Active Learning in Linear Stochastic Bandits

Marta Soare
INRIA Lille

Alessandro Lazaric
INRIA Lille

Rémi Munos
INRIA Lille / MSR - NE

1 Introduction

Consider the situation where a company has to make an accurate prediction of the result of a poll (e.g., an election). The company has relevant information on all the participants (age, profession, etc), but can only question a small number of them on their voting preferences. The obtained responses can then be used to infer a preference model for all the individuals in the group. To obtain an accurate estimation of the preference model, the company should use a strategy which sequentially selects the most **informative** members of the group.

More generally, the previous scenario can be framed as the problem of learning a noisy function uniformly well over a given input set. In this paper we focus on the specific case of linear functions, where the previous objective reduces to learning an accurate estimate of the weight vector that characterizes the function. We formalize this problem in the linear bandit setting, where each arm is a d - dimensional feature vector. In particular, we consider the setting where each pull to an arm x returns the linear combination between x and an unknown parameter vector θ perturbed by heteroscedastic noise. The objective is then to select the arms x within a set $\mathcal{X}$ which better allow to return a good estimate of θ after using a budget of n samples. Because of this objective, our work is closely related to optimal experimental design [6, 7] and to active learning.

The uniform estimation with limited budget problem has been recently studied for the multi-armed-bandit (MAB), where the input space is the set of the orthogonal arms of the standard stochastic bandit. With the objective of estimating uniformly well the mean values of several distributions, the authors in [2] and [4] estimate the variance per arm and exploit heteroscedasticity to allocate more samples to the parts of the input space where the variance is larger. The linear bandit setting that we consider here is an extension to the multi-armed bandit formalization and has the advantage of generalizing the MAB model by allowing the arms to be correlated and by taking into account more than two features at a time. Moreover, the linear bandit setting is a more realistic framework for the allocation problems discussed above.

2 Preliminaries

Let $\mathcal{X} \subseteq \mathbb{R}^d$ be a finite set of bounded arms such that $|\mathcal{X}| = k$ and for any $x \in \mathcal{X}$, $||x|| \leq L$. When an arm x is chosen, a noisy realization of an unknown function f is observed. In particular, we consider the linear case where a random realization r from arm x is defined as

$$r(x) = f(x) + \varepsilon(x) = x^\top \theta + x^\top \eta, \tag{1}$$

where η is a multivariate noise with zero mean and covariance matrix Σ, and θ is an unknown vector. Notice that unlike standard linear regression, in this case we consider an heteroscedastic noise $\varepsilon(x)$ which strictly depends on the arm x. Also, in contrast with the standard linear stochastic bandit problem [1, 3, 5], where the goal is to choose the arm in $\mathcal{X}$ that yields the maximal reward, here we focus on the problem of how to allocate a budget of n pulls on different arms in order to have an accurate estimate of θ.

Let $\mathcal{A}$ be an algorithm which at each round $t = 1, \dots, n$ chooses an arm x_t and observes the corresponding realization $r_t = x_t^\top \theta + \varepsilon_t = x_t^\top \theta + x_t^\top \eta_t$. For any sequence $\{x_t, r_t\}_{t=1}^n$ of arm-observation pairs we denote by $X_n \in \mathbb{R}^{n \times d}$ with $[X_n]_{t,i} = x_{t,i}$ the matrix of chosen arms, by $R_n \in \mathbb{R}^n$ with $[R_n]_t = r_t$ the vector of observations, and by $E_n \in \mathbb{R}^n$ with $[E_n]_t = \varepsilon_t$ the vector of noise. Once $\mathcal{A}$ used all the budget of n, we compute the least-squares estimate of θ:

$$\hat{\theta}_n^{ols} = A_n^{-1} b_n, \tag{2}$$

with $A_n = \sum_{t=1}^n x_t x_t^\top = X_n^\top X_n$ and $b_n = \sum_{t=1}^n r_t x_t = X_n^\top R_n$. For each arm $x \in \mathcal{X}$ we define the prediction error of $\hat{\theta}_n^{ols}$ as the expected quadratic loss $L_n(x) = \mathbb{E}[(x^\top \hat{\theta}_n^{ols} - x^\top \theta)^2]$ where the expectation refers to all possible sources of randomization in the observations r_t and in the choice of the sequence of arms x_t. Overall, we define the performance of the algorithm $\mathcal{A}$ by the loss corresponding to the worst estimated arm, i.e.

$$L_n(\mathcal{A}) = \max_{x \in \mathcal{X}} L_n(x). \tag{3}$$

The objective is, given a fixed budget n, to design an algorithm $\mathcal{A}$ that minimizes the loss $L_n(\mathcal{A})$.

