A dynamic programming strategy to balance exploration and exploitation in the bandit problem

Olivier Caelen, Gianluca Bontempi
Machine Learning Group, Département d'Informatique, Faculté des Sciences, Université Libre de Bruxelles, Brussels, Belgium,{ocaelen@ulb.ac.be, gbonte@ulb.ac.be}

Purpose - The K-armed bandit problem is a well-known formalization of the exploration versus exploitation dilemma. In this problem, a player is confronted to a gambling machine with K arms where each arm is associated to an unknown gain distribution. The goal of the player is to maximize the sum of the rewards. Several approaches have been proposed in literature to deal with the K-armed bandit problem. **Design/Methodology/Approach -** This paper introduces the concept of "*expected reward of greedy actions*" which is based on the notion of *probability of correct selection* (PCS), well-known in simulation literature. It is used to propose a semi-uniform algorithm which relies on the dynamic programming framework and on estimation techniques to optimally balance exploration and exploitation. **Originality/Value -** We introduce the notion of expected reward of a greedy action for which five estimators are proposed and compared. This notion is used in an original dynamic programming algorithm, called DP-greedy, which selects at each round which among a random or a gredy strategy is the one providing the highest expected gain. Experiments with a set of simulated and realistic bandit problems show that the DP-greedy algorithm is competitive with state-of-the-art semi-uniform techniques.

Key words: K-armed bandit problem; Dynamic programming; Applied probability; Assignment
History:

---

1. Introduction

In many real-world problems, decisions are made in order to maximize a gain function either explicitly or implicitly. This task is not trivial if the knowledge about the state of the environment is either partial or uncertain. In this context, it might be convenient to make decisions in the short-term to reduce the degree of uncertainty than to maximize the reward.

An example of such a situation is the design of clinical trials to assess and compare a set of new medical treatments (Hardwick and Stout, 1991). Here, the goal is to determine

the best treatment while minimizing the inconveniences for the patients. Another example is the business problem of selecting the best supplier on the basis of incomplete information (Azoulay-Schwartz et al., 2004).

The main issue in these two examples is how to combine an *exploitation* policy which targets immediate reward based on the acquired knowledge with an *exploration* policy which prefers to obtain additional insight by performing suboptimal actions.

The K-armed bandit problem, first introduced by Robbins (Robbins, 1952), is a classical instance of an exploration/exploitation problem (Meuleau and Bourgine, 1999) in which a casino player has to decide which arm of a slot machine to pull to maximize the total reward in a series of rounds. Each of the K arms of the slot machine returns a reward which is randomly distributed and unknown to the player. The player has to define a sequential selection policy on the basis of limited knowledge about the reward distributions which derives exclusively from previous results.

A taxonomy proposed in (Vermorel and Mohri, 2005) classified the many approaches that exist in the bandit problem literature into the following main families :

- the semi-uniform strategies (Watkins, 1989), which are characterized by the alternation of two working modes, namely the exploration mode and the exploitation mode;
- the interval-estimation strategies, like $UCB1$ (Auer et al., 2002) and $\beta - UCB$ (Audibert et al., 2006), which rely on upper bounds of the confidence intervals of the rewards;
- the probability-matching strategies, like $SoftMax$ (Vermorel and Mohri, 2005) and $Exp3$ (Auer et al., 1995), which choose arms according to a probability distribution that measures how likely the arms are close to the optimal.
- the index based strategies, like the Gittins index (Gittins, 1989), which is a solution based on the dynamic programming framework. This approach computes an index for each arm and selects the one with the highest index.

In this paper, we will focus on a new semi-uniform algorithm based on dynamic programming (Bertsekas, 1987; Powell, 2007). The simplest semi-uniform approach is the pure exploitation policy where the player keeps an updated estimate of the gains of the arms and at each round greedily chooses the arm which on average performed the best so far. Pure

exploitation uses current knowledge to select the seemingly best arm without reserving any time to explore what seems to be inferior arms. A possible alternative to a pure exploitation is the ϵ-greedy policy (Watkins, 1989) which preserves a fixed fraction of the rounds, quantified by a parameter ϵ, for performing random uniform selection.

A variant of the ϵ-greedy is the ϵ-decreasing-greedy policy (Vermorel and Mohri, 2005) where the exploration rate is set initially to a high value and then gradually decreases. For specific conditions on the initial parameters of ϵ-decreasing-greedy, (Auer et al., 2002) found an upper decreasing bound on the probability of selecting a sub-optimal arm. This means that after enough rounds, a player adopting the ϵ-decreasing-greedy policy has a high probability of selecting the best arm.

Most of existing semi-uniform algorithms rely on parametrized selection policies where the parameter (e.g. ϵ) is often set in an empirical manner. What we propose here is instead to adopt a data driven estimation approach to balance exploitation vs. exploration on the basis of historical data. The idea is to estimate from data the probability that a greedy selection returns the best arm and consequently to estimate the expectation of the reward once a greedy action is done. The probability of success of a greedy action is well-known in simulation literature as the probability of correct selection (PCS) (Kim and Nelson, 2006; Caelen and Bontempi, 2007). Here we extend existing work by (i) using the notion of PCS to define the expected greedy reward, (ii) studying in analytical and experimental terms the evolution of the expected greedy reward and (iii) proposing and comparing several estimation algorithms to compute this quantity from data. The expected greedy reward becomes then the quantity which should be taken into account, instead of the observed maximum average reward, to judge the utility of a greedy step.

The resulting algorithm is a new semi-uniform bandit algorithm called *Dynamic Programming greedy* (DP-greedy in short). The algorithm has these main characteristics : (i) it uses intensively estimation techniques to compute from data the *expected reward of a random selection* and the *expected reward of a greedy selection*, (ii) it compares the expected random reward and the expected greedy reward to "*price*" the exploration actions and the exploitation actions, respectively, (iii) it uses a dynamic programming framework (Bertsekas, 1987; Powell, 2007) to balance the exploration and the exploitation expected rewards. Since the DP-greedy algorithm adjusts the exploration-exploitation rate by comparing expected greedy and expected random reward, its strategy favors exploration initially and, as long as data are collected, switches gradually to a pure exploitation mode.

Like the Gittins solution (Gittins, 1989), our approach is based on the dynamic programming framework. However, it is worthy noting that in the Gittins formulation of the bandit problem, the control variables are represented by the single arms. Gittins returns a score index for each arms and the arm with the highest index is played. In our approach, the control is carried out not at the arm level but at the policy level. The control variable can take one of the two values: greedy action or random action. DP-greedy computes scores for each actions (greedy and random) and the action with the highest score is performed.

The outline of this paper is as follows. A formal definition of the bandit problem is given in Section 2. Section 3 introduces and motivates the semi-uniform DP-greedy algorithm in the case of perfect information. This section ends with a discussion on the relation between the Gittins solution and DP-greedy. The notion of *expected reward of greedy actions* is introduced and discussed in Section 4. A set of data driven estimators of the expected greedy reward is defined in Section 5. Section 6 describes the semi-uniform DP-greedy algorithm in the case of imperfect information. Section 7 is devoted to an experimental assessment of the DP-greedy algorithm for synthetic and real bandit benchmarks and we summarize our contributions in Section 8

2. The bandit problem

This section formally defines the K-armed bandit problem and introduces the notation used in the paper.

A K-armed bandit problem can be modeled by a set $\mathbf{z}_{\#} = \{\mathbf{z}_k\}$, $k = 1, \ldots, K$ of K random rewards[1] $\mathbf{z}_k$ with mean μ_k and standard deviation σ_k. Suppose that the goal of the player is to maximize the collected rewards. If we denote

$$k^* = \arg\max_k \mu_k$$

as the index of the optimal arm, we can associate to each arm k the regret

$$\Delta_k = \mu_{k^*} - \mu_k \geq 0$$

which is a measure of the loss related to the selection of the kth arm instead of the optimal one.

[1] We use boldface symbols to denote random variables.

In a game fixed to have H rounds, the player is expected to select one of the K arms at each round $l, l = 1, ..., H$. Let $\mathbf{N}(l) = [\mathbf{n}_1(l), \ldots, \mathbf{n}_K(l)]$ be a *counting vector* whose kth term denotes the number of times that the kth arm was selected during the first l rounds and $\mathbf{Z}_k(l) = \left[\mathbf{z}_k^1,\ \mathbf{z}_k^2,\ \mathbf{z}_k^3,\ \ldots,\ \mathbf{z}_k^{\mathbf{n}_k(l)}\right]$ be the vector of identically and independently distributed observed rewards of the arm k up to time l.

The bandit problem is an example of problem of *imperfect state information* (ISI) since only measured observations of the underlying stochastic process are available. A typical way of simplyfing such problems in dynamic programming relies on the assumption of certainty equivalence where the problem is decomposed in two independent subproblems: an estimation problem and a control problem applied to the *perfect state information* (PSI) version of the original problem. In the rest of this section we will present both the PSI and ISI formulation of the multi-armed bandit problem. The PSI control strategy will be outlined in Section 3 while the estimation strategy will be discussed in Section 5.

In the PSI configuration, we define (i) the *state* $s_l \in \mathcal{S}$ of the game at the lth round as the set $\langle \{\mu_k\}, \{\sigma_k\}, \{n_k(l)\} \rangle_{k=1,\ldots,K}$ and (ii) the *policy* of the player as a function $\pi : \mathcal{S} \rightarrow \{1, \ldots, K\}$ which returns for each state s the arm $\widetilde{\mathbf{k}}$ to be selected.

In the ISI configuration, we define (i) the *observed state* $\widehat{\mathbf{s}}(l) \in \widehat{\mathcal{S}}$ of the game at the lth round as the whole set of observations $\{\mathbf{Z}_k(l)\}$, $k = 1, \ldots, K$ and (ii) the *adopted policy* as a function $\widehat{\pi} : \widehat{\mathcal{S}} \rightarrow \{1, \ldots, K\}$ which returns for each state $\widehat{\mathbf{s}}$ the arm $\widehat{\mathbf{k}}$.

Two commonly adopted strategies in an ISI configuration are the pure random policy and the greedy policy. The pure random policy neglects any information about the state at time l and returns a selection

$$\widehat{\mathbf{k}}_r = \widehat{\pi}(\widehat{\mathbf{s}}(l)) \sim \mathrm{Uni}(1/K, \ldots, 1/K)$$

sampled according to an uniform distribution.

The greedy policy uses instead the information contained in $\widehat{\mathbf{s}}(l)$ and returns

$$\widehat{\mathbf{k}}_g = \arg\max \widehat{\boldsymbol{\mu}}_k(l)$$

where $\widehat{\boldsymbol{\mu}}_k(l)$ is the sample mean of the kth arm at the lth step.

