Target Tracking for Contextual Bandits: Application to Power Consumption Steering

Pierre Gaillard (INRIA, Ecole Normale Supérieure, Paris)

Joint work with Margaux Brégère (EDF R&D), Gilles Stoltz (Univ. Paris-Sud) and Yannig Goude (EDF R&D)

July, 2019

EMS 2019, Palerme

Introduction

Electricity is hard to store at large scale.

→ Balance between production and demand should be maintained at any time to avoid

- physical risks: network reconfiguration,...
- financial risks.

Production  Demand

Typical solution: forecast electricity consumption then adapt the production accordingly.

Limitation:

- Renewable energies subject to climate → hard to adjust the production
- Non-flat consumption is costly -> avoid peaks

What about reversing the process? Choose the production and influence consumers consumptions by sending signals (price)?
→ How to optimize these signals and learn clients behaviors?

Data set: price sensitive clients to influence their electricity consumption

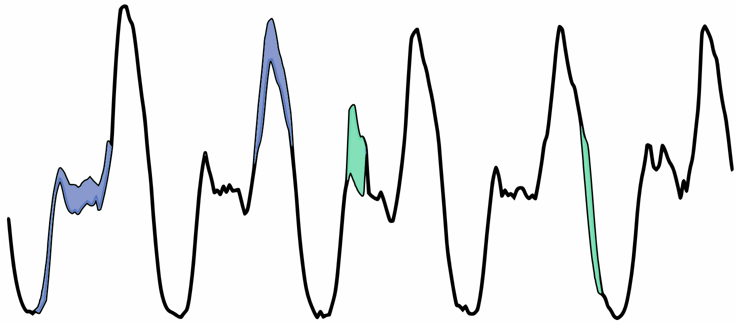
We consider the public data set provided by **UK power network**

"Smart Meter Energy Consumption Data in London Households"

- Individual consumption at half-an-hour intervals throughout 2013
- 1100 price-sensitive clients (3 price levels: high, low, normal)
- 3400 clients on flat-rate price level

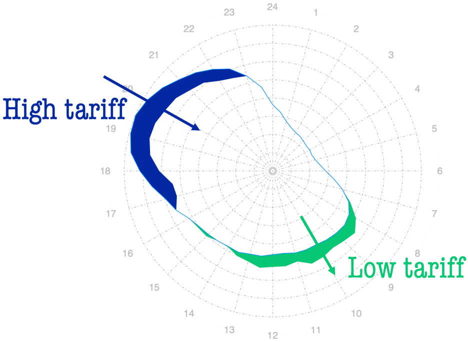
Price sensitive clients

Price sensitive clients: 3 price levels (High, Low, Normal) on five days



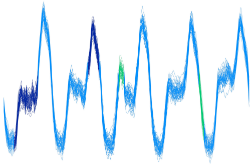
Price sensitive clients

Price-sensitivity depends on contextual variables (climate, temporal)



Simulator

The data set contains the **consumption of customers for some chosen price levels** along 2013.



Yet, we do not know what would have been their consumptions for different price signals at the same times.

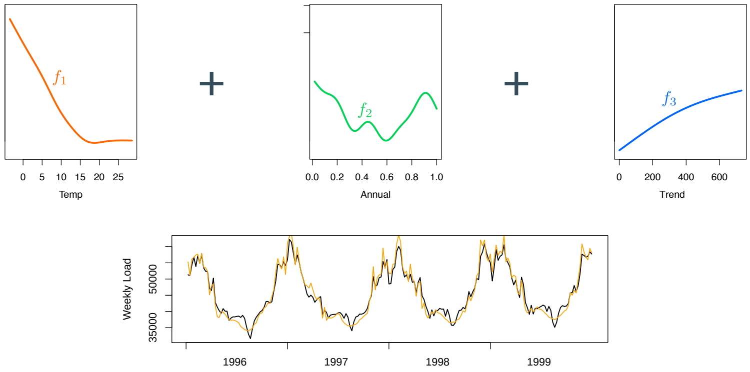
To run our experiments, we build a simulator assuming homogeneous customers:

Context + Price level $\longrightarrow$ Global consumption

Based on Generalized Additive Model.

Generalized Additive Model (Hastie et al. 1990; Pierrot et al. 2011)

$$Y_t = f_1(\mathrm{Temp}_t) + f_2(\mathrm{AnnualPos}_t) + f_3(\mathrm{Trend}_t) + \cdots + \varepsilon_t$$



Objective: optimize price signals and learn behaviors

Optimize price signals sent to price-sensitive clients to influence their consumption.

How ? Through new communication tools such as smart meters.

