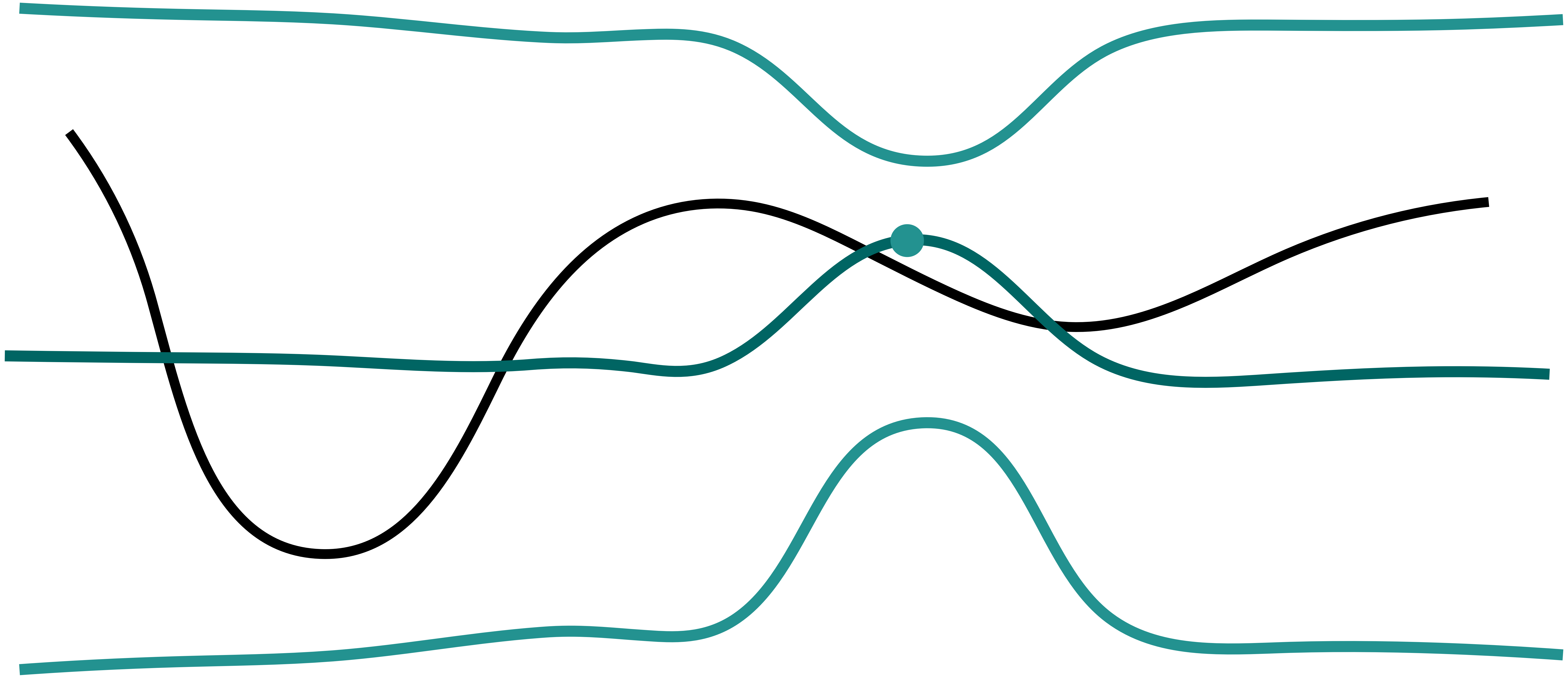
For $t = \tau + 1, \ldots, T$

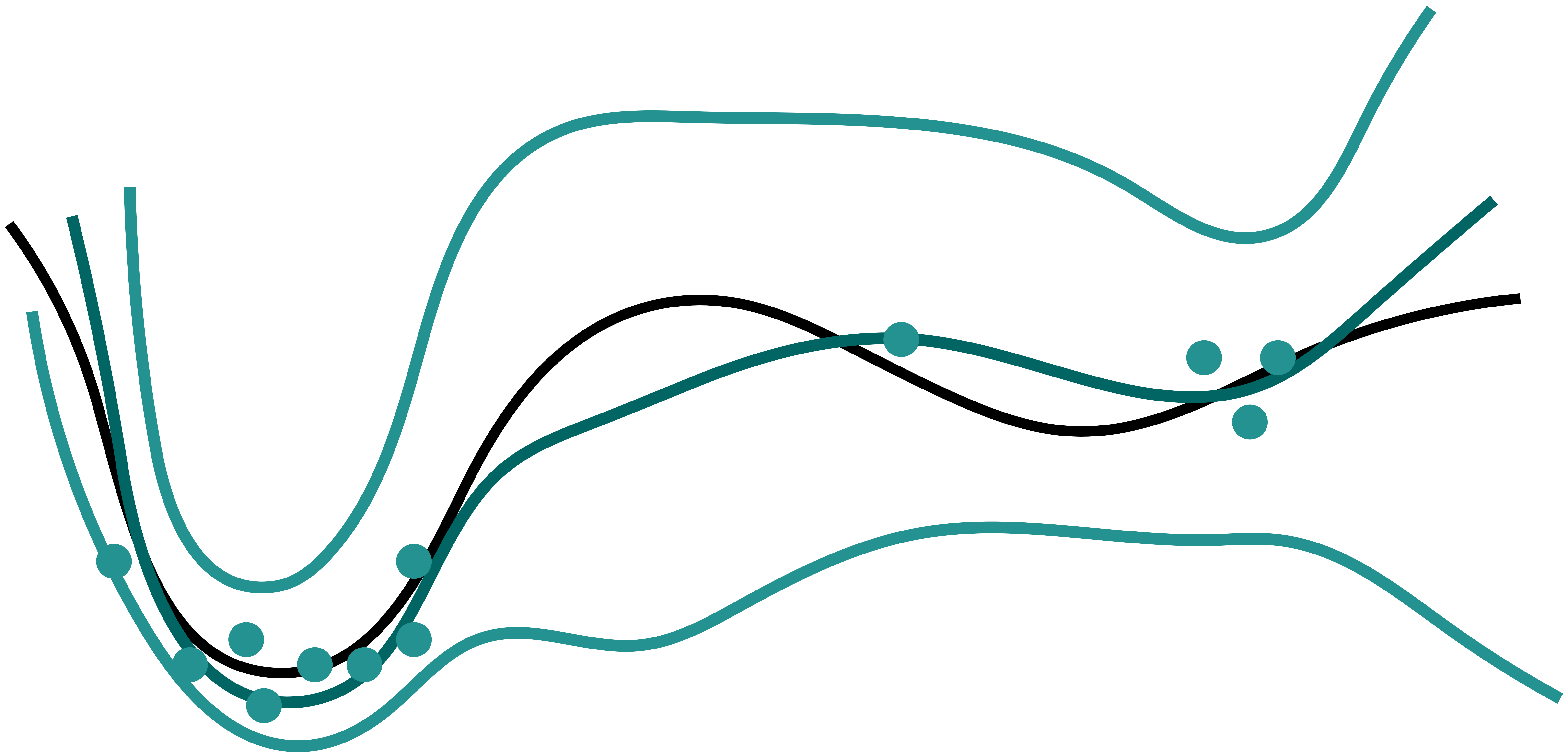
- Estimate θ based on past observation with $\hat{\theta}_t$ thanks to a Ridge regression
- Estimate future expected loss for each price level p: $\hat{\ell}_{p,t} = \left(\phi(x_t, p)^\mathrm{T}\hat{\theta}_t - c\right)^2 + p^\mathrm{T}\hat{\Sigma}_\tau p$
- Get confidence bound on these estimations: $|\hat{\ell}_{p,t} - \ell_p| \leq \alpha_{p,t}$
- Select price levels optimistically:

$$p_t \in \arg\min_p \left\{\hat{\ell}_{p,t} - \alpha_{p,t}\right\}$$



$p \mapsto \left(\phi(x_t, p)^{\mathrm{T}}\theta - c_t\right)^2 + p^{\mathrm{T}}\Sigma p$
$p \mapsto \left(\phi(x_t, p)^{\mathrm{T}}\hat{\theta}_t - c_t\right)^2 + p^{\mathrm{T}}\hat{\Sigma}_\tau p$
$\alpha_{p,t}$





The problem is a bit more complex: curves vary with time t

Regret bound[3]