Since $\mathbf{Z}_k(l)$ is a random vector, the state $\widehat{\mathbf{s}}$ and the output $\widehat{\mathbf{k}} = \widehat{\pi}(\widehat{\mathbf{s}})$ of the policy are random realizations too. At round l it is then possible to associate to a policy the *expected regret*

$$\delta(l) = \mu_{k^*} - \mathbb{E}\big[\mu_{\widehat{\mathbf{k}}}\big]$$

which quantifies the loss caused by the policy adoption. Note that $\mu_{\widehat{\mathbf{k}}}$ is a random variable which denotes the average reward of the arm $\widehat{\mathbf{k}}$ selected at the lth step.

Let us now calculate the quantity δ both in the case of a random and a greedy policy. The expected regret of a uniform random exploration policy at round l is

$$\begin{aligned}\delta_r(l) &= \mu_{k^*} - \frac{1}{K}\sum_{k=1}^{K}\mu_k \\ &= \mu_{k^*} - \mu_r\end{aligned}$$

where the term

$$\mu_r = \frac{1}{K}\sum_{k=1}^{K}\mu_k \tag{1}$$

will be referred to as the *expected gain of a random exploration action*.

The expected regret of a pure greedy exploitation policy at round l is

$$\begin{aligned}\delta_g(l) &= \mu_{k^*} - \sum_{k=1}^{K} P_k(l)\cdot\mu_k \\ &= \mu_{k^*} - \mu_g,\end{aligned}$$

where

$$P_{\bar{k}}(l) = \text{Prob}\left\{\arg\max \widehat{\boldsymbol{\mu}}_k(l) = \bar{k}\right\} \tag{2}$$

is the probability that the greedy algorithm selects the arm $\bar{k}$ at the lth step and

$$\mu_g = \sum_{k=1}^{K} P_k(l)\cdot\mu_k \tag{3}$$

will be referred to as the *expected gain of a greedy exploitation action*. Note that, unlike the term (1), the term (3) is not constant for different values of l.

A measure of the global performance of a policy for an horizon H is returned by the *expected cumulative regret*

$$\rho^H = H\cdot\mu_{k^*} - \sum_{k=1}^{K}\mathbb{E}[\mathbf{n}_k]\cdot\mu_k. \tag{4}$$

3. The semi-uniform DP-greedy algorithm

This section will introduce an optimal semi-uniform strategy, which is characterized by the alternation of a random exploration mode and a greedy exploitation mode. According to

the definition of the expected greedy gain in (3), the performance of a greedy policy at the step l depends on the probability of selection $P_k(l)$. In particular, the closer P_{k^*} (i.e. the probability of correct selection (PCS)) will be to one, the higher will be the gain deriving from a greedy exploitation policy.

At the beginning of the games, that is when the values of l is low, it is unlikely that $P_{k^*}(l)$ would have converged to one and it is, however, interesting to carry out some random exploration in order to accelerate the convergence of P_{k^*} to one. These qualitative considerations, common to all exploitation/exploration approaches (Sutton and Barto, 1998), can be put in a formal framework by describing how, at each state of the bandit problem, the gain would evolve in the case of either a random or a greedy step. For this purpose, it is interesting to formulate the bandit problem as a Markov decision process (Puterman, 1994) with finite horizon and study the semi-uniform optimal solution returned by a dynamic programming approach (Bertsekas, 1987; Powell, 2007).

Accordingly with the certainty equivalence approach (Bertsekas, 1987), we design our dunamic programming strategy assuming first a PSI configuration.

Dynamic programming concerns discrete time problems where at each time v, a decision must be taken. A policy is a function which, for each state $s \in S$, returns the action $u \in U$ where U is the set containing all the possible actions. At state $s_v = i$, the choice of an action u may induce a *transition probability* $p_{ij}(u)$ to the next state $s_{v+1} = j$ where $i \in S$ and $j \in S$. At time v, the transition from state i to state j caused by u generates a gain $\alpha^v g(i, u, j)$ where g is a given gain function and $0 < \alpha \leq 1$ is a discount factor so that the future gain is less important than the present gain. We consider finite horizon problems with V steps. In a V step dynamic problem, the expected gain of a policy $\tilde{\pi}$, starting from an initial state i, is

$$J_V^{\tilde{\pi}}(i) = \mathbb{E}\left[\alpha^V \cdot G(s_V) + \sum_{v=0}^{V-1} \alpha^v \cdot g(s_v, u, s_{v+1}) \middle| s_0 = i \right]$$

where $\alpha^V \cdot G(s_V)$ is the gain in the final state s_V. Let us denote

$$J_V^*(i) = \max_{\tilde{\pi}} J_V^{\tilde{\pi}}(i)$$

the maximum gain that can be obtained during V steps once the initial state is i. This optimal gain function J_V^* can be shown to satisfy the recursive *Bellman's equation*

$$J_V^*(i) = \max_{u \in U} \sum_{j \in S} p_{ij}(u) \cdot \left(g(i, u, j) + \alpha J_{V-1}^*(j) \right) \tag{5}$$

where

$$J_0^*(i) = G(i)$$

and the optimal decision satisfies

$$u^* = \arg\max_{u \in U} \sum_{j \in S} p_{ij}(u) \cdot \left(g(i,u,j) + \alpha J_{V-1}^*(j) \right). \tag{6}$$

In this paper, we propose a new semi-uniform algorithm called DP-greedy. The DP-greedy algorithm considers the bandit problem as a finite V-stages dynamic programming problem in which at each stage two possible actions are available in the set U : a random exploration action r and a greedy exploitation action g. For DP-greedy, policy $\tilde{\pi}$ is a function $\tilde{\pi} : \mathcal{S} \rightarrow \{r, g\}$ which returns for each state s an action (either random or greedy).

Let $s_v = \langle \{\mu_k\}, \{\sigma_k\}, \{n_k^v\} \rangle$, $k = 1, \ldots, K$, be the state of the bandit problem at time v. A transition from the state s_v resulting in choosing arm k implies that an element of the set $\{n_k^v\}$ is increased by one. Let us denote by $s_{v+1}^k = \langle \{\mu_k\}, \{\sigma_k\}, \{n_1^v, \ldots, n_k^v + 1, \ldots, n_K^v\} \rangle$ the successor state of s_v if arm k is tested. According to (2), P_k denotes the probability that a transition from s_v to s_{v+1}^k occurs when we perform a greedy action while $1/K$ denotes the transition probability when we perform a random action. Note also that, whatever the performed action is, a transition from s_v to s_{v+1}^k returns a gain with expected value equal to μ_k.

Once we have interpreted the bandit problem as a Markov decision problem, it is possible to define the associated recursive *Bellman's equation*. The V stage expected α discount gain (5) for an optimal semi-uniform policy, starting in state s_v, is

$$J_V^*(s_v) = \max \left[\, A_g^V(s_v) \; , \; A_r^V(s_v) \, \right] \tag{7}$$

where

$$\begin{aligned} A_g^V(s_v) &= \sum_{k=1}^{K} P_k^{s_v} \cdot \left(\mu_k + \alpha \cdot J_{V-1}^*\left(s_{v+1}^k\right) \right) \\ &= \mu_g(s_v) + \alpha \sum_{k=1}^{K} P_k^{s_v} \cdot J_{V-1}^*\left(s_{v+1}^k\right) \end{aligned} \tag{8}$$

and where

$$\begin{aligned} A_r^V(s_v) &= \sum_{k=1}^{K} \frac{1}{K} \cdot \left(\mu_k + \alpha \cdot J_{V-1}^*\left(s_{v+1}^k\right) \right) \\ &= \mu_r + \alpha \sum_{k=1}^{K} \frac{1}{K} \cdot J_{V-1}^*\left(s_{v+1}^k\right). \end{aligned} \tag{9}$$

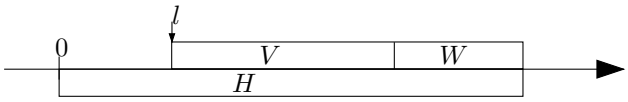

Figure 1: Notations. The horizon H is the total number of rounds. The step l is the number of rounds already played. V is the number of stages for the dynamic program. W is the number of pure exploitation greedy actions used to compute the value of the final gain $J_0^*(s_v)$.

Note that $\mu_g(s_v)$ stands for the expected gain of a greedy action in state s_v.

It follows from (7) that a semi-uniform optimal policy should return the action

$$u^* = \begin{cases} g & \text{if } A_g^V(s_v) - A_r^V(s_v) > 0 \\ r & \text{if } A_g^V(s_v) - A_r^V(s_v) < 0 \end{cases}$$

in a state s_v.

In spite of its optimality, a well known shortcoming of dynamic programming approach is the high computational cost in case of a large horizon H (Powell, 2007). In this case a possible remedy consists in (i) setting the number of stages V to a value lower than $H-l$ (e.g. $H-l-W$) and (ii) define the gain of the last stage $J_0^*(s_v)$ as the *expected discounted gain* μ_G^W of a series of W greedy actions where $W = H-l-V$ (see Figure 1). For a generic state s_w the quantity μ_G^W is recursively defined as follows

$$\begin{aligned} \mu_G^W(s_w) &= \sum_{k=1}^{K} P_k^{s_w} \cdot \left(\mu_k + \beta \cdot \mu_G^{W-1}\left(s_{w+1}^k\right)\right) \\ \mu_G^1(s_w) &= \mu_g(s_w), \end{aligned} \tag{10}$$

where $P_k^{s_w}$ is the probability of moving from the state s_w to the state $s_{w+1}^k = \langle \{\mu_k\}, \{\sigma_k\}, \{n_1^w, \ldots, n_k^w + 1, \ldots, n_K^w\} \rangle$ once a greedy action is applied.

4. The expected reward of a greedy action

The previous section showed that the quantity μ_g (defined in (3)) and the related term P_k (defined in (2)) must be known if we wish to implement an optimal semi-uniform bandit policy. Unfortunately these quantities are not accessible to the bandit player and an estimation procedure is required if we want to use the DP-greedy algorithm in practice. It is however interesting to remark that the definition of an optimal semi-uniform strategy relies on two quantities that are related to the performance of a pure greedy strategy. This means that if

we are able to understand the evolution of the bandit state in presence of a greedy strategy we can have useful insight about the optimal semi-uniform policy.

For this reason, before discussing in Section 5 some estimation procedures, we present here some interesting properties of the quantities μ_g and P_k.

First we will show that an analytical expression of the probability $P_{\bar{k}}$ of selecting the $\bar{k}$ arm in a greedy strategy can be derived in case of a bandit problem with normal distributed arms.

