Contextual Bandits: Approximated Linear Bayes for Large Contexts

Jose Antonio Martin H. JAMARTINH@FDI.UCM.ES
Dept. Computer Architectures and Automation, Complutense University of Madrid (UCM) , Madrid, Spain

Abstract

Contextual bandits, and in general informed decision making, can be studied in the general stochastic/statistical setting by means of the conditional probability paradigm where Bayes' theorem plays a central role. However, when informed decisions have to be made considering very large contextual information or the information is contained in too many variables with large history of observations and the time to take decisions is critical, the exact calculation of the Bayes' rule, to produce the best decision given the available information, is unaffordable. In this increasingly common setting some derivations and approximations to conditional probability and the Bayes' rule will progressively gain greater applicability. In this article, an algorithm able to handle large contextual information in the form of binary features for optimal decision making in contextual bandits is presented. The algorithm is analyzed with respect to its scalability in terms of the time required to select the best choice and the time required to update its policy. Last but not least, we address the exploration and exploitation issue explaining, despite the incomputability of an optimal tradeoff, the way in which the proposed algorithm "naturally" balances exploration and exploitation by using common sense.

1. Introduction

Contextual Bandits (a.k.a. bandits with covariate, side information, associative and associative reinforcement learning) (Li et al., 2010) or simply the reinforcement learning case when there are multiple states but reinforcement is still immediate (Kaelbling et al., 1996), are the natural extension of the multi-armed bandit problems firstly formalized by Robbins (1952) and exposed in detail by Gittins et al. (1989) and Berry & Fristedt (1986). Contextual bandits incorporates additional information (context) to the decision making process in the sense that the payoff (reward) obtained by playing an arm will be (up to some degree if not totally) dependent of such contextual information (i.e. a covariate). This kind of problem arises for instance in news recommendation systems (Li et al., 2010)

Following the terminology of Li et al. (2010) (with some minor variations): a contextual-bandit algorithm A proceeds in discrete trials $t = 1, 2, 3, \ldots T$ So that in trial t:

1. The algorithm observes a set $A(t)$ of arms (e.g., actions, options, choices) together with a feature vector $\mathbf{x}(t)$ (i.e., the *context*).
2. Based on observed payoffs from previous trials, A chooses an arm $a(t) \in \mathcal{A}(t)$, and receives payoff $r(t, a)$ whose expectation depends on both the context $\mathbf{x}(t)$ and the arm $a(t)$.
3. The algorithm then improves its arm-selection strategy from the tuple: context, arm and reward; $\mathbf{x}(t)$, $a(t)$ and $r(t, a)$ respectively.

So under this rules we must design an algorithm that maximizes the cumulative reward in the long-run. Traditionally, in bandit theory this maximization is defined in terms of minimizing the *regret* or loss with respect to an optimal-policy that always plays the best arm a^* at trial t, $a^*(t)$. Hence, a natural measure of the optimality in terms of regret $R_{\mathsf{A}}(T)$ for the algorithm A is (1):

$$R_{\mathsf{A}}(T) \quad \overset{\text{def}}{=} \mathbf{E}\left[\sum_{t=1}^{T} r(t, a^*)\right] - \mathbf{E}\left[\sum_{t=1}^{T} r(t, a)\right]. \quad (1)$$

Note that by making $\mathbf{x}(t)$ and $\mathcal{A}(t)$ constants the problem gets reduced to the classical k-armed bandit in which the paradigm is represented by the so called "exploration / exploitation tradeoff" (see Holland, 1975, March, 1991 or Kaelbling et al., 1996).

Appearing in *Proceedings of the 29th International Conference on Machine Learning*, Edinburgh, Scotland, UK, 2012.

Table 1. Some examples of equivalent terms to exploration and exploitation used in different fields[2].