We recall that: $R_T = \sum_{t=1}^{T} \left(\phi(x_t, p_t)^\mathrm{T}\theta - c_t\right)^2 + p_t^\mathrm{T}\Sigma p_t - \sum_{t=1}^{T} \min_p \left(\phi(x_t, p)^\mathrm{T}\theta - c_t\right)^2 + p^\mathrm{T}\Sigma p$

Theorem

For proper choices of confidence levels $\alpha_{p,t}$ and number of exploration rounds τ, with high probability

$$R_T \leq \mathcal{O}(T^{2/3})$$

If Σ is known, $R_T \leq \mathcal{O}\left(\sqrt{T} \ln T\right)$

Elements of proof

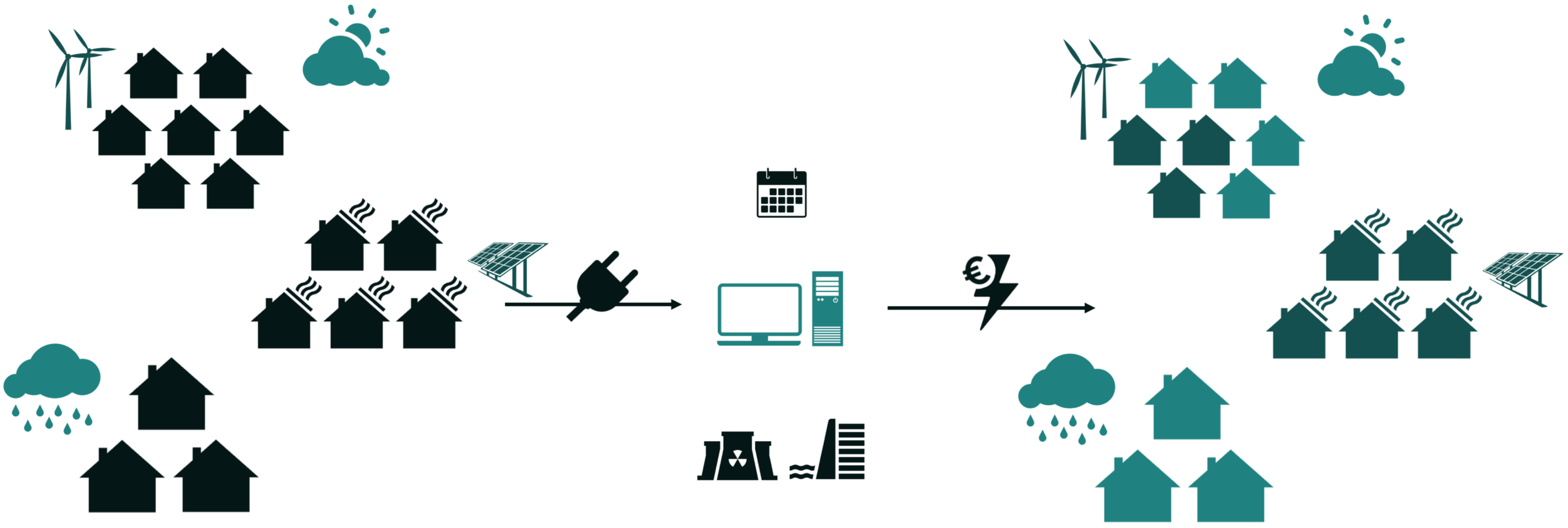
- Deviation inequalities on $\hat{\theta}_t$[4] and on $\hat{\Sigma}_\tau$
- Inspired from LinUCB regret bound analysis[5]

[3] Brégère et al. (2020) - Target tracking for contextual bandits: Application to demand side management

[4] Laplace's method on supermartingales: Abbasi-Yadkori et al. (2011) - Improved algorithms for linear stochastic bandits

[5] Chu et al. (2011) Contextual bandits with linear payoff functions

Extension: personalised demand side management



Flexible devices control (Bianca M. Moreno PhD)