Theorem 1. *Let $\mathbf{z}_{\#} = \{\mathbf{z}_1, \ldots, \mathbf{z}_K\}$ be a set of $K > 1$ normal reward distributions $\mathbf{z}_k \sim \mathcal{N}[\mu_k, \sigma_k]$ with mean μ_k and standard deviation σ_k and suppose that n_k is given.*
If the selection policy is greedy ($\arg\max_{k\in[1\ldots K]}\{\widehat{\boldsymbol{\mu}}_k\}$) then the probability of selecting $\mathbf{z}_{\bar{k}}$ is

$$P_{\bar{k}} = Prob\{\ \widehat{\mathbf{r}}_1 > 0\ ,\ \ldots\ ,\ \widehat{\mathbf{r}}_{\bar{k}-1} > 0\ ,\ \widehat{\mathbf{r}}_{\bar{k}+1} > 0\ ,\ \ldots\ ,\ \widehat{\mathbf{r}}_K > 0\ \} \tag{11}$$

where $(\widehat{\mathbf{r}}_1, \ldots, \widehat{\mathbf{r}}_{\bar{k}-1}, \widehat{\mathbf{r}}_{\bar{k}+1}, \ldots, \widehat{\mathbf{r}}_K)^T$ follows a multivariate normal distribution

$$(\widehat{\mathbf{r}}_1, \ldots, \widehat{\mathbf{r}}_{\bar{k}-1}, \widehat{\mathbf{r}}_{\bar{k}+1}, \ldots, \widehat{\mathbf{r}}_K)^T \sim \mathcal{N}[\Gamma, \Sigma]$$

with mean

$$\Gamma = \begin{pmatrix} \mu_{\bar{k}} - \mu_1 \\ \vdots \\ \mu_{\bar{k}} - \mu_{\bar{k}-1} \\ \mu_{\bar{k}} - \mu_{\bar{k}+1} \\ \vdots \\ \mu_{\bar{k}} - \mu_K \end{pmatrix},$$

and covariance matrix Σ

$$\Sigma = \begin{pmatrix} \frac{\sigma_{\bar{k}}^2}{n_{\bar{k}}} + \frac{\sigma_1^2}{n_1} & \cdots & \frac{\sigma_{\bar{k}}^2}{n_{\bar{k}}} & \frac{\sigma_{\bar{k}}^2}{n_{\bar{k}}} & \cdots & \frac{\sigma_{\bar{k}}^2}{n_{\bar{k}}} \\ \vdots & \ddots & \vdots & \vdots & & \vdots \\ \frac{\sigma_{\bar{k}}^2}{n_{\bar{k}}} & \cdots & \frac{\sigma_{\bar{k}}^2}{n_{\bar{k}}} + \frac{\sigma_{\bar{k}-1}^2}{n_{\bar{k}-1}} & \frac{\sigma_{\bar{k}}^2}{n_{\bar{k}}} & \cdots & \frac{\sigma_{\bar{k}}^2}{n_{\bar{k}}} \\ \frac{\sigma_{\bar{k}}^2}{n_{\bar{k}}} & \cdots & \frac{\sigma_{\bar{k}}^2}{n_{\bar{k}}} & \frac{\sigma_{\bar{k}}^2}{n_{\bar{k}}} + \frac{\sigma_{\bar{k}+1}^2}{n_{\bar{k}+1}} & \cdots & \frac{\sigma_{\bar{k}}^2}{n_{\bar{k}}} \\ \vdots & & \vdots & \vdots & \ddots & \vdots \\ \frac{\sigma_{\bar{k}}^2}{n_{\bar{k}}} & \cdots & \frac{\sigma_{\bar{k}}^2}{n_{\bar{k}}} & \frac{\sigma_{\bar{k}}^2}{n_{\bar{k}}} & \cdots & \frac{\sigma_{\bar{k}}^2}{n_{\bar{k}}} + \frac{\sigma_K^2}{n_K} \end{pmatrix}$$

where n_k is the number of observations of $\mathbf{z}_k$.

Proof. According to (2), $P_{\bar{k}}$ is the probability that $\widehat{\boldsymbol{\mu}}_{\bar{k}}$ is the maximum in $\{\widehat{\boldsymbol{\mu}}_1, \ldots, \widehat{\boldsymbol{\mu}}_K\}$:

$$\begin{aligned} P_{\bar{k}} &= \text{Prob}\left\{\widehat{\mathbf{k}}_g = \bar{k}\right\} \\ &= \text{Prob}\left\{\bar{k} = \arg\max_{k\in[1\ldots K]}\{\widehat{\boldsymbol{\mu}}_k\}\right\} \\ &= \text{Prob}\left\{\ \widehat{\boldsymbol{\mu}}_{\bar{k}} > \widehat{\boldsymbol{\mu}}_1\ ,\ \ldots,\ \widehat{\boldsymbol{\mu}}_{\bar{k}} > \widehat{\boldsymbol{\mu}}_{\bar{k}-1}\ ,\ \widehat{\boldsymbol{\mu}}_{\bar{k}} > \widehat{\boldsymbol{\mu}}_{\bar{k}+1}\ ,\ \ldots,\ \widehat{\boldsymbol{\mu}}_{\bar{k}} > \widehat{\boldsymbol{\mu}}_K\ \right\} \\ &= \text{Prob}\left\{\ \widehat{\mathbf{r}}_1 > 0\ ,\ \ldots\ ,\ \widehat{\mathbf{r}}_{\bar{k}-1} > 0\ ,\ \widehat{\mathbf{r}}_{\bar{k}+1} > 0\ ,\ \ldots\ ,\ \widehat{\mathbf{r}}_K > 0\ \right\}, \end{aligned}$$

where $\widehat{\mathbf{r}}_k = \widehat{\boldsymbol{\mu}}_{\bar{k}} - \widehat{\boldsymbol{\mu}}_k$. It follows that $P_{\bar{k}}$ denotes also the probability that all the components of the vector $(\widehat{\mathbf{r}}_1, \ldots, \widehat{\mathbf{r}}_{\bar{k}-1}, \widehat{\mathbf{r}}_{\bar{k}+1}, \ldots, \widehat{\mathbf{r}}_K)^T$ are positive. Under the assumption of Gaussianity, this vector is a multivariate normal random variable

$$\begin{pmatrix} \widehat{\mathbf{r}}_1 \\ \vdots \\ \widehat{\mathbf{r}}_{\bar{k}-1} \\ \widehat{\mathbf{r}}_{\bar{k}+1} \\ \vdots \\ \widehat{\mathbf{r}}_K \end{pmatrix} = \begin{pmatrix} \widehat{\boldsymbol{\mu}}_{\bar{k}} - \widehat{\boldsymbol{\mu}}_1 \\ \vdots \\ \widehat{\boldsymbol{\mu}}_{\bar{k}} - \widehat{\boldsymbol{\mu}}_{\bar{k}-1} \\ \widehat{\boldsymbol{\mu}}_{\bar{k}} - \widehat{\boldsymbol{\mu}}_{\bar{k}+1} \\ \vdots \\ \widehat{\boldsymbol{\mu}}_{\bar{k}} - \widehat{\boldsymbol{\mu}}_K \end{pmatrix} \sim \mathcal{N}\left[\Gamma, \Sigma\right]$$

with mean vector

$$\Gamma = \begin{pmatrix} \mu_{\bar{k}} - \mu_1 \\ \vdots \\ \mu_{\bar{k}} - \mu_{\bar{k}-1} \\ \mu_{\bar{k}} - \mu_{\bar{k}+1} \\ \vdots \\ \mu_{\bar{k}} - \mu_K \end{pmatrix},$$

and covariance matrix

$$\Sigma = \begin{pmatrix} \sigma^2_{\widehat{\mathbf{r}}_1,\widehat{\mathbf{r}}_1} & \cdots & \sigma^2_{\widehat{\mathbf{r}}_1,\widehat{\mathbf{r}}_{\bar{k}-1}} & \sigma^2_{\widehat{\mathbf{r}}_1,\widehat{\mathbf{r}}_{\bar{k}+1}} & \cdots & \sigma^2_{\widehat{\mathbf{r}}_1,\widehat{\mathbf{r}}_K} \\ \vdots & \ddots & \vdots & \vdots & & \vdots \\ \sigma^2_{\widehat{\mathbf{r}}_{\bar{k}-1},\widehat{\mathbf{r}}_1} & \cdots & \sigma^2_{\widehat{\mathbf{r}}_{\bar{k}-1},\widehat{\mathbf{r}}_{\bar{k}-1}} & \sigma^2_{\widehat{\mathbf{r}}_{\bar{k}-1},\widehat{\mathbf{r}}_{\bar{k}+1}} & \cdots & \sigma^2_{\widehat{\mathbf{r}}_{\bar{k}-1},\widehat{\mathbf{r}}_K} \\ \sigma^2_{\widehat{\mathbf{r}}_{\bar{k}+1},\widehat{\mathbf{r}}_1} & \cdots & \sigma^2_{\widehat{\mathbf{r}}_{\bar{k}+1},\widehat{\mathbf{r}}_{\bar{k}-1}} & \sigma^2_{\widehat{\mathbf{r}}_{\bar{k}+1},\widehat{\mathbf{r}}_{\bar{k}+1}} & \cdots & \sigma^2_{\widehat{\mathbf{r}}_{\bar{k}+1},\widehat{\mathbf{r}}_K} \\ \vdots & & \vdots & \vdots & \ddots & \vdots \\ \sigma^2_{\widehat{\mathbf{r}}_K,\widehat{\mathbf{r}}_1} & \cdots & \sigma^2_{\widehat{\mathbf{r}}_K,\widehat{\mathbf{r}}_{\bar{k}-1}} & \sigma^2_{\widehat{\mathbf{r}}_K,\widehat{\mathbf{r}}_{\bar{k}+1}} & \cdots & \sigma^2_{\widehat{\mathbf{r}}_K,\widehat{\mathbf{r}}_K} \end{pmatrix}$$

Now, since $\widehat{\boldsymbol{\mu}}_i$ and $\widehat{\boldsymbol{\mu}}_j$ are independent for $i \neq j$

$$\sigma^2_{\widehat{\mathbf{r}}_j,\widehat{\mathbf{r}}_j} = \text{Var}\left(\widehat{\boldsymbol{\mu}}_{\bar{k}} - \widehat{\boldsymbol{\mu}}_j\right) = \frac{\sigma^2_{\bar{k}}}{n_{\bar{k}}} + \frac{\sigma^2_j}{n_j},$$

it follows that

$$\begin{aligned}\sigma^2_{\widehat{\mathbf{r}}_i,\widehat{\mathbf{r}}_j} &= \sigma^2_{\widehat{\mathbf{r}}_j,\widehat{\mathbf{r}}_i} = \operatorname{cov}\left[\widehat{\boldsymbol{\mu}}_{\bar{k}} - \widehat{\boldsymbol{\mu}}_i, \widehat{\boldsymbol{\mu}}_{\bar{k}} - \widehat{\boldsymbol{\mu}}_j\right] \\ &= \mathbb{E}\big[(\widehat{\boldsymbol{\mu}}_{\bar{k}} - \widehat{\boldsymbol{\mu}}_i - \mathbb{E}[\widehat{\boldsymbol{\mu}}_{\bar{k}} - \widehat{\boldsymbol{\mu}}_i]) \cdot (\widehat{\boldsymbol{\mu}}_{\bar{k}} - \widehat{\boldsymbol{\mu}}_j - \mathbb{E}\big[\widehat{\boldsymbol{\mu}}_{\bar{k}} - \widehat{\boldsymbol{\mu}}_j\big])\big] \\ &= \mathbb{E}\big[(\widehat{\boldsymbol{\mu}}_{\bar{k}})^2\big] - \mu^2_{\bar{k}} \\ &= \frac{\sigma^2_{\bar{k}}}{n_{\bar{k}}}.\end{aligned}$$

□

A major difficulty of the bandit problem is the dynamic and multivariate nature of the terms involved in the definition of an optimal policy. For this reason, it is important to study how the *expected reward of a greedy action* μ_g (3) changes with time.