Area or Discipline	exploration-observation	vs.	exploitation-prediction
Sequential decision making	exploration	vs.	exploitation
Compressed sensing	sensor-reading	vs.	data-prediction
Statistics and Machine Learning	memorizing data	vs.	generalizing
Curve-fitting	acquire-points	vs.	interpolation
Economics	risk-taking	vs.	risk-avoiding
Finance	investing	vs.	saving
Marketing	Diversification/Proliferation	vs.	Concentration strategy
Medicine	experimental treatments	vs.	safety and efficacy
Data-compression	store-data	vs.	space-savings

That is, the algorithm should find the right proportion between these two opposed "intentions":

1. Exploit the current knowledge of the task to select the best choice.
2. Explore a non-optimal choice to improve the knowledge of the task (if it is possible).

So, in general, each trial can be classified in one of these two categories: an exploration trial or an exploitation trial. However, in the contextual-bandit setting this becomes a tradeoff between feature-based-exploration and feature-based-exploitation, that is, a tradeoff for every context should be found, and not only simply along all the trials.

Although as commented by Sutton & Barto 1998, the soft-max algorithm per se tries to implement a softer dichotomy (or remove it at all), in strict sense it will actually end up, at each trial, selecting between the best choice and suboptimal ones based on a tradeoff defined by the "temperature parameter" of the Boltzmann-Gibs distribution, i.e., another kind of dichotomy after all, however of a different nature, i.e., the payoff that you will gain with a safer exploration will have as a price the information loss of the exploratory trial.

Let us now define the exploration and exploitation tradeoff for the total quantity of trials T, the number of exploratory trials $n(\epsilon)$ and the number of exploitation trials $n(\rho)$ in the following way:

$$n(\epsilon) = T - n(\rho), \tag{2}$$

$$n(\epsilon)/T = 1 - n(\rho)/T \tag{3}$$

$$\epsilon = 1 - \rho. \tag{4}$$

where ϵ, ρ are respectively the proportions of exploration and exploitation trials. It is clear that we want to minimize ϵ in order to get higher payoffs; however, what is the minimum value of ϵ in order to minimize the regret in the long-run? Table 1 show us a subtle clue!

Let us define $K(\varphi)$ to be the value of ϵ that maximizes the payoff under an ϵ exploration rate for a sequential decision problem (φ):

$$K(\varphi) = \underset{\epsilon}{\operatorname{argmax}} \left(\mathbf{E}^{\epsilon} \left[\sum_{t=1}^{T} r(t, a_{\epsilon}) \right] \right), \tag{5}$$

where a_ϵ is an exploration or an exploitation action following ϵ exploration rate.

Then, the following bad-news hold:

Theorem 1. *For any sequential task φ whose optimality depends on a tradeoff ϵ between exploration and exploitation: the optimal tradeoff $\epsilon^* = K(\varphi)$ is not a computable function.*

A proof of this theorem can be derived directly from the incomputability of Kolmogorov's complexity $K(s)$ of a string s, as well as from the fact that there is no possible general lossless compression scheme. Indeed, this last point tell us that there are tasks in which the unique possible optimal solution is a pure exploration approach since there will be problems in which learning (and so prediction) is impossible at all.

Simply, in general, we can't encode any sequence (s) of length $\ell(s)$ in any sequence (ϵ) of length $\ell(\epsilon) < \ell(s)$. Since any run of a sequential task (φ) defines (obviously) a sequence (s_φ), then we cannot in general find any shorter sequence that would predict s_φ (for any

[2] This equivalence list should be taken in a rigorous mathematical sense. I am very interested in improving it with new meaningful terms, so, please, updates, suggestions and discussions are very welcome!

non-trivial s_φ). That is, in general, a shorter sequence than s_φ that specify an optimal exploration / exploitation tradeoff (i.e., ϵ^*) that predicts the best possible playing sequence s_φ (which is simply just one particular sequence) doesn't exists. Otherwise, we would be able to create a universal lossless compression program by encoding strings as particular playing sequences determined by shorter strings, such as for instance a short description of an optimal exploration / exploitation tradeoff.