A sequential problem: at each time step $t \geq 1$

- observe contextual variables (weather, calendar)
- get a target consumption c_t
- choose price signal
- observe the global consumption of the clients
- update the strategy

Two simultaneous objectives: learn client behaviors and optimize price signals.

Exploration vs Exploitation

→ Multi-armed bandit theory (active learning)

Multi-armed bandit (Thomson, 2933)

A simple stochastic model:

- K arms (actions: here price signals)
- Each arm k is associated an unknown probability distribution with mean μ_k

μ_1 μ_2 μ_3 μ_4 μ_5

Setting: sequentially pick an arm k_t and get reward $X_{k_t,t}$ with mean μ_{k_t}

Goal: maximize the expected cumulative reward

$$\mathbb{E}\left[\sum_{t=1}^{T} X_{k_t,t}\right]$$

Exploration vs Exploitation trade-off.

Bandit applications

Maximize one's gains in casino? Hopeless ...

μ_1 μ_2 μ_3 μ_4 μ_5

Historical motivation (Thomson, 1933): clinical trials, for each patient t in a clinical study

- choose a treatment k_t
- observe response to the treatment $X_{k_t,t}$

Goal: maximize the number of patient healed (or find the best treatment)

Successful because of many applications coming from Internet: recommender systems, online advertisements,...

Objective of multi-armed bandit

Goal: maximize the expected cumulative reward

$$\mathbb{E}\left[\sum_{t=1}^{T} X_{k_t,t}\right]$$

Oracle: always play the arm maximizing the expected reward

$$k^* = \operatorname*{arg\,max}_{k\in\{1,\dots,K\}} \mu_k \quad \text{with mean} \quad \mu^* = \max_k \mu_k \,.$$

Can we be almost as good as the oracle?

Performance measure: regret

$$R_T = T\mu^* - \mathbb{E}\left[\sum_{t=1}^{T} X_{k_t,t}\right]$$

Maximizing reward = minimizing regret

Good bandit algorithm: sublinear regret

$$\frac{R_T}{T} \xrightarrow[t\to\infty]{} 0$$

Bandit algorithms: Upper-Confidence-Bound (UCB) (Lai et al. 1985)

Upper-Confidence-Bound strategy: explore and exploit sequentially all along the experiment

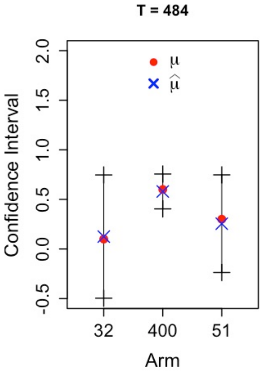
- for each arm, build **a confidence interval** on the mean μ_k based on past observations

$$I_t(k) = [LCB_t(k), UCB_t(k)]$$

LCB = Lower Confidence Bound
UCB = Upper Confidence Bound

- **be optimistic**: act as if the best possible rewards where the true rewards and choose the next arm accordingly

$$k_t = \arg\max_{k \in \{1,\dots,K\}} UCB_t(k)$$



Bandit algorithms: Upper-Confidence-Bound (UCB) (Lai et al. 1985)

Upper-Confidence-Bound strategy: explore and exploit sequentially all along the experiment

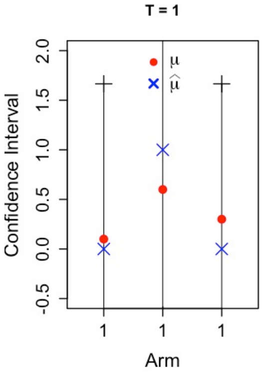
- for each arm, build **a confidence interval** on the mean μ_k based on past observations

$$I_t(k) = [LCB_t(k), UCB_t(k)]$$

LCB = Lower Confidence Bound
UCB = Upper Confidence Bound

- **be optimistic**: act as if the best possible rewards where the true rewards and choose the next arm accordingly

$$k_t = \arg\max_{k \in \{1,\ldots,K\}} UCB_t(k)$$



Bandit algorithms: Upper-Confidence-Bound (UCB) (Lai et al. 1985)

Upper-Confidence-Bound strategy: explore and exploit sequentially all along the experiment

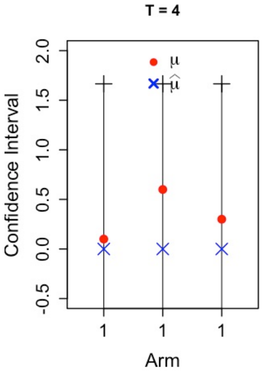
- for each arm, build **a confidence interval** on the mean μ_k based on past observations

$$I_t(k) = [LCB_t(k), UCB_t(k)]$$

  LCB = Lower Confidence Bound
  UCB = Upper Confidence Bound

- **be optimistic**: act as if the best possible rewards where the true rewards and choose the next arm accordingly

$$k_t = \arg\max_{k \in \{1,\ldots,K\}} UCB_t(k)$$



Bandit algorithms: Upper-Confidence-Bound (UCB) (Lai et al. 1985)