We will first assume that $\mathbf{z}_\#$ contains only two arms where k^* is the index of the best arm and $\bar{k}^*$ is the other index. This is the simplest case and some analytical results can be derived about the time evolution of μ_g.

In particular, Theorem 2 shows that if $K = 2$ then testing $\mathbf{z}_1$ or $\mathbf{z}_2$ at round l will, in both cases, improve the value of μ_g at round $l+1$. The following Theorem 3 defines instead an optimal exploration policy which maximizes μ_g at round $l+1$.

Theorem 2. *Let $K = 2$ and $\mu_g\left(s^k_{l+1}\right)$ be the next expected reward of a greedy action when arm k is tested at round l. Then*

$$\forall k \in \{1,2\},\ \mu_g\left(s^k_{l+1}\right) > \mu_g.$$

Proof. $\forall k \in \{1,2\}, \mu_g\left(s^k_{l+1}\right) > \mu_g$ if and only if $\forall k \in \{1,2\}, P^{l+1}_{k^*}(k) > P_{k^*}$ where $P^{l+1}_{k^*}(k)$ is the probability of selecting the best alternative through a pure greedy algorithm at round $l+1$ if arm k is tested at round l.
By definition

$$P_{k^*} = \operatorname{Prob}\{\widehat{\boldsymbol{\mu}}_{k^*} - \widehat{\boldsymbol{\mu}}_{\bar{k}^*} > 0\} \tag{12}$$

and, under the assumption of Gaussianity,

$$(\widehat{\boldsymbol{\mu}}_{k^*} - \widehat{\boldsymbol{\mu}}_{\bar{k}^*}) \sim \mathcal{N}\left(\mu_{k^*} - \mu_{\bar{k}^*}, \frac{\sigma^2_{k^*}}{n_{k^*}} + \frac{\sigma^2_{\bar{k}^*}}{n_{\bar{k}^*}}\right).$$

Equation (12) can be rewritten as

$$\begin{aligned}P_{k^*} &= 1 - \operatorname{Prob}\{\widehat{\boldsymbol{\mu}}_{k^*} - \widehat{\boldsymbol{\mu}}_{\bar{k}^*} \leq 0\} \\ &= 1 - \frac{1}{2}\left[1 + \operatorname{erf}\left(\frac{\mu_{\bar{k}^*} - \mu_{k^*}}{\left(\frac{\sigma^2_{k^*}}{n_{k^*}} + \frac{\sigma^2_{\bar{k}^*}}{n_{\bar{k}^*}}\right)\sqrt{2}}\right)\right]\end{aligned}$$

where $\mathrm{erf}(\cdot)$ is the Gauss error function (Tong, 1990). Because $\mu_{\bar{k}^*} - \mu_{k^*}$ is negative and the derivative of the Gauss error function is allows positive, testing $\mathbf{z}_{k^*}$ or $\mathbf{z}_{\bar{k}^*}$ will both increase the probability of correct selection P_{k^*}.

□

Theorem 3. *Let us consider a bandit problem with two arms* $\mathbf{z}_{\#} = \{\mathbf{z}_{k_1}, \mathbf{z}_{k_2}\}$ *distributed according to a normal distribution. The policy*

$$\begin{cases} \textit{if } N_\Delta < 0 & \textit{then test } \mathbf{z}_{k_1} \\ \textit{if } N_\Delta > 0 & \textit{then test } \mathbf{z}_{k_2} \\ \textit{if } N_\Delta = 0 & \textit{then test either } \ \mathbf{z}_{k_1} \ \textit{or } \mathbf{z}_{k_2} \end{cases}$$

where

$$N_\Delta = n_{k_1} \cdot (n_{k_1} + 1) \cdot \left(\sigma_{k_2}^2 - \sigma_{k_1}^2\right) + \sigma_{k_1}^2 \cdot (n_{k_2} + n_{k_1} + 1) \cdot (n_{k_1} - n_{k_2})$$

maximizes μ_g *at the* $l+1$*th round.*

Proof. If $\left[\mu_g\left(s_{l+1}^{k_1}\right) - \mu_g\left(s_{l+1}^{k_2}\right)\right]$ is positive then $\mathbf{z}_{k_1}$ must be tested to maximize μ_g at round $l+1$ and if $\left[\mu_g\left(s_{l+1}^{k_1}\right) - \mu_g\left(s_{l+1}^{k_2}\right)\right]$ is negative then $\mathbf{z}_{k_2}$ must be tested.

$$\begin{aligned}
&\mu_g\left(s_{l+1}^{k_1}\right) - \mu_g\left(s_{l+1}^{k_2}\right) \\
&= P_{k^*}^{l+1}(k_1) \cdot \mu_{k^*} + P_{\bar{k}^*}^{l+1}(k_1) \cdot \mu_{\bar{k}^*} - P_{k^*}^{l+1}(k_2) \cdot \mu_{k^*} - P_{\bar{k}^*}^{l+1}(k_2) \cdot \mu_{\bar{k}^*} \\
&= \left[P_{k^*}^{l+1}(k_1) - P_{k^*}^{l+1}(k_2)\right] \cdot \mu_{k^*} + \left[P_{\bar{k}^*}^{l+1}(k_1) - P_{\bar{k}^*}^{l+1}(k_2)\right] \cdot \mu_{\bar{k}^*} \\
&= \left[P_{k^*}^{l+1}(k_1) - P_{k^*}^{l+1}(k_2)\right] \cdot \mu_{k^*} - \left[P_{k^*}^{l+1}(k_1) - P_{k^*}^{l+1}(k_2)\right] \cdot \mu_{\bar{k}^*} \\
&= \left[P_{k^*}^{l+1}(k_1) - P_{k^*}^{l+1}(k_2)\right] \cdot (\mu_{k^*} - \mu_{\bar{k}^*}) \\
&= \Delta P_{k^*} \cdot (\mu_{k^*} - \mu_{\bar{k}^*}).
\end{aligned}$$

Since $(\mu_{k^*} - \mu_{\bar{k}^*}) > 0$, the sign of $\left[\mu_g\left(s_{l+1}^{k_1}\right) - \mu_g\left(s_{l+1}^{k_2}\right)\right]$ is the same as the sign of ΔP_{k^*}.

Let $\mathrm{Var}^{k_1}\left(\widehat{\boldsymbol{\mu}}_{k^*}^{l+1} - \widehat{\boldsymbol{\mu}}_{\bar{k}^*}^{l+1}\right)$ and $\mathrm{Var}^{k_2}\left(\widehat{\boldsymbol{\mu}}_{k^*}^{l+1} - \widehat{\boldsymbol{\mu}}_{\bar{k}^*}^{l+1}\right)$ be respectively the variance of $\left(\widehat{\boldsymbol{\mu}}_{k^*}^{l+1} - \widehat{\boldsymbol{\mu}}_{\bar{k}^*}^{l+1}\right)$ at $l+1$ if either $\mathbf{z}_{k_1}$ or $\mathbf{z}_{k_2}$ is tested at l. $\Delta\mathrm{Var}$ is defined as the difference between the two variances at $l+1$. A reduction of the variance of $\left(\widehat{\boldsymbol{\mu}}_{k^*}^{l+1} - \widehat{\boldsymbol{\mu}}_{\bar{k}^*}^{l+1}\right)$ improves the probability of selecting the best random variable $\mathbf{z}_{k^*}$. The sign of $\Delta\mathrm{Var}$ is thus the opposite of the sign of ΔP_{k^*}.

$$\begin{aligned}
\Delta\mathrm{Var} &= \mathrm{Var}^{k_1}\left(\widehat{\boldsymbol{\mu}}_{k^*}^{l+1} - \widehat{\boldsymbol{\mu}}_{\bar{k}^*}^{l+1}\right) - \mathrm{Var}^{k_2}\left(\widehat{\boldsymbol{\mu}}_{k^*}^{l+1} - \widehat{\boldsymbol{\mu}}_{\bar{k}^*}^{l+1}\right) \\
&= \frac{\sigma_{k_1}^2}{n_{k_1}+1} + \frac{\sigma_{k_2}^2}{n_{k_2}} - \frac{\sigma_{k_1}^2}{n_{k_1}} - \frac{\sigma_{k_2}^2}{n_{k_2}+1} \\
&= \frac{N_\Delta}{D_\Delta}
\end{aligned}$$

where

$$
\begin{aligned}
N_\Delta &= n_{k_2} n_{k_1} \left(n_{k_2}+1\right) \sigma_{k_1}^2 + \left(n_{k_1}+1\right) n_{k_1} \left(n_{k_2}+1\right) \sigma_{k_2}^2 \\
&\quad - \left(n_{k_1}+1\right) n_{k_2} \left(n_{k_2}+1\right) \sigma_{k_1}^2 - \left(n_{k_1}+1\right) n_{k_2} n_{k_1} \sigma_{k_2}^2 \\
&= n_{k_1} \left(\sigma_{k_2}^2 - \sigma_{k_1}^2\right) \left(n_{k_1}+1\right) + \sigma_{k_1}^2 \left(n_{k_1} - n_{k_2}\right) \left(n_{k_1} + n_{k_2} + 1\right) \\
D_\Delta &= \left(n_{k_1}+1\right) n_{k_2} n_{k_1} \left(n_{k_2}+1\right)
\end{aligned}
$$

Since $D_\Delta > 0$, the following policy

$$
\begin{cases}
\text{if } N_\Delta < 0 & \text{then test } \mathbf{z}_{k_1} \\
\text{if } N_\Delta > 0 & \text{then test } \mathbf{z}_{k_2} \\
\text{if } N_\Delta = 0 & \text{then test either } \mathbf{z}_{k_1} \text{ or } \mathbf{z}_{k_2}
\end{cases}
$$

maximizes μ_g at round $l+1$. □

The theoretical results discussed above do not easily extend to the case where $K > 2$. In particular, in (Caelen and Bontempi, 2008) we performed an experimental study showing that the evolution of μ_g is very complex for $K > 2$ and that the monotone property $\mu_g\left(s_{l+1}^k\right) > \mu_g$ (Theorem 2) does not always hold when the quantity $\frac{\sigma}{\sqrt{n}}$ in the least rewarding arms is high.