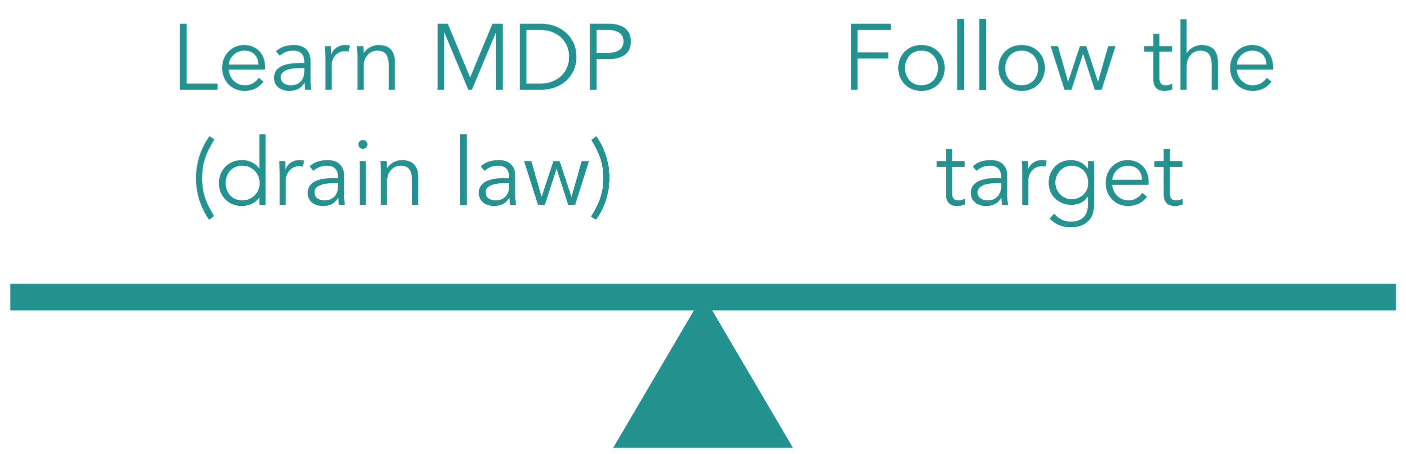
At each round $t = 1,\ldots,T$

- Observe a target c_t
- Send to all water-heaters the probability of switching on $p_t \in [0,1]$
- Observe the consumption

N water-heaters to be controlled without compromising service quality

Assumptions:

- N water-heaters with same characteristics
- Demand of water-heater i is zero if off and constant if on
- State $x_{i,t} = \left(\mathrm{Temperature}_t, \mathrm{ON/OFF}_t\right)$ of water-heater i follows an unknown Markov Decision Process (MDP)
- It is possible to control demand if the MDP is known[6]



At this stage, online learning seams enough: drain law does not depend on the signals sent