Upper-Confidence-Bound strategy: explore and exploit sequentially all along the experiment

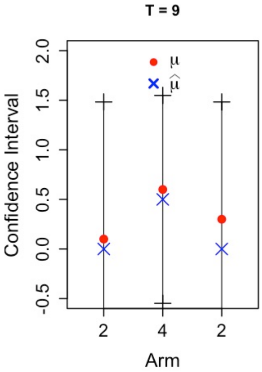
- for each arm, build **a confidence interval** on the mean μ_k based on past observations

$$I_t(k) = [LCB_t(k), UCB_t(k)]$$

LCB = Lower Confidence Bound
UCB = Upper Confidence Bound

- **be optimistic**: act as if the best possible rewards where the true rewards and choose the next arm accordingly

$$k_t = \arg\max_{k \in \{1,\ldots,K\}} UCB_t(k)$$



Bandit algorithms: Upper-Confidence-Bound (UCB) (Lai et al. 1985)

Upper-Confidence-Bound strategy: explore and exploit sequentially all along the experiment

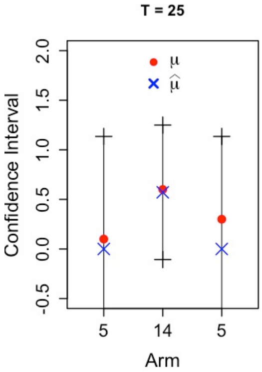
- for each arm, build **a confidence interval** on the mean μ_k based on past observations

$$I_t(k) = [LCB_t(k), UCB_t(k)]$$

  LCB = Lower Confidence Bound
  UCB = Upper Confidence Bound

- **be optimistic**: act as if the best possible rewards where the true rewards and choose the next arm accordingly

$$k_t = \underset{k \in \{1,\ldots,K\}}{\arg\max}\, UCB_t(k)$$



Bandit algorithms: Upper-Confidence-Bound (UCB) (Lai et al. 1985)

Upper-Confidence-Bound strategy: explore and exploit sequentially all along the experiment

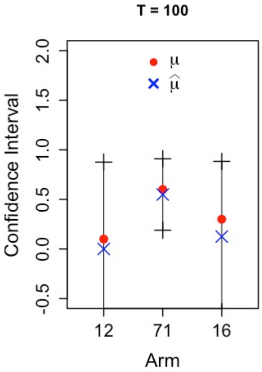
- for each arm, build **a confidence interval** on the mean μ_k based on past observations

$$I_t(k) = [LCB_t(k), UCB_t(k)]$$

LCB = Lower Confidence Bound
UCB = Upper Confidence Bound

- **be optimistic**: act as if the best possible rewards where the true rewards and choose the next arm accordingly

$$k_t = \arg\max_{k \in \{1,\dots,K\}} UCB_t(k)$$



Bandit algorithms: Upper-Confidence-Bound (UCB) (Lai et al. 1985)

Upper-Confidence-Bound strategy: explore and exploit sequentially all along the experiment

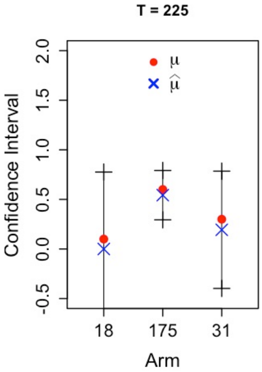
- for each arm, build **a confidence interval** on the mean μ_k based on past observations

$$I_t(k) = [LCB_t(k), UCB_t(k)]$$

LCB = Lower Confidence Bound
UCB = Upper Confidence Bound

- **be optimistic**: act as if the best possible rewards where the true rewards and choose the next arm accordingly

$$k_t = \arg\max_{k \in \{1,\dots,K\}} UCB_t(k)$$



Bandit algorithms: Upper-Confidence-Bound (UCB) (Lai et al. 1985)

Upper-Confidence-Bound strategy: explore and exploit sequentially all along the experiment

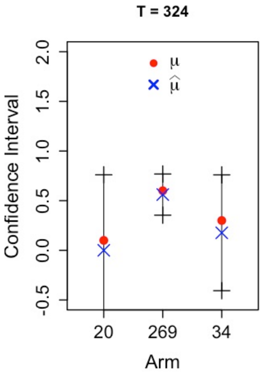
- for each arm, build **a confidence interval** on the mean μ_k based on past observations

$$I_t(k) = [LCB_t(k), UCB_t(k)]$$

LCB = Lower Confidence Bound
UCB = Upper Confidence Bound

- **be optimistic**: act as if the best possible rewards where the true rewards and choose the next arm accordingly

$$k_t = \underset{k \in \{1,\ldots,K\}}{\arg\max}\, UCB_t(k)$$



Bandit algorithms: Upper-Confidence-Bound (UCB) (Lai et al. 1985)