This means that for $K > 2$ the working mode of a semi-uniform bandit policy affects the evolution of μ_g. At the beginning of the task or when the reward variances of the arms are high, random exploration is the best mode to improve the expected reward of a greedy action and we show that greedy actions, which test the best alternative, can reduce the value of μ_g at the next step such that the future PCS can decrease.

When enough knowledge is collected, such that the $\frac{\sigma}{\sqrt{n}}$ values are small, a good policy should converge to a more greedy exploitation mode. The optimal balance of a semi-uniform bandit policy between exploration and exploitation is therefore a non trivial function of the evolution of μ_g.

5. Estimation of the expected greedy reward

In previous sections we assumed that the means and the standard deviations of the arm distributions are known. This is of course an unrealistic assumption which can be relaxed by adopting estimation techniques. In this section five different methods to estimate μ_g on the basis of the observed rewards $\{\mathbf{Z}_k(l)\}$, $k = 1, \ldots, K$, are introduced and discussed. An experimental assessment of these five techniques will be later presented in Section 7.1.

5.1. The naive estimator

The simplest way to design an estimator of the quantity μ_g is to take the sample average of the arm

$$\widehat{\mathbf{k}}_g = \arg\max \widehat{\boldsymbol{\mu}}_k(l)$$

that is the one returned by a greedy action. Unfortunately the following theorem shows that such an estimator, henceafter denoted by $\widehat{\boldsymbol{\mu}}_g^{max}$, is a biased estimator of μ_g.

Theorem 4. *Let us consider a set of K random variables $\{\widehat{\boldsymbol{\mu}}_1, \dots, \widehat{\boldsymbol{\mu}}_K\}$ and the naive estimator $\widehat{\boldsymbol{\mu}}_g^{max} = \max\{\widehat{\boldsymbol{\mu}}_1, \dots, \widehat{\boldsymbol{\mu}}_K\}$. It can be shown that $\mathbb{E}[\widehat{\boldsymbol{\mu}}_g^{max}] > \mu_g$.*

Proof. Let $k^* = \arg\max\{\mu_k\}$ be the index of the random variable with the highest mean and $a \geq 0$ be a non-negative real number. First, we will show that $\mathbb{E}[\widehat{\boldsymbol{\mu}}_g^{max} - \widehat{\boldsymbol{\mu}}_{k^*}]$ is positive

$$\begin{aligned}
\mathbb{E}[\widehat{\boldsymbol{\mu}}_g^{max} - \widehat{\boldsymbol{\mu}}_{k^*}] &= \mathbb{E}[\boldsymbol{\Delta}] \\
&= \int_0^\infty x f_{\boldsymbol{\Delta}}(x)dx \\
&\geq \int_a^\infty x f_{\boldsymbol{\Delta}}(x)dx \geq a \int_a^\infty f_{\boldsymbol{\Delta}}(x)dx = \\
&= a \cdot \text{Prob}\{\boldsymbol{\Delta} \geq a\}.
\end{aligned}$$

Let a_i be a positive sequence of real numbers decreasing to zero ($a_i = 1/i$ for example)

$$\begin{aligned}
\sum_{i=1}^{\infty} \text{Prob}\{\boldsymbol{\Delta} \geq a_i\} &\geq \text{Prob}\left\{\bigcup_{i=1}^{\infty} \{\boldsymbol{\Delta} \geq a_i\}\right\} \\
&= \text{Prob}\{\boldsymbol{\Delta} > 0\}.
\end{aligned}$$

Since $\text{Prob}\{\boldsymbol{\Delta} > 0\} \neq 0$, we know that $\text{Prob}\{\boldsymbol{\Delta} \geq a_i\} \neq 0$ must be true for some a_i and thus $\mathbb{E}[\widehat{\boldsymbol{\mu}}_g^{max} - \widehat{\boldsymbol{\mu}}_{k^*}] > 0$

$$\begin{aligned}
\mathbb{E}[\widehat{\boldsymbol{\mu}}_g^{max}] - \mathbb{E}[\widehat{\boldsymbol{\mu}}_{k^*}] > 0 \qquad &\iff \\
\mathbb{E}[\widehat{\boldsymbol{\mu}}_g^{max}] > \mu_{k^*} \geq \sum_{k=1}^{K} P_k \cdot \mu_k &= \mu_g.
\end{aligned}$$

□

5.2. The plug-in approach

A possible alternative to the naive estimator discussed in the previous section is provided by the plug-in approach (Efron and Tibshirani, 1993). By replacing in the definition of μ_g the

terms $\{\mu_k\}$ and $\{P_k\}$ (Equation (11)) by their respective plug-in estimators $\{\widehat{\boldsymbol{\mu}}_k\}$ and $\{\widehat{\mathbf{P}}_k\}$ we obtain the estimator

$$\widehat{\boldsymbol{\mu}}_g^{pl} = \sum_{k=1}^{K} \widehat{\mathbf{P}}_k \cdot \widehat{\boldsymbol{\mu}}_k$$

In spite of its plug-in nature it can be shown that this estimator is biased too.

Theorem 5. *Consider a set of K random variables $\{\widehat{\boldsymbol{\mu}}_1, \ldots, \widehat{\boldsymbol{\mu}}_K\}$. If $\widehat{\boldsymbol{\mu}}_g^{pl} = \sum_{k=1}^{K} \widehat{\mathbf{P}}_k \cdot \widehat{\boldsymbol{\mu}}_k$ is the plug-in estimator of μ_g then $\mathbb{E}\left[\widehat{\boldsymbol{\mu}}_g^{pl}\right] > \mu_g$.*

Proof.

$$\begin{aligned}
\mathbb{E}\left[\widehat{\boldsymbol{\mu}}_g^{pl}\right] - \mu_g &= \mathbb{E}\left[\sum_{k=1}^{K} \widehat{\mathbf{P}}_k \cdot \widehat{\boldsymbol{\mu}}_k\right] - \sum_{k=1}^{K} P_k \cdot \mu_k \\
&= \sum_{k=1}^{K} \mathbb{E}\left[\widehat{\mathbf{P}}_k \cdot \widehat{\boldsymbol{\mu}}_k\right] - \sum_{k=1}^{K} P_k \cdot \mu_k \\
&= \sum_{k=1}^{K} \left(\mathbb{E}\left[\widehat{\mathbf{P}}_k \cdot \widehat{\boldsymbol{\mu}}_k\right] - P_k \cdot \mu_k\right) \\
&= \sum_{k=1}^{K} \operatorname{cov}\left(\widehat{\mathbf{P}}_k, \widehat{\boldsymbol{\mu}}_k\right)
\end{aligned}$$

where $\operatorname{cov}(\cdot,\cdot)$ is the covariance function and $\widehat{\mathbf{P}}_k$ is the estimated probability that k is the index of the highest average. Since this probability term is proportional to $\widehat{\boldsymbol{\mu}}_k$, the estimators $\widehat{\mathbf{P}}_k$ and $\widehat{\boldsymbol{\mu}}_k$ are positively correlated and consequently the bias term is different from zero. □

5.3. The holdout approach

As shown in the proof of Theorem 5 the bias of the plug-in estimator derives from the correlation of the two estimators $\widehat{\mathbf{P}}_k$ and $\widehat{\boldsymbol{\mu}}_k$. This section proposes an estimation technique which decorrelates the terms $\widehat{\mathbf{P}}_k$ and $\widehat{\boldsymbol{\mu}}_k$ by splitting the observed datasets into two overlapping portions. Let

$$\widehat{\boldsymbol{\mu}}_k^A = \frac{1}{\lfloor \mathbf{n}_k/2 \rfloor} \sum_{j=1}^{\lfloor \mathbf{n}_k/2 \rfloor} \mathbf{z}_k^j \tag{13}$$

$$\widehat{\boldsymbol{\sigma}}_k^A = \sqrt{\frac{1}{\lfloor \mathbf{n}_k/2 \rfloor - 1} \sum_{j=1}^{\lfloor \mathbf{n}_k/2 \rfloor} (\mathbf{z}_k^j - \widehat{\boldsymbol{\mu}}_k^A)^2} \tag{14}$$

be respectively the mean and the standard deviation estimations computed with the first half of the samples and let

$$\widehat{\boldsymbol{\mu}}_k^B = \frac{1}{\lceil \mathbf{n}_k/2 \rceil} \sum_{j=\lfloor \mathbf{n}_k/2 \rfloor + 1}^{\mathbf{n}_k} \mathbf{z}_k^j \tag{15}$$

$$\widehat{\boldsymbol{\sigma}}_k^B = \sqrt{\frac{1}{\lceil \mathbf{n}_k/2 \rceil - 1} \sum_{j=\lfloor \mathbf{n}_k/2 \rfloor + 1}^{\mathbf{n}_k} (\mathbf{z}_k^j - \widehat{\boldsymbol{\mu}}_k^B)^2} \tag{16}$$

be the related estimations made with the remaining half of the samples. ¿From the holdout estimates of mean and variance we can derive the holdout estimators $\widehat{\mathbf{P}}_k^A$ and $\widehat{\mathbf{P}}_k^B$. Note that, in spite of their holdout nature, both of these estimators use the entire set of $\mathbf{N}$ observations. The resulting holdout estimator of μ_g can be defined as follows

$$\widehat{\boldsymbol{\mu}}_g^{SPL1} = \sum_{k=1}^{K} \widehat{\mathbf{P}}_k^A \cdot \widehat{\boldsymbol{\mu}}_k^B.$$

Note that a more robust version of the holdout estimator can be designed by taking advantage of the averaging principle (Perrone and Cooper, 1993) according to which the average of two unbiased estimators with the same variance leads to a resulting estimator which is still unbiased but with half of the variance. The averaged estimator is

$$\widehat{\boldsymbol{\mu}}_g^{SPL2} = \frac{1}{2} \left(\sum_{k=1}^{K} \widehat{\mathbf{P}}_k^A \cdot \widehat{\boldsymbol{\mu}}_k^B + \sum_{k=1}^{K} \widehat{\mathbf{P}}_k^B \cdot \widehat{\boldsymbol{\mu}}_k^A \right),$$

where both the decorrelated pairs $(\widehat{\mathbf{P}}_k^A, \widehat{\boldsymbol{\mu}}_k^B)$ and $(\widehat{\mathbf{P}}_k^B, \widehat{\boldsymbol{\mu}}_k^A)$ are taken into consideration.

The experimental session will assess the gain derived from the averaging version of the holdout estimator.