3 The Optimal Static Allocation Algorithm

Consider a static allocation strategy $\mathcal{A}$ which selects arms $\{x_t\}$ independently from the observations. The covariance of $\hat{\theta}_n^{ols}$ can be computed as

$$\begin{aligned} \mathbb{V}[\hat{\theta}_n^{ols}|X_n] &= \mathbb{E}\Big[(\hat{\theta}_n^{ols} - \theta)(\hat{\theta}_n^{ols} - \theta)^\top | X_n\Big] \\ &= (X_n^\top X_n)^{-1} X_n^\top \mathbb{E}[E_n E_n^\top] X_n (X_n^\top X_n)^{-1} \\ &= (X_n^\top X_n)^{-1} X_n^\top \Omega_n X_n (X_n^\top X_n)^{-1}, \end{aligned} \tag{4}$$

where $\Omega_n = \text{diag}(\sigma^2(x_1), \ldots, \sigma^2(x_n))$ with $\sigma^2(x_t) = \mathbb{V}[\varepsilon_t] = x_t^\top \Sigma x_t$. Since $\mathcal{A}$ is static, then the previous expectations are conditioned on the fixed set of arms chosen over n rounds, which are summarized by the matrix X_n. Thus, the loss $L_n(x)$ can now be expressed directly as

$$L_n(x; \Sigma, X_n) = \mathbb{E}[(x^\top \hat{\theta}_n^{ols} - x^\top \theta)^2 | X_n] = x^\top \mathbb{V}[\hat{\theta}_n^{ols} | X_n] x,$$

where we make explicit the dependency of the loss on the sequence of arms X_n and the covariance matrix Σ in $L_n(x; \Sigma, X_n)$. As a result, an optimal static allocation should select the sequence of arms $X_n^{ols} \in \arg\min_{X_n} \max_{x \in \mathcal{X}} L_n(x; \Sigma, X_n)$. Although this allocation cannot be computed in closed form, an almost equivalent allocation can be obtained by pulling at each time t the arm

$$x_t^{ols} = \arg\max_{x \in \mathcal{X}} L_t(x; \Sigma, X_{t-1}^{ols}), \tag{5}$$

which corresponds to pulling the arm with the largest loss at each time step. Notice that this allocation does not require any actual observation r_t since it only relies on the set of arms $\mathcal{X}$ and on the covariance matrix Σ. It is interesting to analyze the behavior of optimal allocation in simple cases:

- If the arms form an orthogonal basis in $\mathbb{R}^d$, then the arms are all independent and the problem reduces to the active learning in multi-armed bandit setting studied in [2, 4]. As a result, the optimal strategy directly allocates the budget over arms proportionally to their variance (i.e., the number of times an arm x is pulled is proportional to $\sigma^2(x) / \sum_{x'} \sigma^2(x')$).
- If the noise is homoscedastic (i.e., $\Sigma = \sigma^2 I$), then the optimal allocation is no longer driven by the variance of the arms, but it still needs to compensate for a possibly uneven distribution of the arms in $\mathbb{R}^d$ by allocating less samples to the arms in regions of $\mathbb{R}^d$ which are dense of many arms.
- In the general case of heteroscedastic noise and an arbitrary set of arms $\mathcal{X}$, the optimal strategy implements an allocation that balances both the different variance of the arms and their uneven distribution in $\mathbb{R}^d$.

This qualitative description of the behavior of the optimal static allocation can be also illustrated by inspecting the definition of the loss L. Let $s \in \mathbb{R}^n$ be $s = x^\top (X_n^\top X_n)^{-1} X^\top$ such that for any t, $s_t = (x^\top A_n^{-1}) x_t$. Then, the loss can be written as follows:

$$L(x; \Sigma, X_n) = \sum_{t=1}^n s_t^2 \sigma^2(x_t) = \sum_{x' \in \mathcal{X}} T_n(x') \underbrace{(x^\top A_n^{-1} x')^2}_{(a)} \underbrace{((x')^\top \Sigma \, x')}_{(b)}, \tag{6}$$

where $T_n(x')$ denotes the number of times that arm x' was pulled up to time n. This form of the loss emphasizes the two elements that should be taken into account in designing an allocation strategy: the shape of the input space (term a) and the noise covariance matrix (term b).

