Optimization &
Statistical
Inference Lab

Paper Review

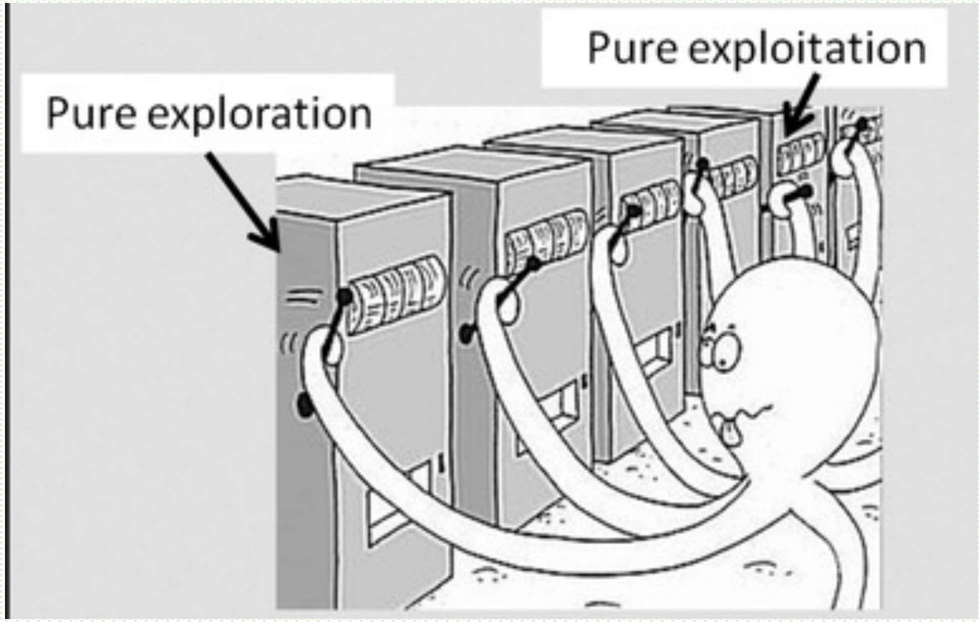
Contextual Gaussian Process Bandit Optimization

Andreas Krause and **Cheng Soon Ong**

Department of Computer Science, ETH Zurich

1. Problem Setting

What is the multi-armed bandit problem?



Exploration
an agent simultaneously attempts to acquire new knowledge

Exploitation
an agent optimizes its decision based on existing knowledge

Figure. Should I keep pulling the best lever so far or should I explore a new lever?

Source from http://www.primarydigit.com/blog/multi-arm-bandits-explorationexploitation-trade-off

1. Problem Setting

What is the multi-armed bandit problem?

How should we sample $x_1, x_2, \dots$ sequentially from the k populations in order to achieve the greatest possible expected value of the sum $S_n = x_1 + \cdots + x_n$ as $n \to \infty$?

Rule : The player wants to choose at each stage one of the k arms, the choice depending in some way on the record of previous trials.

Goal : to maximize the long-run total expected reward

1. Problem Setting

What is the Contextual bandit problem?

In most real-life applications, we have access to information that can be used to make a better decision when choosing among all actions in a MAB setting, this extra information is what gives Contextual Bandits their name

In stochastic contextual bandit, the reward $r_{i,t}$ can be represented as a function of the context $c_{i,t}$ and noise $\epsilon_{i,t}$