However, despite these bad-news, in a sense, this is a full employment theorem for bandits and so it is possible to find suboptimal exploration rates and exploration / exploitation balancing techniques that significantly improve learning. Having said that, the subject of this article is to present an algorithm that "naturally" balances exploration and exploitation in an intuitive and effective way. It should be emphasized the importance of finding simple enough and intuitive explanations of effective exploration / exploitation techniques for many obvious reasons, for instance, Kuleshov & Precup (2010) show that very simple techniques such as ϵ-greedy and soft-max perform extremely competitive when compared to more elaborate and theoretically-regret-guaranteed-proved techniques which are far from being intuitive for the uninitiate despite their simple and elegant key-ideas.

We can study general informed-sequential-decision-making under the stochastic/statistical framework by means of the posterior probability paradigm where Bayes' theorem plays a central role. However, when such informed decisions must be computed from very large contextual information or the information is contained in too many feature variables that may contain as well a large history of observations and also the time to take decisions is critical; the exact calculation of the Bayes' rule to produce the optimal decision, given the available information, is computationally unaffordable.

Nowadays, this is an increasingly common picture and hence derivations and approximations to conditional probability and to the Bayes' rule will gain progressive interest. Here, an algorithm able to efficiently handle large contextual information in the form of binary features that naturally balances exploration and exploitation in contextual bandits is presented.

The algorithm is derived from intuitive observations that converge to a "linear" approximation to the Bayes' rule. The algorithm is analyzed with respect to its scalability in terms of the time required to select the best choice and the time required to update its policy. We address as well, how the proposed algorithm "naturally" balances exploration and exploitation using common sense arguments.

2. Algorithm: Linear Bayes' rule for contextual bandits

The current paper develops under the frame the "Exploration and Exploitation 3 Challenge"[3]. So taking advantage of this context is a good opportunity to describe the proposed algorithm in terms of the proposed challenge, i.e., serving news articles on a web site[4]. Here the quantity to be optimized is the overall click through rate (CTR) of the algorithm, that is, the number of trials in which the user clicked the recommended article.

The presentation of the algorithm would be progressive so that the final form will be easily deduced and understood. The algorithm handles contextual information (visitor features) in the form of a vector of binary features $\mathbf{x}(t) \in \{0,1\}^*$ provided at each trial t when also the algorithm is confronted to the selection of just one article from a set $\mathcal{A}$ of possible actions.

The algorithm is divided in three main (sub)routines or functions:

1. Article recommendation (getActionToPerform): the algorithm at trial t has to select one article $a(t)$ to recommend from the list $\mathcal{A}(t)$ of possible actions taking advantage (if possible) of the visitor features $\mathbf{x}(t)$ (the context).

2. "Preference-values" computation (p_value): upon request this function computes a preference-value for an article a given the context $\mathbf{x}(t)$ at trial t, i.e., $v = p_v(a, \mathbf{x}, t)$. The returned value v can be used to produce a ranking over the set $\mathcal{A}(t)$ in order to select the article with a higher "preference-value".

3. Policy update (updatePolicy): once the selected article is recommended (and if it coincides with the data, see Li et al., 2010 for evaluation methodology), a feedback in the form of a binary number $r \in \{0,1\}$ is received and it is used to update the necessary statistical information required by the "Preference-values" computation.

[3] New Challenges for Exploration and Exploitation workshop at the International Conference on Machine Learning - June 26–July 1, 2012 - Edinburgh, Scotland. http://icml.cc/2012/

[4] https://explochallenge.inria.fr/the-evaluation-process/

2.1. Naive approach I

The most naive algorithm is to forget the context and just select always at trial t the article $a(t)$ which historically received more clicks $r = 1$. This algorithm can be implemented just by having a counter ($clicks[a]$) for each article a.

$$a(t) = \operatorname*{argmax}_{a} \Big(clicks[a] \Big). \tag{6}$$

However, if different articles are recommended (and selected) in different proportions, this criterion is unfair since an article (a_1) recommended (and selected), say, on 1000 trials having received only 10 clicks will be preferred to an article (a_2) that have been recommended (and selected) only on 10 trials but received 9 clicks. It is of common sense to feel a preference for the a_2 article.