Upper-Confidence-Bound strategy: explore and exploit sequentially all along the experiment

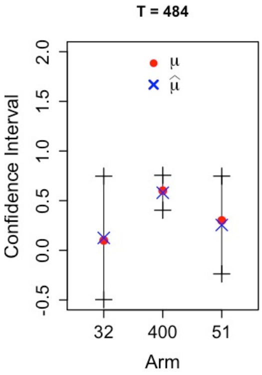
- for each arm, build **a confidence interval** on the mean μ_k based on past observations

$$I_t(k) = [LCB_t(k), UCB_t(k)]$$

LCB = Lower Confidence Bound
UCB = Upper Confidence Bound

- **be optimistic**: act as if the best possible rewards where the true rewards and choose the next arm accordingly

$$k_t = \arg\max_{k \in \{1,\ldots,K\}} UCB_t(k)$$



Upper-Confidence-Bound result (Auer et al. 2002)

Choice of the upper-bound

$$UCB_k(t) = \hat{\mu}_k(t) + \sqrt{\frac{2 \log t}{N_k(t)}}$$

For UCB algorithm:

$$R_T = T\mu^* - \mathbb{E}\left[\sum_{t=1}^{T} X_{k_t,t} \right] \lesssim \sqrt{T \log T}$$

Setting 1: Toy setting with non-realistic assumptions

Back to our problem: optimize tariffs to track target consumption

Assumptions:

- no impact of contextual variables (weather, temporal,...) on price-sensitivity
- choose at each time the same tariff for all clients

Setting 1

K different tariffs

$\mu_1, \dots, \mu_K$: global consumption laws associated with each tariff

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

- receive target consumption $c_t > 0$
- choose tariff $k_t \in \{1, \dots, K\}$
- observe global consumption Y_t with $Y_t \sim \mu_{k_t}$
- suffer loss $\ell(Y_t, c_t) \in [0, 1]$

Algorithm for setting 1: inspired from UCB

Initial stage: Choose each tariff ones $k_t = t$ for $t = 1, \dots, K$ For $t \geq K+1$

1. Compute empirical loss of each tariff for target c_t:

$$\hat{\ell}_{k,t} \in \frac{1}{N_k(t)} \sum_{s=1}^{t} \ell(Y_s, c_t) \mathbb{1}_{k_s = k}$$

2. Choose tariff with optimistic loss

$$k_t \in \underset{k \in \{1,\dots,K\}}{\arg\min} \left\{ \hat{\ell}_{k,t} - \sqrt{\frac{2 \log t}{N_k(t)}} \right\}.$$

Theorem

$$R_T = \mathbb{E}\left[\sum_{t=1}^{T} \ell_{k_t,t} - \min_k \ell_{k,t} \right] \lesssim \sqrt{T \log T}$$

where $\ell_{k,t} = \ell(Y, c_t)$ with $Y \sim \mu_k$.

→ Average loss is approximatively the average loss of the best possible tariffs to track c_t on the long term.

Model for simulations

We assume that the context does not impact customers reaction to tariff changes: additive effect.

Consumption = Known deterministic dependence on context + Random tariff effect

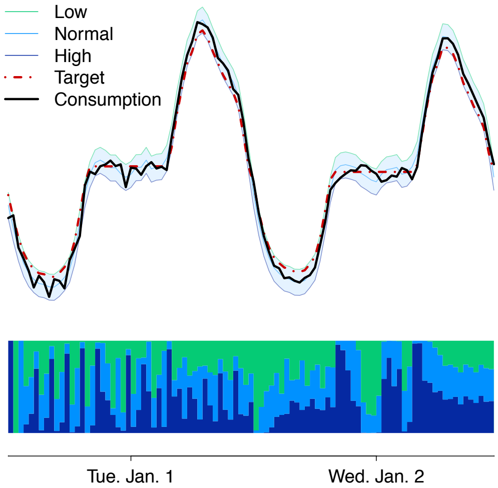
We model the consumption for a chosen tariff k as

$$Y_{k,t} = f(x_t) + X_{k,t}$$

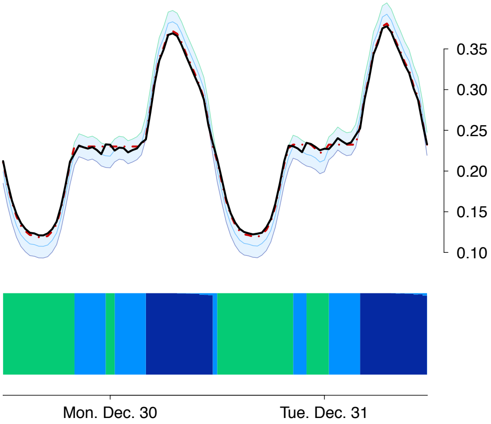
where $X_{k,t} \sim \mu_k$ is an additive random variable modeling the impact of tariff k (negative for high tariff and positive for low tariff).