5.4. The leave-one-out approach

This approach extends the holdout approach by adopting the well-known leave-one-out strategy (Bishop, 2006). Let us define by D the number of leave-one-out iterations, such that for $d = 1, \ldots, D$ and for each k, an index $\mathbf{o}_k^d$ is randomly chosen in the set $\{1, \ldots, \mathbf{n}_k\}$. Let

$$\mathbf{Z}_k(-\mathbf{o}_k^d) = \left\{ \mathbf{z}_k^1, \ldots, \mathbf{z}_k^{\mathbf{o}_k^d - 1}, \mathbf{z}_k^{\mathbf{o}_k^d + 1}, \ldots, \mathbf{z}_k^{\mathbf{n}_k} \right\}$$

be the set containing all the $\mathbf{n}_k$ realizations of the kth arm with $\mathbf{z}_k^{\mathbf{o}_k^d}$ set aside. The K samples $\left\{ \mathbf{z}_1^{\mathbf{o}_1^d}, \ldots, \mathbf{z}_K^{\mathbf{o}_K^d} \right\}$ are now used to greedily select the best arm $\mathbf{z}_b$ for the dth l-o-o iteration

where $b = \arg\max_k \left\{ \mathbf{z}_k^{\mathbf{o}_k^d} \right\}$. Let us denote by $\widehat{\boldsymbol{\mu}}_b^{-\mathbf{o}_b^d}$ the leave-one-out estimation of the mean of $\mathbf{z}_b$, that is the sample average of $\mathbf{Z}_b(-\mathbf{o}_b^d)$. Note that the estimate $\widehat{\boldsymbol{\mu}}_b^{-\mathbf{o}_b^d}$ is now decorrelated with respect to the selected index b.

After D leave-one-out iterations, we obtain a set $\left\{ \widehat{\boldsymbol{\mu}}_b^{-\mathbf{o}_b^1}, \ldots, \widehat{\boldsymbol{\mu}}_b^{-\mathbf{o}_b^d}, \ldots, \widehat{\boldsymbol{\mu}}_b^{-\mathbf{o}_b^D} \right\}$ of D estimations of the quantity μ_g each obtained on a different training set. The leave-one-out estimator of μ_g is then

$$\widehat{\boldsymbol{\mu}}_g^{loo} = \frac{1}{D} \sum_{d=1}^{D} \widehat{\boldsymbol{\mu}}_b^{-\mathbf{o}_b^d}.$$

6. The imperfect information DP-greedy algorithm

In Section 3 we described the perfect information version of the DP-greedy algorithm where the policy definition takes advantage of the full knowledge of the state in order to compute A_g and A_r. Here, we introduce the imperfect information version of the algorithm where, at each round, DP-greedy is assumed to have only a partial and noisy information about the state.

An initialization phase, where each arm is tested I times, gives us an initial prior distribution of the states. Suppose that we test the kth arm at round l and that we receive a reward $\mathbf{z}_k^{\mathbf{n}_k(l)+1}$. Let us add the reward to the vector $\mathbf{Z}_k(l) = [\mathbf{z}_k^1,\ \mathbf{z}_k^2,\ \ldots,\ \mathbf{z}_k^{\mathbf{n}_k(l)}]$ and let us define $\widehat{\mathbf{s}}_l = \{\mathbf{Z}_k(l)\}_{k=1,\ldots,K}$ as the state of the dynamic programing problem with imperfect state information.

In order to avoid the high computational complexity of DP-greedy discussed in Section 3, we used a version of the policy where $V = 1$ and $\alpha = 1$, that is

$$A_g\left(s_v\right) = \sum_{k=1}^{K} P_k^{s_v} \cdot \left(\mu_k + \mu_G^W\left(s_{v+1}^k\right)\right) \tag{17}$$

and

$$A_r\left(s_v\right) = \frac{1}{K} \sum_{k=1}^{K} \left(\mu_k + \mu_G^W\left(s_{v+1}^k\right)\right). \tag{18}$$

One of the techniques discussed in Section 5, for instance the holdout technique, can now be used to estimate the term A_g in (17) and the term A_r in (18) from observed data. First, the samples in $\{\mathbf{Z}_k(l)\}_{k=1,\ldots,K}$ are partitioned in two portions to compute $\widehat{\boldsymbol{\mu}}_k^A$, $\widehat{\boldsymbol{\sigma}}_k^A$, $\widehat{\boldsymbol{\mu}}_k^B$ and $\widehat{\boldsymbol{\sigma}}_k^B$ (see equations 13, 14, 15 and 16). Then the two plug-in versions (i.e. $\widehat{\mathbf{P}}_k^A$ and $\widehat{\mathbf{P}}_k^B$) of the probability P are derived. We also compute $\widehat{\boldsymbol{\mu}}_g^{SPL2}$, $\left(\widehat{\boldsymbol{\mu}}_g^{SPL2}\right)^A$ and $\left(\widehat{\boldsymbol{\mu}}_g^{SPL2}\right)^B$ which are

estimators of $\mu_g\left(s_{v+1}^k\right)$ built with the entire set, the first portion and the second portion, respectively. The associated estimator of A_g is then

$$\widehat{\mathbf{A}}_g = \frac{\sum_{k=1}^{K} \widehat{\mathbf{P}}_k^A \cdot \left(\widehat{\boldsymbol{\mu}}_k^B + \sum_{i=0}^{W-1} \beta^i \cdot \left(\widehat{\boldsymbol{\mu}}_g^{SPL2}\right)^B\right) + \widehat{\mathbf{P}}_k^B \cdot \left(\widehat{\boldsymbol{\mu}}_k^A + \sum_{i=0}^{W-1} \beta^i \cdot \left(\widehat{\boldsymbol{\mu}}_g^{SPL2}\right)^A\right)}{2}$$

where $\sum_{i=0}^{W-1} \beta^i \cdot \mu_g\left(s_{v+1}^k\right)$ is an approximation of $\mu_G^W\left(s_{v+1}^k\right)$ and the estimator of A_r is returned by

$$\widehat{\mathbf{A}}_r = \frac{1}{K} \sum_{k=1}^{K} \left(\widehat{\boldsymbol{\mu}}_k + \sum_{i=0}^{W-1} \beta^i \cdot \widehat{\boldsymbol{\mu}}_g^{SPL2}\right).$$

The DP-greedy strategy in the imperfect information state consists then in returning the action

$$\begin{cases} \mathbf{u}^* = g & \text{if } \widehat{\mathbf{A}}_g - \widehat{\mathbf{A}}_r > 0 \\ \mathbf{u}^* = r & \text{if } \widehat{\mathbf{A}}_g - \widehat{\mathbf{A}}_r < 0 \end{cases}.$$

A pseudo-code of the DP-greedy algorithm follows:

Algorithm 1 The DP-greedy algorithm

```
1: play each arm at least I times
2: loop
3:    Compute the values of Â_g and Â_r.
4:    if Â_g − Â_r > 0 then
5:       k̂ ← arg max_k{μ̂_k}
6:    else
7:       k̂ ← random
8:    end if
9:    play z_k̂
10: end loop
```

Note that during initialization each arm is tested at least I times (line 1). Then, for each following step, DP-greedy computes $\widehat{\mathbf{A}}_g$ and $\widehat{\mathbf{A}}_r$ (line 3) and accordingly decides the action. If $\widehat{\mathbf{A}}_g - \widehat{\mathbf{A}}_r > 0$, it exploits the current best arm (line 5), otherwise it explores (line 7).

7. Experiments and discussions

This section is divided in two parts. The first one assesses the performance of the five estimators described in Section 5 and the second one assesses the performance of the imperfect information DP-greedy described in Section 6.

	task 1			*task 2*			*task 3*			*task 4*			*task 5*		
	μ	σ	n	μ	σ	n	μ	σ	n	μ	σ	n	μ	σ	n
$\mathbf{z}_1$	0	0.5	5	0	1	5	0	2	5	0	4	5	0	6	5
$\mathbf{z}_2$	0	0.5	10	0	1	10	0	2	5	0	4	5	0	6	5
$\mathbf{z}_3$	0	0.5	5	0	1	5	0.5	1	10	0.5	4	5	0	6	5
$\mathbf{z}_4$	0.5	0.5	15	0.5	1	15	0.5	1	4	0.5	2	4	0.5	5	5
$\mathbf{z}_5$	1	0.5	10	1	1	10	0.5	1	4	0.5	3	4	0.5	5	5
$\mathbf{z}_6$							1	1	5	1	2	5	0.5	5	5
$\mathbf{z}_7$													1	2	5
μ_g	0.9967			0.8936			0.639625			0.4603			0.327		

	task 6			*task 7*		
	μ	σ	n	μ	σ	n
$\mathbf{z}_1$	0	8	4	0	8	4
$\mathbf{z}_2$	0	8	4	0	8	4
$\mathbf{z}_3$	0	8	4	0	8	4
$\mathbf{z}_4$	0.5	1	4	0.5	1	4
$\mathbf{z}_5$	0.5	1	4	0.5	1	4
$\mathbf{z}_6$	0.5	1	4	0.5	1	4
$\mathbf{z}_7$	1	1	4	1	1	400
μ_g	0.189			0.176		

Table 1: Seven synthetic tasks to compare the performance of some μ_g estimators. For each task, the first, the second and the last column respectively contains the mean, the standard deviation and the test number of each arm. The last line contains the value of μ_g. The tasks are sorted in a μ_g-complexity order.

7.1. Estimation experiments

This section compares experimentally the five estimators of μ_g by means of a set of 7 synthetic tasks (see table 1). The tasks can be sorted in a μ_g-complexity increasing order such that the first tasks can be considered as "*easy*" and the last one as "*very difficult*". Note that the only difference between task 6 and task 7 is the number of tests made on the best alternative which is hundred time higher in task 7.