$$r_{i,t} = f(c_{i,t}) + \epsilon_{i,t}$$

2. Previous Algorithm

LinUCB

Algorithm 2 BaseLinUCB: Basic LinUCB with Linear Hypotheses at Step t

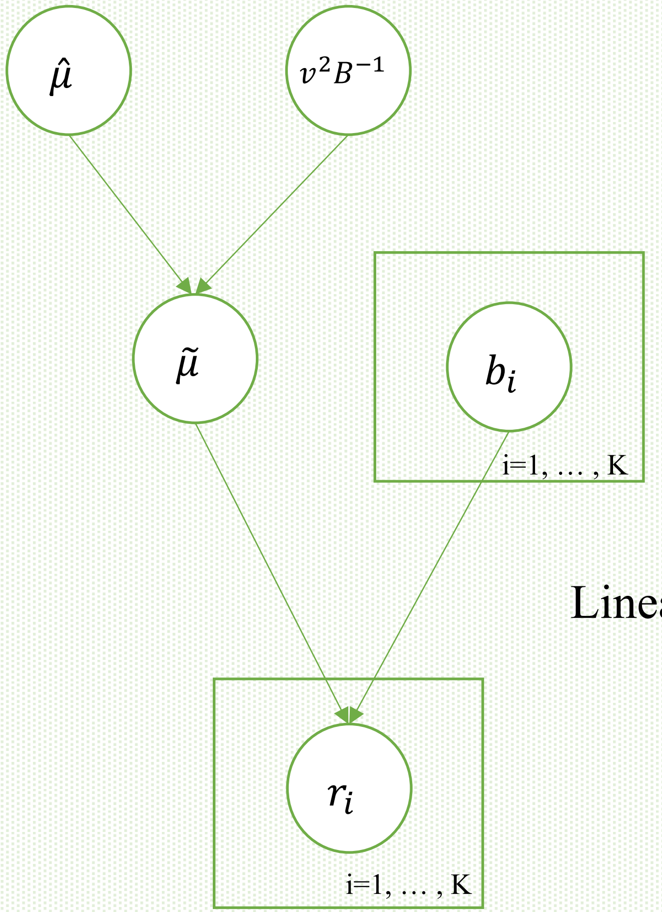
0: Inputs: $\alpha \in \mathbb{R}_+$, $\Psi_t \subseteq \{1, 2, \cdots, t-1\}$
1: $A_t \leftarrow I_d + \sum_{\tau \in \Psi_t} x_{\tau, a_\tau}^\top x_{\tau, a_\tau}$
2: $b_t \leftarrow \sum_{\tau \in \Psi_t} r_{\tau, a_\tau} x_{\tau, a_\tau}$
3: $\theta_t \leftarrow A_t^{-1} b_t$
4: Observe K arm features, $x_{t,1}, x_{t,2}, \cdots, x_{t,K} \in \mathbb{R}^d$
5: **for** $a \in [K]$ **do**
6: $\quad w_{t,a} \leftarrow \alpha \sqrt{x_{t,a}^\top A_t^{-1} x_{t,a}}$
7: $\quad \hat{r}_{t,a} \leftarrow \theta_t^\top x_{t,a}$
8: **end for**

2. Previous Algorithm

Thompson Sampling

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 $\mathcal{N}(\hat{\mu}, v^2B^{-1})$.
  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



Linear Payoff

All previous algorithms deal with the linear case. Then how about nonlinear case?

If we assume f is a member of exponential family, we can use GLM−UCB[1].

If we assume f is sampled from a Gaussian Process, we can use GP−UCB[2]/CGP−UCB[3].

If we assume f is an element of Reproducing Kernel Hilbert Space, we can use KernelUCB[4].

Also, we can use Thompson Sampling if we know the form of probability distribution.

GP-UCB is the algorithm of the context-free case.

1. Filippi et al. Parametric Bandits: The Generalized Linear Case NIPS 2010
2. Srinivas et al. Gaussian Process Optimization in the Bandit Setting: No Regret and Experimental Design ICML, 2010.
3. This Paper will deal with this part (NIPS 2011)
4. Valko el al. Finite-Time Analysis of Kernelized Contextual Bandits, UAI, 2013.

Stochastic Contextual Bandit

Before algorithm….

- $P(Y) = N(Y|0, K)$
  - $K_{nm} = k(x_n, x_m) = \frac{1}{\alpha}\phi(x_n)^T\phi(x_m)$
- $t_n = y_n + e_n$
  - t_n : Observed value with noise
  - y_n : Latent, error-free value
  - e_n : Error term distributed by following the Gaussian distribution
- $P(t_n|y_n) = N(t_n|y_n, \beta^{-1})$
  - β : Hyper-parameter of the error precision (or, variance considering the invert)
- $P(T|Y) = N(T|Y, \beta^{-1}I_N)$
  - $T = (t_1, \ldots, t_N)^T$, $Y = (y_1, \ldots, y_N)^T$
  - Assuming that the error terms are independent
- $P(T) = \int P(T|Y)P(Y)dY = \int N(T|Y, \beta^{-1}I_N)N(Y|0, K)dY$

These are from the lecture note of IE661-AI and DM2-Gaussian Process-ver-2 made by prof Moon

Before algorithm….