$f(x_t)$ is fitted before-hand on the dataset and assumed to be known.

Simulations (Early stage: exploration)



Simulations (End: exploitation)



Limitations of this toy setting

Consumption = Known deterministic dependence on context + Random tariff effect

Limitations of previous setting:

- discrete: a single tariff k_t needs to be chosen for all consumers
  → we might want intermediate scenarios
  Solution: assume homogeneous customers and choose proportion of customers associated with each tariff

$$p_t \in [0,1]^K \quad \text{such that} \quad \sum_{k=1}^{K} p_t(k) = 1$$

- Context independence of tariff impacts: additive effect
- Known dependence of average consumption on context

Can we remove all these assumptions by considering an algorithm that learns how to optimize p_t in a general model?

General setting with contexts

At instance t, the electricity provider sends tariff k to a share $p_{t,k}$ of the customers.

We assume that the mean consumption observed equals

$$Y_{t,p_t} = \sum_{k=1}^{K} p_{t,k}\, \varphi(x_t, k) + \text{noise}\,.$$

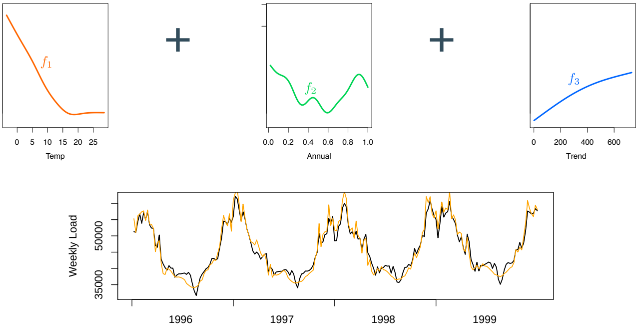
where φ is some function associating with a context x_t and a tariff k an expected consumption $\varphi(x_t, k)$. We assume that there exists some unknown $\theta \in \mathbb{R}^d$ and some known transfer function ϕ such that $\varphi(x_t, j) = \phi(x_t, j)^\top \theta$:

$$Y_{t,p_t} = \phi(x_t, p_t)^\top \theta + \text{noise}\,.$$

Transfer function ϕ is known, Price levels p_t are to be optimized, Parameter θ is to be estimated.

Particular case: generalized Additive Model

$$Y_{t,p_t} = f_1(\mathrm{Temp}_t, p_t) + f_2(\mathrm{AnnualPos}_t, p_t) + f_3(\mathrm{Trend}_t, p_t) + \cdots + \varepsilon_t$$



Protocol: Target tracking for contextual bandits

Inputs

Parametric context set $\mathcal{X}$

Set of legible convex weights $\mathcal{P}$

Bound on mean consumptions C

Transfer function $\phi : \mathcal{X} \times \mathcal{P} \to \mathbb{R}^d$

Unknown parameter: $\theta \in \mathbb{R}^d$

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

Observe a context $x_t \in \mathcal{X}$ and a target $c_t \in (0, C)$

Choose an allocation of price levels $p_t \in \mathcal{P}$

Observe a resulting mean consumption

$$Y_{t,p_t} = \phi(x_t, p_t)^\top \theta + Noise$$

Suffer a loss $\ell_{p_t,t} = (Y_{t,p_t} - c_t)^2$

End for

Aim: Minimize the regret

$$R_T = \sum_{t=1}^{T} \left(\phi(x_t, p_t)^\top \theta - c_t\right)^2 \quad - \sum_{t=1}^{T} \min_{p_t^* \in \mathcal{P}} \left(\phi(x_t, p_t^*)^\top \theta - c_t\right)^2$$

Optimistic Algorithm for tracking target with context

Inspired from LinUCB (Li et al. 2010)

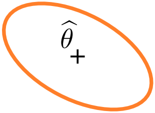
1. **Estimate the parameter** θ from observations

$$\hat{\theta}_t = V_t^{-1} \sum_{s=1}^{t-1} Y_{s,p_s} \phi(x_s, p_s) \quad \text{where} \qquad V_t = \lambda I_d + \sum_{s=1}^{t-1} \phi(x_s, p_s) \phi(x_s, p_s)^\top .$$