To assess the performance of the estimators, each of the seven estimation tasks is randomly sampled $M = 10000$ times. For $m = 1, \ldots, M$, a data frame $\left(\{\mathbf{Z}_k\}_{k\in\{1,\ldots,K\}}\right)_m$ is created and used to compute the five estimates $(\widehat{\boldsymbol{\mu}}_g)_m$. Three error measures are considered:

1. the bias

$$\mathbf{bias} = Average\left(\widehat{\boldsymbol{\mu}}_g\right) - \mu_g,$$

where

$$Average\left(\widehat{\boldsymbol{\mu}}_g\right) = \frac{1}{M}\sum_{m=1}^{M}(\widehat{\boldsymbol{\mu}}_g)_m$$

2. the variance

$$\mathbf{var} = \frac{1}{M}\sum_{m=1}^{M}\left(Average\left(\widehat{\boldsymbol{\mu}}_g\right) - (\widehat{\boldsymbol{\mu}}_g)_m\right)^2$$

	task 1			*task 2*			*task 3*		
	bias	**var**	**mse**	**bias**	**var**	**mse**	**bias**	**var**	**mse**
$\widehat{\boldsymbol{\mu}}_g^{max}$	0.004	0.0244	0.0244	0.138	0.081	0.100	0.6253	0.145	0.536
$\widehat{\boldsymbol{\mu}}_g^{PL}$	−0.009	0.0277	0.0278	0.0452	0.096	0.098	0.4055	0.156	0.321
$\widehat{\boldsymbol{\mu}}_g^{SPL1}$	−0.073	0.0615	0.0669	−0.223	0.223	0.273	−0.165	0.497	0.524
$\widehat{\boldsymbol{\mu}}_g^{SPL2}$	−0.065	0.0432	0.0476	−0.204	0.167	0.208	−0.147	0.328	0.350
$\widehat{\boldsymbol{\mu}}_g^{loo}$	−0.216	0.0341	0.0811	−0.335	0.070	0.182	−0.1997	0.163	0.203

	task 4			*task 5*			*task 6*		
	bias	**var**	**mse**	**bias**	**var**	**mse**	**bias**	**var**	**mse**
$\widehat{\boldsymbol{\mu}}_g^{max}$	1.858	1.011	4.466	3.094	2.182	11.754	3.615	6.252	19.321
$\widehat{\boldsymbol{\mu}}_g^{PL}$	1.314	1.044	2.773	2.177	2.206	6.946	2.636	6.425	13.377
$\widehat{\boldsymbol{\mu}}_g^{SPL1}$	−0.060	2.607	2.610	−0.054	5.457	5.460	−0.0004	16.711	16.71
$\widehat{\boldsymbol{\mu}}_g^{SPL2}$	−0.061	1.942	1.945	−0.063	4.212	4.215	−0.017	11.412	11.411
$\widehat{\boldsymbol{\mu}}_g^{loo}$	−0.073	1.009	1.015	−0.068	2.0240	2.028	−0.015	7.057	7.057

	task 7		
	bias	**var**	**mse**
$\widehat{\boldsymbol{\mu}}_g^{max}$	3.542	6.154	18.706
$\widehat{\boldsymbol{\mu}}_g^{PL}$	2.563	6.338	12.908
$\widehat{\boldsymbol{\mu}}_g^{SPL1}$	−0.073	16.740	16.744
$\widehat{\boldsymbol{\mu}}_g^{SPL2}$	−0.068	11.106	11.110
$\widehat{\boldsymbol{\mu}}_g^{loo}$	−0.061	6.908	6.911

Table 2: Results for the seven synthetic tasks. For each task and for each estimator, the **bias**, the **var** and the **mse** are given.

3. the mean square error

$$\mathbf{mse} = \frac{1}{M}\sum_{m=1}^{M}\left(\mu_g - \left(\widehat{\boldsymbol{\mu}}_g\right)_m\right)^2.$$

All the results are reported in Table 2. An analysis of the results in terms of *bias variance trade-off* follows. For easy problems (i.e. task 1 and 2), $\widehat{\boldsymbol{\mu}}_g^{max}$ and $\widehat{\boldsymbol{\mu}}_g^{PL}$ are both good estimators with low **mse**. This is due to a combination of a relatively small variance and a small bias. On more difficult tasks, the **bias** quickly increases and deteriorates the value of the **mse**. On the other hand, because of their higher variance and small bias, $\widehat{\boldsymbol{\mu}}_g^{SPL1}$, $\widehat{\boldsymbol{\mu}}_g^{SPL2}$ and $\widehat{\boldsymbol{\mu}}_g^{loo}$ are poor estimators for easy problems but powerful on complex tasks. Note that thanks to the averaging effect the variance of $\widehat{\boldsymbol{\mu}}_g^{SPL2}$ is always lower than the variance of $\widehat{\boldsymbol{\mu}}_g^{SPL1}$. Also, made exception for the trivial tasks (1 and 2), the estimator of μ_g which appears to be the most robust is the leave-one-out estimator $\widehat{\boldsymbol{\mu}}_g^{loo}$.

	B-1	B-2	B-3	B-4	B-5	B-6	B-7	B-8	B-9	B-10	B-11	B-12
K	3	5	10	3	5	10	3	5	10	3	5	10
σ_k	0.1	0.1	0.1	1	1	1	2	2	2	3	3	3

Table 3: The twelve synthetic benchmarks differ in the number of arms K and in the standard deviations of the rewards σ_k.

7.2. Bandit experiments

This section presents the experiments we used to benchmark the performance of our original DP-greedy algorithm against some semi-uniform state-of-the-art approaches. Nineteen semi-uniform bandit methods are tested : Ten ϵ-greedy instances with $\epsilon = \{0.00, 0.05, 0.10, \ldots, 0.45\}$, eight ϵ-decreasing-greedy (ϵ-Dgreedy in short) instances with $\epsilon_0 = \{1, 20, 40, 60, 80, 120, 160, 200\}$ and our DP-greedy method where the term β is set to 0.98.

We consider fifteen benchmark problems denoted B-1, B-2,...,B-15, respectively. The first twelve tasks are based on synthetically generated datasets, the remaining ones on real networking datasets. The cumulative regret (4) is used to measure and compare the performance of the policies. Each experiment is initialized by collecting $I = 6$ reward values per arm.

For the synthetic benchmarks we set the horizon $H = 4000$. Each synthetic benchmark is made of 100 randomly generated K-armed bandit tasks obtained by uniformly sampling the value means in the interval $[0, 1]$. The benchmarks differ in terms of number of arms and standard deviation of the rewards (see table 3). For each task, the rewards are normally distributed ($\mathbf{z}_k \sim N[\mu_k, \sigma_k]$). The performance of a bandit algorithm on these tasks is obtained by averaging over the whole 100 tasks.

The three real benchmarks are based on the real data benchmark proposed in (Vermorel and Mohri, 2005). In this task, an agent wants to recover data through different network sources. At each step, the agent selects one source and waits until data is received. The goal of the agent is to minimize the sum of the waiting times for the successive tests. In order to simulate the delay, a dataset was built by accessing the home pages of 700 universities (every 10 minutes for about 10 days) and storing the time delay (in milliseconds)[2]. If we interpret this task as a bandit problem, each university home page plays the role of an arm and each delay the role of a (negative) reward. In our experiments, in order to generate a sufficient number of problem instances, we randomly selected 100 times $K = 3$, $K = 5$ or $K = 10$ universities and computed the performance of the methods over an horizon of $H = 4000$

[2]The dataset can be downloaded from http://sourceforge.net/projects/bandit

Strategies	B-1	B-2	B-3	B-4	B-5	B-6	B-7	B-8	B-9
$\epsilon = 0.00$-greedy	**1.7**	**2.5**	**3.0**	144.1	245.8	249.2	329	466.4	484.5
$\epsilon = 0.05$-greedy	54.8	66.9	81.2	81.9	147.0	**232.7**	**166.8**	300.1	395.6
$\epsilon = 0.10$-greedy	109.6	132.4	160.6	117.8	191.7	279.4	**186.8**	317.8	408.3
$\epsilon = 0.15$-greedy	165	195.8	240.6	158.2	238.5	336.4	216.5	332.8	460.5
$\epsilon = 0.20$-greedy	218.6	262.1	323.2	203.0	302.9	408.2	249.9	385.6	516.9
$\epsilon = 0.25$-greedy	274.5	329.5	399.8	251.2	361.1	469.7	289.8	434.0	580.6
$\epsilon = 0.30$-greedy	326.5	394.0	482.1	296.9	423.3	543.8	331.3	489.0	639.5
$\epsilon = 0.35$-greedy	382.8	458.5	561.5	342.8	483.5	615.3	369.3	542.3	704.7
$\epsilon = 0.40$-greedy	437.9	525.6	638.1	390.0	548.1	686.4	410.1	596.9	764.4
$\epsilon = 0.45$-greedy	491.9	589.6	722.4	440.0	609.0	767.7	455.5	654.6	840.1
$\epsilon = 1$-Dgreedy	3.2	4.1	4.7	**98.7**	248.5	**245.8**	235.9	443.2	476.1
$\epsilon = 20$-Dgreedy	30.4	32.6	36.7	**59.2**	**101.7**	**210.2**	**149.9**	295.1	404.7
$\epsilon = 40$-Dgreedy	56.1	65.4	70.3	71.5	120.5	**194.5**	**152.6**	**253.4**	**363.7**
$\epsilon = 60$-Dgreedy	80.6	93.9	104.1	84.3	146.9	214	**158.5**	**254.3**	**344.0**
$\epsilon = 80$-Dgreedy	103.6	120.7	135.9	102.0	151.6	231.1	**167.3**	**246.3**	371.7
$\epsilon = 120$-Dgreedy	144.8	167.8	195.7	135.6	193.0	280.8	184.5	**269.0**	379.4
$\epsilon = 160$-Dgreedy	180.9	212	249.5	163.2	235.1	325.2	203.4	294.6	434.1
$\epsilon = 200$-Dgreedy	213	254.6	302.0	192.6	274.9	365.8	228.1	324.0	472.7
DP-greedy	**1.7**	**2.5**	**3.0**	**42.2**	**113.3**	**204.5**	**149.8**	**225.5**	**371.2**

Strategies	B-10	B-11	B-12	B-13	B-14	B-15
$\epsilon = 0.00$-greedy	599.3	534.9	603.6	**269216.5**	**218262.2**	**703681.6**
$\epsilon = 0.05$-greedy	281.9	384.9	559.9	304536.1	279927.3	632359.4
$\epsilon = 0.10$-greedy	270.7	382.5	561	397862.4	415458.7	910185.1
$\epsilon = 0.15$-greedy	300.1	401.8	592.4	483358.0	557590.3	1159969.6
$\epsilon = 0.20$-greedy	323.4	446.8	628.1	598526.7	701810.4	1429159.4
$\epsilon = 0.25$-greedy	349.0	490.2	684.3	711834.9	855150.9	1716417.5
$\epsilon = 0.30$-greedy	383.5	541.1	743.8	842439.0	999679.3	1992845.8
$\epsilon = 0.35$-greedy	424.1	587.3	796.8	972912.6	1157161.6	2254550.4
$\epsilon = 0.40$-greedy	464.2	639.2	839.9	1095672.3	1308751.1	2572096.9
$\epsilon = 0.45$-greedy	508.2	693.0	911.6	1225084.2	1464515.5	2853342.4
$\epsilon = 1$-Dgreedy	569.5	528.4	597.4	**271872.5**	**212768.5**	**694380.8**
$\epsilon = 20$-Dgreedy	240.3	394.4	**543.5**	**241194.0**	**209618.8**	**463682.9**
$\epsilon = 40$-Dgreedy	**215.1**	**347.8**	**532.5**	294683.3	268010.1	566396.1
$\epsilon = 60$-Dgreedy	**216.1**	**331.4**	**514.1**	**286044.7**	320430.4	687074.8
$\epsilon = 80$-Dgreedy	**212.2**	**326.0**	**497**	305760.5	368216.9	793232.9
$\epsilon = 120$-Dgreedy	**216.1**	**334.8**	**510.7**	380555.2	479727.5	978683.0
$\epsilon = 160$-Dgreedy	237.7	**365.4**	**519.6**	468543.4	576326.0	1153664.9
$\epsilon = 200$-Dgreedy	256.6	395.6	557.1	551275.7	670023.0	1327895.8
DP-greedy	**209.3**	**322.7**	585.9	**267489.2**	**224806.0**	**675138.0**

Table 4: The cumulative regret at round $H = 4000$ of the nineteen bandit strategies on the fifteen benchmark problems. Bold figures means that the method is not significantly different (p-value > 0.01) from the best according to a statistical paired t-test.

tests. The resulting bandit benchmarks are denoted B-13, B-14 and B-15.