- $P(T) = \int P(T|Y)P(Y)dY = \int N(T|Y, \beta^{-1}I_N)N(Y|0, K)dY$
- $P(T|Y)P(Y) = P(T, Y) = P(Z)$
- $lnP(Z) = lnP(Y) + lnP(T|Y)$
$= -\frac{1}{2}(Y-0)^T K^{-1}(Y-0) - \frac{1}{2}(T-Y)^T \beta I_N (T-Y) + const. = -\frac{1}{2}Y^T K^{-1} Y - \frac{1}{2}(T-Y)^T \beta I_N (T-Y)$
- Second order term of $lnP(Z)$
  - $-\frac{1}{2}Y^T K^{-1} Y - \frac{\beta}{2}T^T T + \frac{\beta}{2}TY + \frac{\beta}{2}YT - \frac{\beta}{2}Y^T Y$
  $= -\frac{1}{2}\begin{pmatrix} Y \\ T \end{pmatrix}^T \begin{pmatrix} K^{-1} + \beta I_N & -\beta I_N \\ -\beta I_N & \beta I_N \end{pmatrix} \begin{pmatrix} Y \\ T \end{pmatrix} = -\frac{1}{2}Z^T R Z$
  - R becomes the precision matrix of Z
    - $M = (K^{-1} + \beta I_N - \beta I_N (\beta I_N)^{-1} \beta I_N)^{-1} = K$
  - $R^{-1} = \begin{pmatrix} K & K\beta I_N (\beta I_N)^{-1} \\ (\beta I_N)^{-1} \beta I_N K & (\beta I_N)^{-1} + (\beta I_N)^{-1} \beta I_N K \beta I_N (\beta I_N)^{-1} \end{pmatrix}$
  $= \begin{pmatrix} K & K \\ K & (\beta I_N)^{-1} + K \end{pmatrix}$
- First order term of $lnP(Z)$ → None
- $P(Z) = N(Z|0, R^{-1})$

These are from the lecture note of IE661-AI and DM2-Gaussian Process-ver-2 made by prof Moon

- $P(T) = \int P(T|Y)P(Y)dY = \int N(T|Y, \beta^{-1}I_N)N(Y|0,K)dY$
  - $P(T|Y)P(Y) = P(Y,T) = P(Z)$
  - $P(Y,T) = N(Y,T|(0 \quad 0), \begin{pmatrix} K & K \\ K & (\beta I_N)^{-1} + K \end{pmatrix})$
    - Precision Matrix $= \begin{pmatrix} K^{-1} + \beta I_N & -\beta I_N \\ -\beta I_N & \beta I_N \end{pmatrix}$
- Two theorems on multivariate normal distributions
  - Given $X = [X_1 \quad X_2]^T, \mu = [\mu_1 \quad \mu_2]^T, \Sigma = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}$
  - $P(X_1) = N(X_1|\mu_1, \Sigma_{11})$
  - $P(X_1|X_2) = N(X_1|\mu_1 + \Sigma_{12}{\Sigma_{22}}^{-1}(X_2 - \mu_2), \Sigma_{11} - \Sigma_{12}{\Sigma_{22}}^{-1}\Sigma_{21})$
- $P(T) = N(T|0, (\beta I_N)^{-1} + K)$
  - $K_{nm} = k(x_n, x_m) = \frac{1}{\alpha}\phi(x_n)^T\phi(x_m)$
  - One example → $k(x_n, x_m) = \theta_0 \exp\left(-\frac{\theta_1}{2}||x_n - x_m||^2\right) + \theta_2 + \theta_3 x_n^T x_m$
- Our ultimate question as a regression problem is
  - $P(t_{N+1}|T_N)$=? → $P(T_{N+1})$=!

These are from the lecture note of IE661-AI and DM2-Gaussian Process-ver-2 made by prof Moon

Optimization & Statistical Inference Lab

CGP-UCB

How it work?