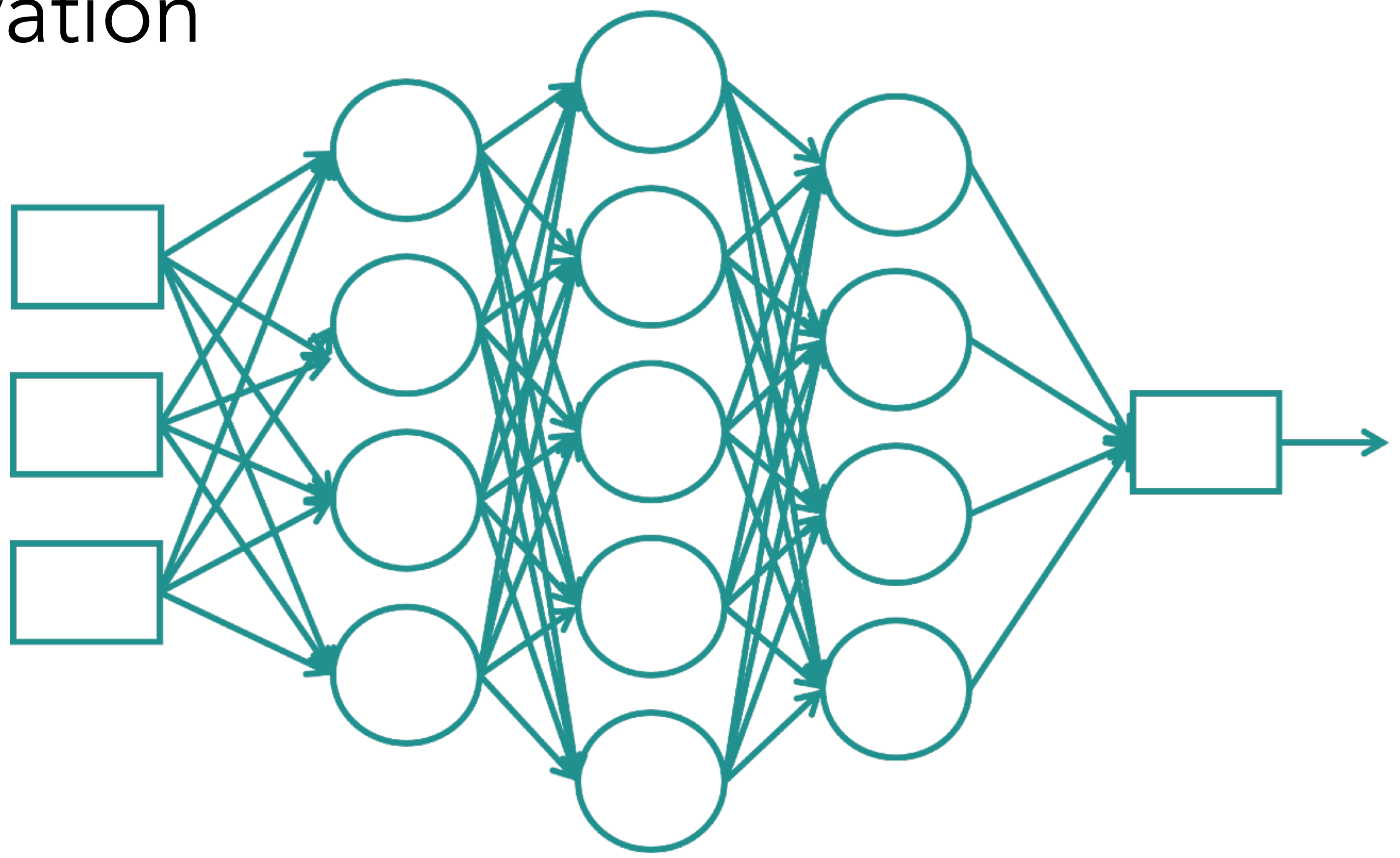
[6] Marin Moreno et al. (2023) - *A mirror descent approach for Mean Field Control applied to Demand-Side management*

Hyper-parameter optimisation (Julie Keisler PhD)

Train a neural network is expensive and time-consuming

Aim: for a set of hyper-parameters Λ (number of neurons, activation functions etc.) and a budget T, find the best neural network:

$$\arg\min_{\lambda\in\Lambda} \ell\left(f_\lambda(\mathscr{D}_{\text{TEST}})\right)$$



At each round $t = 1,\ldots,T$

- Choose hyper-parameters $\lambda_t \in \Lambda$
- Train network f_{λ_t} on $\mathscr{D}_{\text{TRAIN}}$
- Observe the forecast error $\ell_t = \ell\left(f_{\lambda_t}(\mathscr{D}_{\text{VALID}})\right)$

Output (best arm identification): $\arg\min_{f_{\lambda_t}} \ell\left(f_{\lambda_t}(\mathscr{D}_{\text{VALID}})\right)$

Train many neural network

Find the best neural network

KernelUCB[7]

Assumption: there exists a known mapping function Φ and an unknown parameter $\theta^\star$ such that

$$\ell\left(f_\lambda(\mathcal{D})\right) = \Phi(\lambda)^\mathrm{T}\theta^\star + \text{noise}$$

Optimistic algorithm:

Inputs: exploration parameter E

For $t = \tau + 1, \ldots, T$

- Estimate the loss function $\ell\left(f_\lambda(\mathcal{D})\right)$ based on past observation thanks to a kernel regression
- Estimate future expected loss for each price level λ: $\hat{\ell}_t(\lambda)$
- Get confidence bound on these estimations: $|\hat{\ell}_t(\lambda) - \ell\left(f_\lambda(\mathcal{D})\right)| \leq \alpha_{\lambda,t}$
- Select next hyper-parameters optimistically:

$$\lambda_t \in \arg\min_\lambda \left\{\hat{\ell}_t(\lambda) - E\alpha_{\lambda,t}\right\}$$

[7] Valko et al. (2013) - *Finite-time analysis of kernelised contextual bandits*

Thank you for your attention

QUESTIONS?

Experiments