2.2. Context-free: naive approach II

The second approach is to maintain also a counter ($selections[a]$) of the quantity of trials in which each article have been recommended (and selected), so that a proportion $clicks[a]/selections[a]$ could be calculated and used as the preference criterion. A key-advantage of this second (proportion-based) approach is that it allows to "learn" preferences even with a very different selection rate for each article, and so it creates room for balancing exploration and exploitation.

$$P_a = \frac{clicks[a]}{selections[a]}, \tag{7}$$

$$a(t) = \operatorname*{argmax}_{a} \Big(P_a \Big). \tag{8}$$

This approach, although extremely basic, will perform well under the assumptions that: (1) the preferences for the recommended news are universal across all the users of the web site and (2) that a well enough exploration / exploitation tradeoff is used.

Condition 1 is extremely restrictive and it may only happen in very specialized contexts in which additional contextual information is redundant, i.e., a non-contextual bandit (a simple k-armed bandit), which is not the current case. Condition 2 can be addressed in many ways since there are many alternatives and combinations between them, e.g., ϵ-greedy, soft-max, UCB (see Kuleshov & Precup, 2010 for a comparison of some).

Here we simply use "optimistic initial values" (see Sutton & Barto, 1998, chap. 2.7) to force exploration, which is a simpler approach to the key-idea of "optimism in front of uncertainty" implemented in UCB-like algorithms. This method can be implemented simply by assuming, as a staring point, that every article has been selected and clicked one time, i.e., a proportion of 1:

$$initial\ values \begin{cases} clicks[a] & = 1, \\ selections[a] & = 1, \end{cases} \tag{9}$$

for any non-previously selected articles a.

This exploration strategy works in the following way, at trial 1 it selects the first available article a_1 since all non yet recommended articles have preference 1 at starting point; and continue to recommend a_1 until a no click event occur in which a case the preference of a_1 will be less than 1. Then at the next trial the next article with preference 1 is recommended. This cycle continues until all articles adapt its preference estimate very close to the true click-rate of every article.

2.2.1. Incremental proportions

Some optimizations can be made in order to avoid the computation of the proportion (7) at every query. For this purpose, it is just needed to maintain the current proportion $P[a]$ for each article a continuously updated:

$$P_a(t+1) = P_a(t) + \frac{r(t) - P_a(t)}{selections[a] + 1}, \tag{10}$$

where $r(t) = 1$ indicates that article a was clicked at trial t and $r(t) = 0$ if not. Note that equations 7 and 10 compute exactly the same value and that (10) is equivalent to the pursuit methods and the well known single-step temporal difference equation of reinforcement learning:

$$p_i(t+1) = p_i(t) + \alpha * [r - p_i(t)], \tag{11}$$

where α is known as the learning rate parameter and r is the target to be learned (in this case 0 or 1).

2.3. Contextual bandits: a naive approach III

Let us now see how to incorporate in a useful way the available contextual information $\mathbf{x}(t)$. The first key-idea is quite simple:

Let us assume that each context $\mathbf{x}(t)$ define some characteristic features of a group of users; such as time-zone, country, language, previous navigation history and even direct knowledge about the kind of news they prefer to read. Hence, there would be some contexts (user-groups from now on) $\mathbf{x}(t)$ that are more likely to click on certain articles than on others. Here the assumption is that each binary feature give us information of a particular fact and so each feature should be treated as a positive evidence of a visitor belonging to a certain user-group, i.e., the features are independent and non-mutually exclusive: a user-group is defined by a set of features that individual users may have in common but not that must have in common, for example, a user-group interested in sports may have interest in baseball OR football OR tennis instead of being defined as having interest in baseball AND football AND tennis. In this case, there will be preference for a sports related news when there is a preference for baseball OR football and, more importantly, this preference will be maximized if the visitor shows preference for baseball AND football.