- context $z_t \in Z$ from a set Z of contexts.
- action $s_t \in S$ from a set S of action
- payoff $y_t = f(s_t, z_t) + \epsilon_t$ where $f: S \times Z \rightarrow R$ (unknown)
- $\epsilon_t \sim N(0, \sigma^2)$: noise (independent across the rounds)

$r_t = sup_{s' \in S} f(s', z_t) - f(s_t, z_t)$ regret at each round
$R_T = \sum_{t=1}^{T} r_t$: cumulative regret

$X = S \times Z$: the set of all action-context pairs
$\mu: X \rightarrow R, \ \mu(x) = E[f(x)]$

Mean function

$k: X \times X \rightarrow R, k(x, x') = E[(f(x) - \mu(x))(f(x') - \mu(x'))]$
[WLOG] $\mu \equiv 0, k(x, x) \leq 1, \ for\ all\ x \in X$

Covariance function

$$\mu_T(\mathbf{x}) = \boldsymbol{k}_T(\boldsymbol{x})^T (\boldsymbol{K}_T + \sigma^2 \boldsymbol{I})^{-1} \boldsymbol{y}_T,$$
$$k_T(\mathbf{x}, \mathbf{x}') = k(\mathbf{x}, \mathbf{x}') - \boldsymbol{k}_T(\mathbf{x})^T (\boldsymbol{K}_T + \sigma^2 \boldsymbol{I})^{-1} \boldsymbol{k}_T(\mathbf{x}'),$$
$$\sigma_T^2(\mathbf{x}) = k_T(\mathbf{x}, \mathbf{x}),$$

where $\boldsymbol{k}_T(\mathbf{x}) = [k(\mathbf{x}_1, \mathbf{x}) \dots k(\mathbf{x}_T, \mathbf{x})]^T$ and $\boldsymbol{K}_T$ is the (positive semi-definite) kernel matrix $[k(\mathbf{x}, \mathbf{x}')]_{\mathbf{x}, \mathbf{x}' \in A_T}$. The choice of the kernel function turns out to be crucial in regularizing the function class to achieve sublinear regret (Section 4).

CGP-UCB

How to train?

- context $z_t \in Z$ from a set Z of contexts.
- action $s_t \in S$ from a set S of action
- payoff $y_t = f(s_t, z_t) + \epsilon_t$ where $f: S \times Z \rightarrow R$ (unknown)
- $\epsilon_t \sim N(0, \sigma^2)$: noise (independent across the rounds)

$r_t = sup_{s' \in S} f(s', z_t) - f(s_t, z_t)$ regret at each round
$R_T = \sum_{t=1}^{T} r_t$: cumulative regret

$X = S \times Z$: the set of all action-context pairs
$\mu: X \rightarrow R, \ \mu(x) = E[f(x)]$

Mean function

$k: X \times X \rightarrow R, k(x, x') = E[(f(x) - \mu(x))(f(x') - \mu(x'))]$
[WLOG] $\mu \equiv 0, k(x, x) \leq 1, \ for\ all\ x \in X$

Covariance function

Sigma is from error, and the identity matrix is from the assumption of the independence between error terms

$$\mu_T(\mathbf{x}) = \boldsymbol{k}_T(\boldsymbol{x})^T(\boldsymbol{K}_T + \sigma^2\boldsymbol{I})^{-1}\boldsymbol{y}_T,$$
$$k_T(\mathbf{x}, \mathbf{x}') = k(\mathbf{x}, \mathbf{x}') - \boldsymbol{k}_T(\mathbf{x})^T(\boldsymbol{K}_T + \sigma^2\boldsymbol{I})^{-1}\boldsymbol{k}_T(\mathbf{x}'),$$
$$\sigma_T^2(\mathbf{x}) = k_T(\mathbf{x}, \mathbf{x}),$$

where $\boldsymbol{k}_T(\mathbf{x}) = [k(\mathbf{x}_1, \mathbf{x}) \dots k(\mathbf{x}_T, \mathbf{x})]^T$ and $\boldsymbol{K}_T$ is the (positive semi-definite) kernel matrix $[k(\mathbf{x}, \mathbf{x}')]_{\mathbf{x}, \mathbf{x}' \in A_T}$. The choice of the kernel function turns out to be crucial in regularizing the function class to achieve sublinear regret (Section 4).

$$\mathbf{s}_t = \underset{\mathbf{s} \in S}{\operatorname{argmax}}\, \mu_{t-1}(\mathbf{s}, \mathbf{z}_t) + \beta_t^{1/2}\sigma_{t-1}(\mathbf{s}, \mathbf{z}_t),$$