4 Learning Algorithm

In a more realistic setting, the noise covariance matrix is unknown in advance, thus we need a learning strategy which is able to estimate Σ and at the same time to implement the optimal allocation suggested by equation 5. This requires to find a suitable trade-off between the *exploration* of the entire input space, with the objective of learning a good estimate of Σ (denoted $\widehat{\Sigma}$), and the *exploitation* of the current estimate to select arms according to the (estimated) optimal allocation. In order to define such a trade-off we rely on the construction of confidence bounds on the loss of each arm. The resulting algorithm is sketched in Figure 1.

```
Input: input space $\mathcal{X}$, budget $n$
while $t \leq n$ do
    Compute $B_t(x) = \max_{\tilde{\Sigma} \in \Gamma_t} L(x; \tilde{\Sigma}, X_{t-1})$
    Select $x_t = \arg\max_{x \in \mathcal{X}} B_t(x)$
    Pull $x_t$ twice and observe $r_t, r'_t$
    Compute $\hat{v}_t = C_t^{-1} d_t$ and $\hat{\Sigma}_t = \text{diag}(\hat{v}_t)$
    Compute the confidence set $\Gamma_t$ (eq. 9)
    t = t + 2
end while
Return $\hat{\theta}_n = A_n^{-1} b_n$
```

Figure 1: Learning algorithm

The most critical aspect of the algorithm is how to actually compute an estimate $\widehat{\Sigma}$ and how to build a confidence bound on it. In fact, although the idea of using upper confidence bounds has already been used in [4], unlike in the multi-arm bandit setting, the estimation of the variance of the noise is not trivial. In fact, we can only rely on noisy observations perturbed by a multivariate heteroscedastic noise which cannot be observed directly and which depend on the choice of the arm itself.

In order to simplify the derivation of an estimate of the covariance matrix and the construction of a confidence bound, we first introduce an assumption on the noise η.

Assumption 1. *Let the noise η be bounded in $[0, 1]^d$. Furthermore, let $v \in \mathbb{R}^d$ be the vector $v = [\sigma_1^2, \ldots, \sigma_d^2]$ such that the covariance matrix is the diagonal matrix $\Sigma = diag(\sigma_1^2, \ldots, \sigma_d^2)$.*

While the boundedness of the noise allows us to use standard concentration inequalities, a diagonal covariance matrix makes it possible to reduce the covariance estimation to a regularized regression problem. In fact, we notice that for any arm $x \in \mathcal{X}$, if we denote $y = x^2$, then the variance of the corresponding observations can be written as:

$$\sigma^2(x) = x^\top \Sigma x = \sum_{i=1}^{d} x_i^2 \sigma_i^2 = y^\top v \tag{7}$$

Equation (7) shows that the variance of the observations is a linear function with respect to the inputs y and the unknown variance vector v.

Although this simplifies the estimation of Σ, it is still required that at each time step t, when arm x_t is selected, two independent samples r_t and $r_{t'}$ need to be generated. This way we can construct the sample $z_t = \frac{1}{2}(r_t - r'_t)^2$, which is an unbiased sample of the variance of the observations corresponding to x_t, since $\mathbb{E}[z_t] = y_t^\top v = \sigma^2(x_t)$. Thus, we can set up the following regularized least squares problem[1]

$$\hat{v}_t = \arg\min_{v \in \mathbb{R}^d} \Big[\frac{2}{t} \sum_{t'=1}^{t} \big(y_{t'}^\top v - z_{t'} \big)^2 + \lambda ||v||_2 \Big] = C_t^{-1} d_t, \tag{8}$$

with $C_t = \sum_{t'=1}^{n} y_{t'} y_{t'}^\top + \lambda I = Y_t^\top Y_t + \lambda I$ and $d_t = \sum_{t'=1}^{n} z_{t'} y_{t'} = Y_t^\top Z_t$.

Note that the requirement to sample each arm twice does not necessarily correspond to a worsening of the performance. In fact, the same arm can be the one maximizing the loss several times before consuming the budget. Also, according to the performance measure of the algorithm, its efficiency can only be measured after the final sampling round, thus the order in which the arms are selected does not matter.

For the estimated variance vector $\hat{v}$ we can now rely on the self-normalized martingale techniques previously developed in the linear bandit setting [1]. Thus, we can derive the following lemma.

[1] Notice that because of the double sampling, the total amount of samples available after t steps is only $t/2$.

Lemma 1. *Let $\hat{v}_n$ be the regularized least-squares estimate of the variance vector v. Let $||v||_2^2 = \sum_{i=1}^d \sigma_i^4 \leq V^2$ and η satisfy Assumption 1. Then the confidence interval (recall that $||y_t||_2 \leq L^2$)*

$$\Gamma_t = \left\{ s \in \mathbb{R}^d, \|\hat{v}_t - s\|_{C_t} \leq \sqrt{d \, \log\left(\frac{1 + tL^4/\lambda}{\delta}\right)} + \lambda^{1/2} V \right\}, \tag{9}$$

is such that $v \in \Gamma_t$ with probability at least $1 - \delta$, for all $\delta > 0$, and $t \geq 0$.