Hence, the visitor preference for a particular article can be measured simply by adding up all the individual preferences for each feature $\mathbf{x}_i$: $P_{a,i}$, i.e., a preference (feature) is specified when the i^{th} element of $\mathbf{x}_i = 1$ and so on:

In this case, $clicks[a][i]$ accumulates clicks for article a when $\mathbf{x}_i(t) = 1$ (i.e., the feature is present) and $r(t) = 1$, while $selections[a][i]$ does the same independent of the value of $r(t)$:

$$clicks[a][i] = \sum_t \mathbf{x}_i(t) r(t); \text{ for } a(t) = a \tag{12}$$

$$selections[a][i] = \sum_t \mathbf{x}_i(t); \text{ for } a(t) = a. \tag{13}$$

And hence the incrementally calculated proportion is:

$$P_{a,i}(t+1) = P_{a,i}(t) + \mathbf{x}_i(t) \frac{r(t) - P_{a,i}(t)}{selections[a][i] + 1}. \tag{14}$$

And the recommended article $a(t)$ at trial t is:

$$a(t) = \underset{a}{\operatorname{argmax}} \left(\frac{\sum_i clicks[a][i]}{\sum_i selections[a][i]} \right), \tag{15}$$

$\forall i \; s.t. \; \mathbf{x}_i(t) \neq 0$, or, alternatively:

$$a(t) = \underset{a}{\operatorname{argmax}} \left(\sum_i P_{a,i}(t) \right), \tag{16}$$

$$\text{where } P_{a,i} = \left(\frac{clicks[a][i]}{selections[a][i]} \right). \tag{17}$$

$\forall i \; s.t. \; \mathbf{x}_i(t) \neq 0$.

2.4. Contextual bandits: a common sense approach IV

Now, an extension to the last preference measure is derived from an intuitive observation: instead of basing the preference in the simple summation of the proportions $\sum_i P_{a,i}(t)$ between clicks to specific article a by some user-group and the number of times this article has been recommended to this user-group; what about if we find a way of determining which specific features and so which specific proportion $P_{a,i}(t)$ should contribute in a mayor degree to the overall sum?

Let us define the overall click-rate for a feature i as:

$$P_i(t+1) = P_i(t) + \mathbf{x}_i(t) \frac{r(t) - P_i(t)}{1 + \sum_a selections[a][i]}. \tag{18}$$

In this sense, what feature should indicate or predict in a mayor degree a user-click?

a) A feature i whose click-rate for article a_1 is very high and whose overall click-rate is very low? Or,

b) A feature i whose click-rate for article a_1 is high and whose overall click-rate is very high? Or,

c) A feature i whose click-rate for article a_1 is very low and whose overall click-rate is very high? Or,

d) A feature i whose click-rate for article a_1 is very low and whose overall click-rate is low?

These are some common sense answers to the above list:

(a) Indicates that almost all clicks related to feature i are going to article a_1.

(b) Indicates that only a relatively small fraction of clicks related to feature i are going to article a_1.

(c) Indicate that almost no click related to feature i is going to article a_1

(d) Indicate that only a relatively small fraction of clicks related to feature i are going to article a_1.

Even more, what role plays the overall click-rate $P_a(t)$ of article a in this puzzle? I.e, a feature i whose click-rate for article a is very high and whose overall click-rate is very low while the overall click-rate $P_a(t)$ of article a is high or low?

This is also a common sense question: if we assume $P_a(t)$ is very low; it indicates that, although almost all clicks related to feature i are going to article a, the difference with respect other features can't be much significant since as assumed the overall click-rate of action a is very low, hence $P_a(t)$ is a measure of up to which level this weighted average preference will differ from the naive summation, i.e., if probabilities are to much low then there will be no practical difference between the two methods.