2. **Estimate the future loss** $\ell_{p,t}$ of each price level p

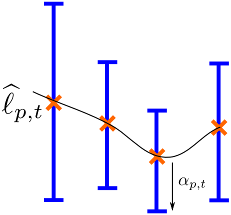
$$\hat{\ell}_{p,t} = \left(\phi(x_t, p)^\top \hat{\theta}_t - c_t \right)^2 .$$

2. **Build confidence set** for θ



$$\left\| \hat{\theta}_t - \theta \right\|_{V_t} \leq B_t .$$

3. **Get confidence bound for losses** of each price level



$$|\ell_{p,t} - \hat{\ell}_{p,t}| \leq \alpha_{p,t} .$$

4. Select price level **optimistically**

$$p_t \in \operatorname*{arg\,min}_{p \in \mathcal{P}} \left\{ \hat{\ell}_{t,p} - \alpha_{t,p} \right\} .$$

Theoretical guarantee

Model 1:

$$Y_{t,p_t} = \phi(x_t, p_t)^\top \theta + \text{noise}\,.$$

Noise assumption: noise $= p_t^\top \varepsilon_t$ where ε_t are i.i.d. subGaussian variables in $\mathbb{R}^K$ with covariance Γ.

Goal: choose p_t sequentially to track target c_t

Theorem

For proper choices of confidence levels $\alpha_{p,t}$, B_t, regularization λ, and subGaussian noise with high probability the regret is upper-bounded as

$$R_T = \sum_{t=1}^{T} \left(\phi(x_t, p_t)^\top \theta - c_t\right)^2 \quad - \sum_{t=1}^{T} \min_{p \in \mathcal{P}} \left(\phi(x_t, p_t)^\top \theta - c_t\right)^2 \lesssim T^{2/3}$$

If the covariance Γ of the noise is known, $R_T \lesssim \sqrt{T}$.

Remarks on the theorem

Bias-Variance trade-off. If the noise depends on the tariffs (more volatility for non-normal tariffs), we should take it into account as a bias-variance trade-off

$$\ell_{p,t} = \underbrace{\left(\phi(x_t, p_t)^\top \theta - c_t\right)^2}_{\text{bias}} + \text{Variance of price level } p_t$$

Sophisticated price level sets. We might not want to allocate simultaneously high and low price levels

$$\mathcal{P} = \left\{p \in [0,1]^3 : \; p_1 p_3 = 0\right\}$$

Limitation. The optimization problem $p_t \in \arg\min_{p \in \mathcal{P}} \left\{\hat{\ell}_{t,p} - \alpha_{t,p}\right\}$ is nonconvex and hard to solve.

Faster rate with additional assumptions

Assumptions:

1. The noise does not depend on the tariff

$$Y_{t,p_t} = \phi(x_t, p_t)^\top \theta + \varepsilon_t \,. \qquad \text{where} \quad \varepsilon_t \quad \text{i.i.d. subGaussian}$$

2. The target is attainable:

$$\forall t \geq 1, \quad \exists p \in \mathcal{P} \quad \phi(x_t, p) = c_t \,.$$

Theorem

Under these assumptions, with well-calibrated parameters, the regret is upper-bounded with high probability as

$$R_T = O\big((\log T)^2\big) \,.$$

Data set: price sensitive clients to influence their electricity consumption

We consider the public data set provided by **UK power network**

"Smart Meter Energy Consumption Data in London Households"

- Individual consumption at half-an-hour intervals throughout 2013
- 1100 price-sensitive clients (3 price levels: high, low, normal)
- 3400 clients on flat-rate price level

Design of the experiment

Simulator:

$$Y_t = f_1(\text{Temp}_t, \text{hour}_t) + f_2(\text{AnnualPos}_t, \text{hour}_t) + f_3(\text{Trend}_t, \text{hour}_t) + f_4(\text{weekday}_t) + \text{Tariff effect} + \text{noise}$$

Assumption: exogenous factors do not impact customers' reaction to tariff changes + known covariance of the noise.