The construction of the confidence set allows to choose at each time step the arm which maximizes the loss, for all possible values of the estimate of Σ. Now, it is crucial to be able to progressively tighten the confidence sets and to select at each step the worst-case arm, i.e., x_t which maximizes the loss for all possible estimate of Σ in the current confidence set Γ_t.

As showed in the pseudo code, the adaptive algorithm proceeds at every time step t according to the following steps. First, it estimates $\hat{v}_t$ and builds a confidence set around it. Then, choosing as $\widetilde{\Sigma}_t$ all possible vectors in the confidence set, it computes an upper bound on the possible loss, B_t. The choice of $\widetilde{\Sigma}$ and $x \in \mathcal{X}$ which generated B_t are then used at time step $t+1$ when the adaptive algorithm pulls $x_t = \arg\max_{x \in \mathcal{X}} B_t$.

5 Experiments and Conclusion

We illustrate the performance of the learning strategy and compare it with the optimal static strategy (when Σ is known in advance), and with the uniform strategy (*unif+*) which focuses only on the subset of d arms in $\mathcal{X}$ which are the closest to form an orthogonal basis, i.e., the subset of arms selected by the optimal allocation. We consider an input set consisting of five vectors in $\mathbb{R}^2$, as pictured in Fig. 2, and we define a noise η with a variance vector $v = [0.1, 0.4]$. In Fig. 3 we report the loss $L_n(\mathcal{A})$ multiplied by n. In fact, any static allocation strategy is expected to have a decreasing loss of the order of $O(1/n)$, thus in order to remove this trend and to emphasize the behavior of the different algorithms we plot $nL_n(\mathcal{A})$.

As illustrated in Fig. 3, the rescaled loss of the learning algorithm actually decreases from a performance similar to the uniform allocation down to the performance of the optimal allocation. This behavior suggests that as the budget grows, the loss of the learning algorithm tends to decrease as fast as for the optimal static allocation.

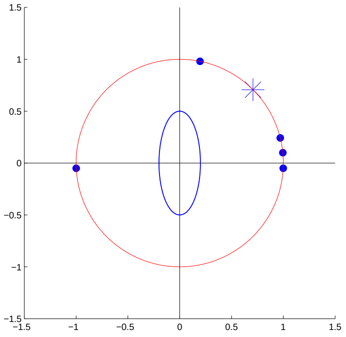

Figure 2: The input set $\mathcal{X}$, the covariance matrix (blue), and the θ vector (star).

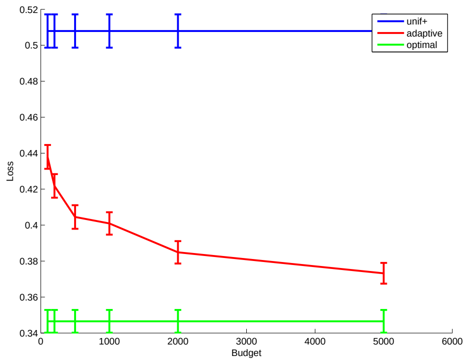

Figure 3: Rescaled loss $nL_n(\mathcal{A})$.

These preliminary numerical results show that the learning algorithm is effective in allocating the available budget n over arms to first estimate their variance and then to perform a nearly-optimal allocation. This preliminary work opens a number of interesting future challenges. In particular, we will focus on the general case of an infinite arm space $\mathcal{X}$, the case of an arbitrary covariance matrix Σ (and not diagonal), and we will derivate regret guarantees for the learning algorithm in the line of [4].

References

[1] Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, 2011.

[2] András Antos, Varun Grover, and Csaba Szepesvári. Active learning in heteroscedastic noise. *Theoretical Computer Science*, 411:2712–2728, 2010.

[3] Peter Auer. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3:397–422, 2002.

[4] Alexandra Carpentier, Alessandro Lazaric, Mohammad Ghavamzadeh, Rémi Munos, and Peter Auer. Upper-confidence-bound algorithms for active learning in multi-armed bandits. In *ALT - the 22nd conference on Algorithmic Learning Theory*, 2011.

[5] Varsha Dani, Thomas P. Hayes, and Sham M. Kakade. Stochastic Linear Optimization under Bandit Feedback. In *COLT 2008*, pages 355–366, 2008.

[6] V. V. Fedorov. *Theory of Optimal Experiments*. Academic Press, New York, 1972.

[7] J. Kiefer and J. Wolfowitz. The equivalence of two extremum problems. *Canadian Journal of Mathematics*, 12:363–366, 1960.