Using this upper confidence bound

Next week

Before preview…

The regret R_T of the GP-UCB algorithm can be bounded as $O^*(\sqrt{T\gamma_T})$

$$\gamma_T := \max_{A \subset S : |A| = T} \mathrm{I}(\boldsymbol{y}_A; f),$$

$$\mathrm{I}(\boldsymbol{y}_A; f) = \mathrm{H}(\boldsymbol{y}_A) - \mathrm{H}(\boldsymbol{y}_A | f)$$

Shannon entropy

It quantifies the mutual information between the observed context-action pairs and the estimated payoff function f

Theorem 1 *Let $\delta \in (0,1)$. Suppose one of the following assumptions holds*

1. *X is finite, f is sampled from a known GP prior with known noise variance σ^2, and $\beta_t = 2\log(|X|t^2\pi^2/6\delta)$*
2. *$X \subseteq [0,r]^d$ is compact and convex, $d \in \mathbb{N}$, $r > 0$. Suppose f is sampled from a known GP prior with known noise variance σ^2, and that $k(\mathbf{x},\mathbf{x}')$ satisfies the following high probability bound on the derivatives of GP sample paths f: for some constants $a, b > 0$,*
$$\Pr\{\sup_{\boldsymbol{x}\in X} |\partial f/\partial x_j| > L\} \leq ae^{-(L/b)^2}, \quad j = 1,\dots,d.$$
Choose $\beta_t = 2\log(t^2 2\pi^2/(3\delta)) + 2d\log\left(t^2 dbr\sqrt{\log(4da/\delta)}\right)$.
3. *X is arbitrary; $||f||_k \leq B$. The noise variables ϵ_t form an arbitrary martingale difference sequence (meaning that $\mathbb{E}[\varepsilon_t \mid \varepsilon_1,\dots,\varepsilon_{t-1}] = 0$ for all $t \in \mathbb{N}$), uniformly bounded by σ. Further define $\beta_t = 2B^2 + 300\gamma_t \ln^3(t/\delta)$.*

Then the contextual regret of CGP-UCB is bounded by $\mathcal{O}^*(\sqrt{T\gamma_T\beta_T})$ w.h.p. Precisely,
$$\Pr\left\{R_T \leq \sqrt{C_1 T\beta_T\gamma_T} + 2 \quad \forall T \geq 1\right\} \geq 1 - \delta.$$
where $C_1 = 8/\log(1+\sigma^{-2})$.

(1) A known GP prior and finite X

(2) Infinite X with mild assumptions about k

(3) f has low "complexity" as quantified in terms of the Reproducing Kernel Hilbert Space norm associated with kernel k.

$$\gamma(T;k;V) = \max_{A\subseteq V, |A|\le T} \frac{1}{2}\log\left|\boldsymbol{I} + \sigma^{-2}[k(\mathbf{v},\mathbf{v}')]_{\mathbf{v},\mathbf{v}'\in A}\right|,$$

Theorem 2 *Let k_Z be a kernel function on Z with rank at most d (i.e., all Gram matrices over arbitrary finite sets of points $A \subseteq Z$ have rank at most d). Then*

$$\gamma(T; k_S \otimes k_Z; X) \le d\gamma(T; k_S; S) + d\log T.$$

The assumptions of Theorem 2 are satisfied, for example, if $|Z| < \infty$ and $\mathrm{rk}\,\mathbf{K}_Z = d$, or if k_Z is a d-dimensional linear kernel on $Z \subseteq \mathbb{R}^d$. Theorem 2 also holds with the roles of k_Z and k_S reversed.

Theorem 3 *Let k_S and k_Z be kernel functions on S and Z respectively. Then for the additive combination $k = k_S \oplus k_Z$ defined on X it holds that*

$$\gamma(T; k_S \oplus k_Z; X) \le \gamma(T; k_S; S) + \gamma(T; k_Z; Z) + 2\log T.$$

CGP-UCB

Main Contribution

1. Develop an efficient algorithm, CGP-UCB, for the contextual GP bandit problem;
2. Show that by flexibly combining kernels over contexts and actions, CGP-UCB can be applied to a variety of applications;
3. Provide a generic approach for deriving regret bounds for composite kernel functions;
4. Evaluate CGP-UCB on two case studies, related to automated vaccine design and sensor management.
5. The posterior inference can be performed in closed form.