Hence, to create such a weighted average, the idea is just weighting each $P_{a,i}(t)$ by the inverse of the overall click-rate of feature i and $P_a(t)$:

$$a(t) = \underset{a}{\operatorname{argmax}} \left(\sum_i \frac{P_{a,i}(t)}{P_i(t)} P_a(t) \right), \tag{19}$$

where $P_a(t)$ goes out the summation because it remains constant.

From this, we can make the following equivalences with the standard probability theory notation:

$$P(\mathbf{x}_i|a) = P_{a,i}(t) \tag{20}$$

$$P(\mathbf{x}_i) = P_i(t), \tag{21}$$

$$P(a) = P_a(t), \tag{22}$$

$$P(a| \cup \mathbf{x}_i) = \sum_i \left(\frac{P(\mathbf{x}_i|a)}{P(\mathbf{x}_i)} \right) P(a) \tag{23}$$

So the following three equations are equivalent:

$$a(t) = \underset{a}{\operatorname{argmax}} \left(\sum_i \frac{P_{a,i}(t)}{P_i(t)} P_a(t) \right), \tag{24}$$

$$a(t) = \underset{a}{\operatorname{argmax}} \left(\sum_i \frac{P(\mathbf{x}_i|a)}{P(\mathbf{x}_i)} P(a) \right), \tag{25}$$

$$a(t) = \underset{a}{\operatorname{argmax}} \left(P(a| \cup \mathbf{x}_i) \right). \tag{26}$$

And therefore the applied preference selection is a linear approximation to the Bayes' rule, i.e., for the union of the informative events. Finally a slight common sense variation is to include an additional term $P(\cap \mathbf{x}_i|a)$ defined as:

$$P(\cap \mathbf{x}_i|a) = \prod_i P(\mathbf{x}_i|a), \tag{27}$$

that expresses the joint probability of all posteriors $P(\mathbf{x}_i|a)$ assuming independence. Hence the final recommendations of the presented algorithm are done in the following way:

$$a(t) = \underset{a}{\operatorname{argmax}} \Big(P(a| \cup \mathbf{x}_i) P(\cap \mathbf{x}_i|a) \Big). \tag{28}$$

2.5. Exploration / Exploitation: Does exactly what it says on the tin

How does this algorithm explore? How does it balances exploration and exploitation? In previous sections it was described how optimistic initial values induce a natural exploration that converge to near optimal click-rates. However, by applying the final algorithm (28) things are slightly different. So, again, common sense can be applied to analyze how a natural tradeoff occurs.

Let us suppose we want to create a special exploration procedure that tries to gain as much information as possible from every exploratory trial. The following is a powerful but quite simple idea for what could be a good exploratory trial: recommend an article a such that the following *concurrent* conditions hold:

1. Article has been little selected by the current user-group $\mathbf{x}(t)$.
2. Article has near the maximum selection rate over all the other user-groups.
3. Despite it's low inner-group selection rate it shows some clicks for the current user-group.
4. The article has a high prior, i.e., $P(a)$ is high.
5. The article has received little clicks over all the user-groups.

That is, an extremely informative trial. For instance, any unrecommended article or a new one just arriving on the web site (a truly news article) will have maximal priority to be explored by using the above list of desired conditions. However, to understand the complete picture, let us observe the following equations which are equivalent to (25):

$$a(t) = \underset{a}{\operatorname{argmax}} \left(\sum_i \frac{C_{ai}/S_{ai}}{C_i/S_i} P(a) \right) \tag{29}$$

$$= \underset{a}{\operatorname{argmax}} \left(\sum_i \frac{C_{ai} S_i}{C_i S_{ai}} P(a) \right), \tag{30}$$

where $C_{ai} = \mathit{clicks}[a][i]$, $S_{ai} = \mathit{selections}[a][i]$, $C_i = \sum_a \mathit{clicks}[a][i]$ and $S_i = \sum_a \mathit{selections}[a][i]$.

We can see that, indeed, these equations maximize all the conditions above, e.g. the term S_i as well as C_{ai} and $P(a)$ are directly proportional to the selection preference (conditions 2,3,4), however C_i and S_{ai} are inversely proportional to the selection preference (conditions 1 and 5).