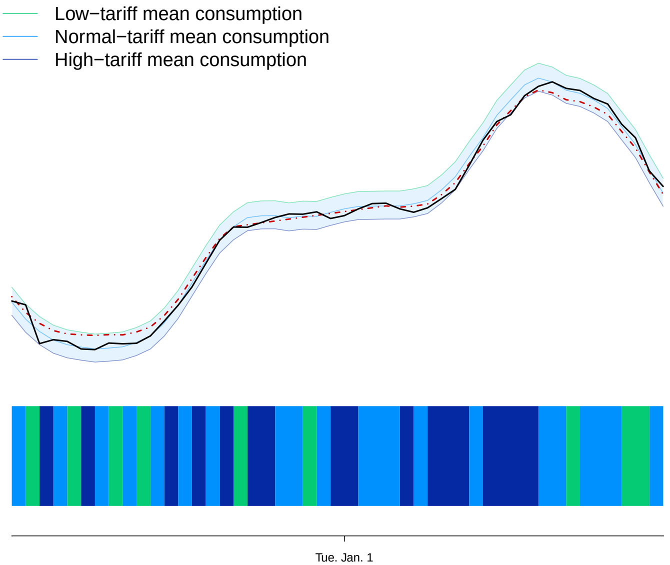
Training period: The model $(f_1, \ldots, f_4)$ is pre-trained on one year of past historical data with normal tariff only.

Testing period: the provider starts exploring the effects of tariffs for an additional month and freely picks the pt according to our algorithm.

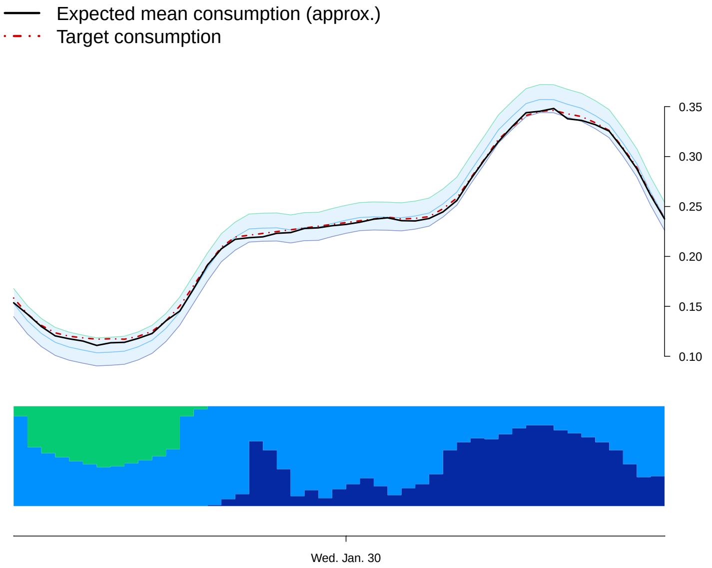
Target creation: we focus on attainable targets. To smooth consumption, we pick high c_t during the night and small c_t in the evening.

Experiments are repeated 200 times.

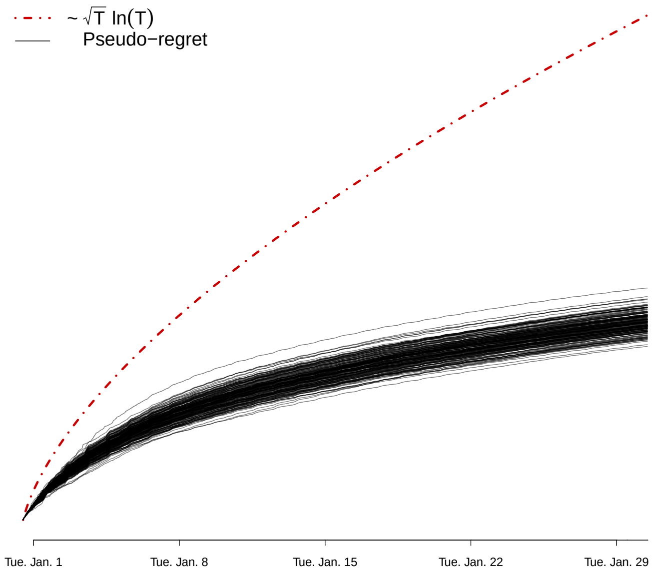
Results with noise depending on tariff (Early stage – exploration)



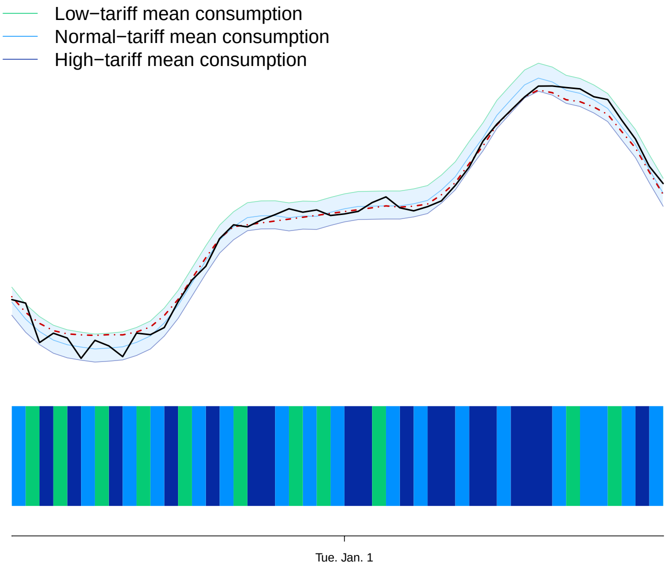
Results with noise depending on tariff (End – exploitation)