The experimental results are shown in Table 4 and Figure 2. Table 4 returns for the nineteen bandit strategies and the fifteen benchmarks, the average of the cumulative regret over 100 repetitions at the round $H = 4000$. A bold value in table 4 means that, for a given benchmark, the average cumulative regret of a strategy is not significantly different (p-value > 0.01) from the regret of the best strategy according to a paired t-test.

Two main results have to be stressed:

1. Apart from the benchmark B-12, there is no significant difference between the DP-greedy performance and the one of the best strategy.

2. In ten benchmarks out of fifteen, DP-greedy belongs to the set of the best semi-uniform

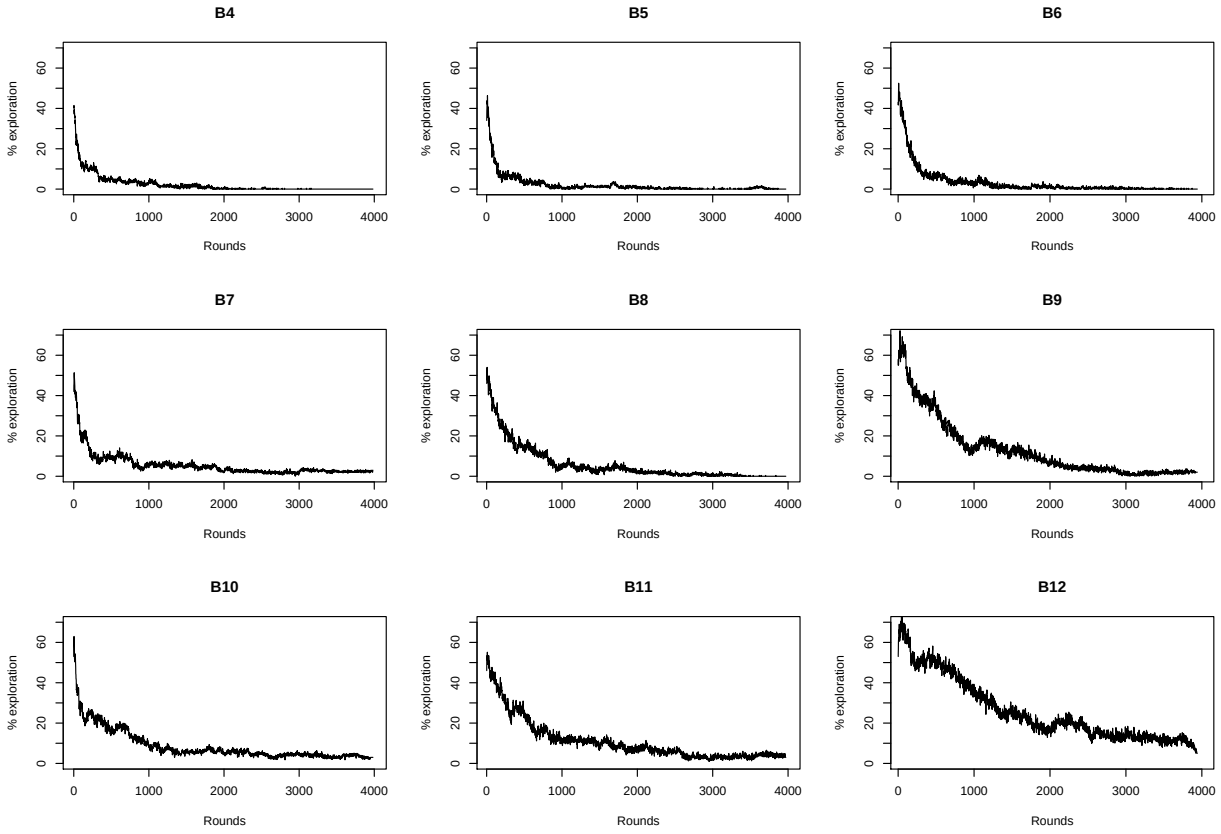

Figure 2: Evolution of the percentage of exploration during 4000 rounds of DP-greedy on the synthetic benchmarks. The percentage of exploration in B-1, B-2 and B-3 is always null.

strategies.

Using DP-greedy appears then to be equivalent to use an ϵ-greedy strategy with the optimal value of ϵ. In other words, the DP-greedy strategy appears to outperform semi-uniform state-of-the-art techniques since it attains the performance of the a-posteriori optimal ϵ-greedy strategies. The success of DP-greedy is explained by the adaptative behavior of DP-greedy which automatically adjusts the exploitation/exploration rate as a function of the bandit problem and his current state.

An additional insight about the behavior of DP-greedy is returned by the set of Figures 2 which show the curves of the evolution of the percentage of exploration actions during 4000 rounds for the benchmarks B-4 to B-12. Note that for the simplest bandit problems (i.e. B-1, B-2 and B-3), DP-greedy behaves as a pure exploitation algorithm which seldom explores the arms. Figure 2 shows that on more complex bandit problems (i.e. when either K or σ increases), DP-greedy automatically increases the exploration rate at the beginning and,

when more samples are collected, switches gradually to a pure exploitation mode.

8. Conclusion

In this paper, we propose a new semi-uniform algorithm for the bandit problem and we introduce the concept of the *expected greedy reward*. Instead of choosing directly an arm in a set of K alternatives (like classical bandit algorithms), a semi-uniform bandit algorithm chooses between two actions : a greedy exploitation action and a random exploration action. A major issue in such bandit algorithms is then how to trade exploitation and exploration.

In this work, we interpret the bandit problem as a Markov decision problem with perfect state information where only two actions are possible : greedy or random selection. The Markov decision problem is solved via dynamic programming techniques whose solution, in the case of perfect state information, is an algorithm which optimally balances greedy exploration and random exploitation actions.

We showed that the expected greedy reward is a quantity which plays a major role in the definition of the algorithm. However, since perfect state information is an unrealistic assumption for the bandit problem we introduce and discuss five methods to estimate, from historical data, the expected greedy reward. The availability of accurate estimators makes possible the definition of a running version of DP-greedy, to be adopted in real tasks with imperfect state information. This implemented version of DP-greedy addresses the curses of dimensionality problem of the dynamic program by reducing the recursive horizon. A set of real and synthetic benchmarks showed that the DP-greedy approach is a feasible, adaptive and effective way to solve the bandit problem in a semi-uniform way.

Future work will focus on alternative ways to solve the problem of the curses of dimensionality in DP-greedy. Although in this paper we adopted the solution of reducing the recursive horizon, we deem that neuro-dynamic programming techniques ((Bertsekas and Tsitsiklis, 1996)) could speed up the dynamic program by replacing the optimal gain function $J^*(\cdot)$ (eq. (5)) by a suitable approximation function.

Acknowledgments

The authors thank their colleague Abhilash Alexander Miranda for his appreciable comments.

References

Audibert, J.-Y., R. Munos, C. Szepesvri. 2006. Use of variance estimation in the multi-armed bandit problem. *NIPS 2006 Workshop on On-line Trading of Exploration and Exploitation*.

Auer, P., N. Cesa-Bianchi, P. Fischer. 2002. Finite-time analysis of the multiarmed bandit problem. *Machine Learning* **47** 235–256.

Auer, P., N. Cesa-Bianchi, Y. Freund, R. E. Schapire. 1995. Gambling in a rigged casino: the adversarial multi-armed bandit problem. *Proceedings of the 36th Annual Symposium on Foundations of Computer Science*. IEEE Computer Society Press, Los Alamitos, CA, 322–331.

Azoulay-Schwartz, R., S. Kraus, J. Wilkenfeld. 2004. Exploitation vs. exploration: choosing a supplier in an environment of incomplete information. *Decision support systems* **38** 1–18. doi:http://dx.doi.org/10.1016/S0167-9236(03)00061-7.

Bertsekas, D. P. 1987. *Dynamic Programming - Deterministic and Stochastic Models*. Prentice-Hall.

Bertsekas, D. P., J. N. Tsitsiklis. 1996. *Neuro-Dynamic Programming*. Athena Scientific.

Bishop, C.M. 2006. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer.

Caelen, O., G. Bontempi. 2007. Improving the exploration strategy in bandit algorithms. Vittorio Maniezzo, Roberto Battiti, Jean-Paul Watson, eds., *Learning and Intelligent OptimizatioN LION 2007 II*, *Lecture Notes in Computer Science*, vol. 5313. Springer, 56–68.

Caelen, O., G. Bontempi. 2008. On the evolution of the expected gain of a greedy action in the bandit problem. Tech. Rep. 589, Département d'Informatique, Université Libre de Bruxelles, Brussels, Belgium.

Efron, B., R. J. Tibshirani. 1993. *An Introduction to the Bootstrap*. Chapman and Hall, New York.

Gittins, J. C. 1989. *Multi-armed Bandit Allocation Indices*. Wiley.

Hardwick, J., Q. Stout. 1991. Bandit strategies for ethical sequential allocation. *Computing Science and Statistics* **23** 421–424.

Kim, S., B. Nelson. 2006. *Handbooks in Operations Research and Management Science: Simulation*, chap. Selecting the Best System. Elsevier Science.

Meuleau, Nicolas, Paul Bourgine. 1999. Exploration of multi-state environments: Local measures and back-propagation of uncertainty. *Machine Learning* **35** 117–154.

Perrone, M. P., L. N. Cooper. 1993. When networks disagree: Ensemble methods for hybrid neural networks. R. J. Mammone, ed., *Artificial Neural Networks for Speech and Vision*. Chapman and Hall, 126–142.

Powell, W.B. 2007. *Approximate Dynamic Programming - Solving the Curses of Dimensionality*. Wiley, Princeton, New Jersey.

Puterman, M. L. 1994. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley.

Robbins, H. 1952. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society* **58** 527–535.

Sutton, R.S., A.G. Barto. 1998. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA.

Tong, Y.L. 1990. *The Multivariate Normal Distribution*. Springer Verlag.

Vermorel, J., M. Mohri. 2005. Multi-armed bandit algorithms and empirical evaluation. *16th European Conference on Machine Learning (ECML05)*. ecml, 437–448.

Watkins, C. 1989. Learning from delayed rewards. Ph.D. thesis, Cambridge University.