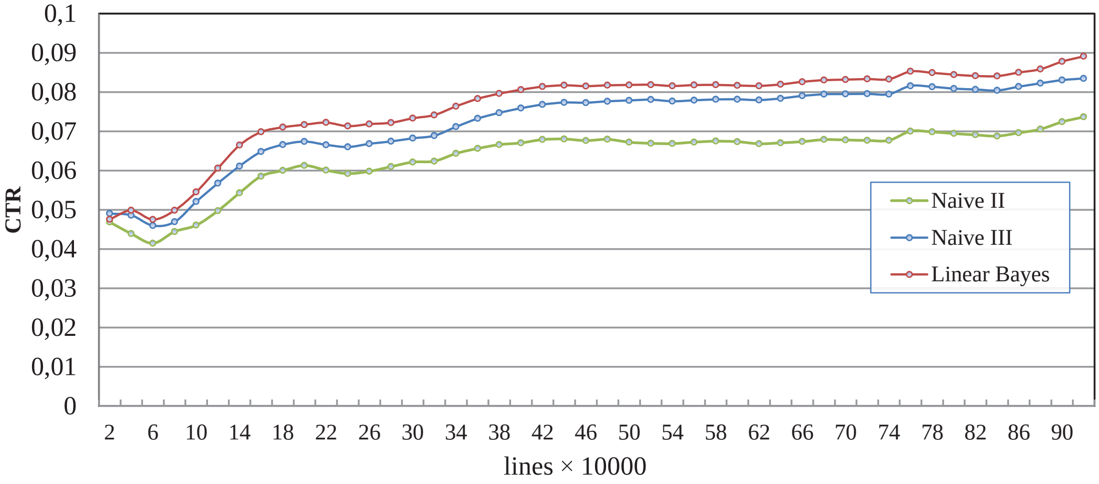

Figure 1. Performance comparison of the explained methods

Therefore, in the presented algorithm converge the two opposed "intentions" in just one selection rule; it explores and exploits depending on the particular contexts and the particular information associated to that context in a specific time. Indeed, all the attempts (by now) to combine this algorithm with other complimentary exploration / exploitation procedures, such as ϵ-greedy or soft-max, have failed to beat the selection rule (28).

As a final remark, it is very important to mention that the algorithm scales linear in the number of binary features and also scales linear in the number of articles to choose from. These are definitely the mayor advantages (together with its predictive performance) of the presented approach to contextual bandits with large binary features context.

Finally, figure 1 shows the different performances obtained for the naive II, naive III and Linear Bayes approaches implemented so far.

Acknowledgments

This work is partially supported by Spanish CICYT Project: DPI2009-14552-C02-01 (Surveillance, search and rescue at sea through collaboration of autonomous marine and aerial vehicles).

References

Berry, DA and Fristedt, B. Bandit problems. *Journal of Applied Statistics*, 13(2), 1986.

Gittins, J.C., Weber, R., and Glazebrook, K.D. *Multi-armed bandit allocation indices*, volume 25. Wiley Online Library, 1989.

Holland, J.H. *Adaptation in natural and artificial systems*. Number 53. University of Michigan press, 1975.

Kaelbling, L.P., Littman, M.L., and Moore, A.W. Reinforcement learning: A survey. *Arxiv preprint cs/9605103*, 1996.

Kuleshov, V. and Precup, D. Algorithms for the multi-armed bandit problem. *Journal of Machine Learning*, 2010.

Li, L., Chu, W., Langford, J., and Schapire, R.E. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pp. 661–670. ACM, 2010.

March, J.G. Exploration and exploitation in organizational learning. *Organization science*, pp. 71–87, 1991.

Robbins, H. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5):527–535, 1952.

Sutton, R.S. and Barto, A.G. *Reinforcement learning: An introduction*, volume 1. Cambridge Univ Press, 1998.