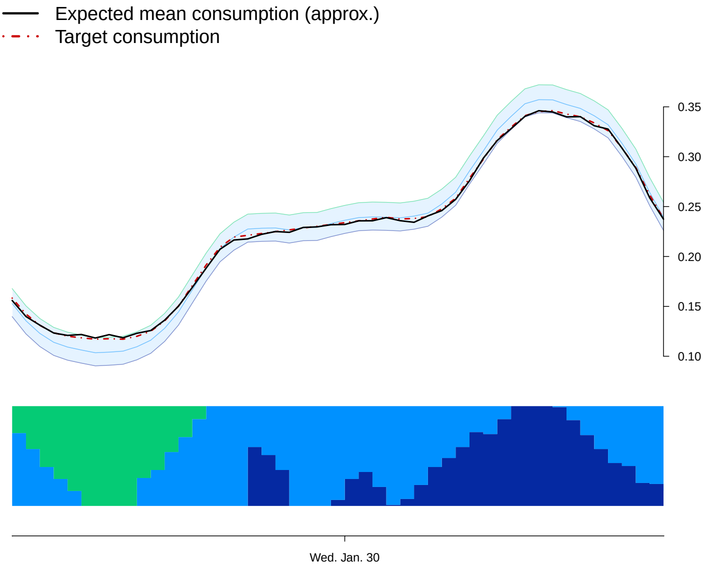
Results with noise depending on tariff (Regret)



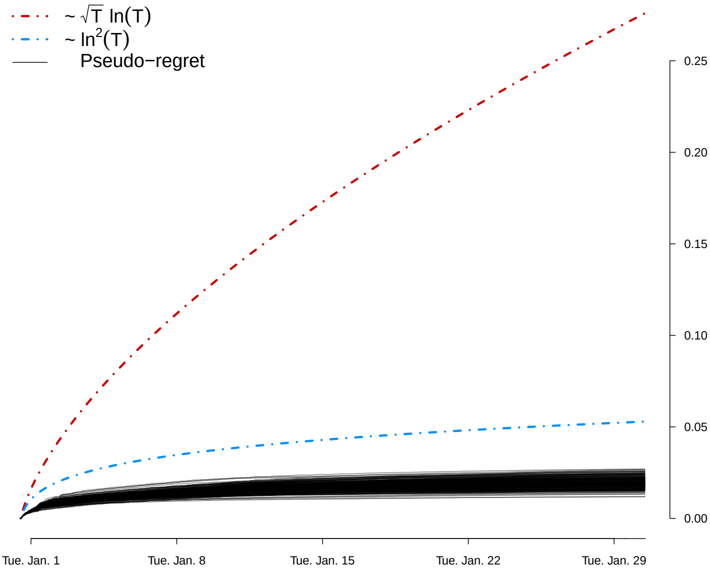
Results with noise not depending on tariff (Early stage – exploration)



Results with noise not depending on tariff (End – exploitation)



Results with noise not depending on tariff (Regret)



Conclusion and perspectives

Summary

- Design, implement and test an efficient algorithm with theoretical guaranties to track a target consumption under basic assumptions.

What's next?

- More experiments, simulations
- Non homogeneous consumers: create client clusters to send individual signals (device dependent, clients with battery) and improve power consumption control.
- Network configuration: hierarchical structure
- More complex models? Anticipation of future high prices, ...
- Operational constraints
- How to choose target consumption?

References

Thank you!

Auer, P. et al. (2002). “Finite-time analysis of the multiarmed bandit problem”. *Machine learning.*

Brégère, M. et al. (2019). “Target Tracking for Contextual Bandits: Application to Power Consumption Steering”.

Hastie, T. J. and R. J. Tibshirani (1990). *Generalized additive models.*

Lai, T. L. and H. Robbins (1985). “Asymptotically efficient adaptive allocation rules”. *Advances in applied mathematics.*

Li, L. et al. ([2010]). “A contextual-bandit approach to personalized news article recommendation”. Proceedings of the 19th International Conference on World Wide Web (WWW’10).

Pierrot, A. and Y. Goude (2011). “Short-term electricity load forecasting with generalized additive models”. *Proceedings of ISAP power.*