Approche contextuelle par régression pour les tests A/B

Emmanuelle Claeys*, Pierre Gançarski *
Myriam Maumy-Bertrand**

*ICube – Université de Strasbourg – 67412 – Illkirch – France
{claeys, gançarski}@unistra.fr
**IRMA – Université de Strasbourg – 67084 – Strasbourg – France

Résumé. Les tests A/B sont des procédures utilisées par les entreprises du web et de la santé entre autres, pour mesurer l'impact d'un changement de version d'une variable par rapport à un objectif. Bien qu'un nombre de plus en plus important de données soit disponible, la mise en place concrète d'un tel test peut impliquer un coût important relatif à l'observation et à l'évaluation d'une variation lorsque celle-ci n'est pas optimale.

Dans ce papier, nous présentons une nouvelle approche intégrant le principe d'un bandit contextuel prenant en compte ces variables via une procédure de stratification.

1 Introduction

Dans de nombreux domaines économiques, industriels voire sociaux, il peut être intéressant d'évaluer l'impact d'un changement sur un gain attendu.

Il devient alors nécessaire d'établir un processus permettant d'évaluer l'impact de différentes alternatives d'une entité (médicament, page web,...) sur les gains et ainsi de choisir la plus optimale en fonction de ces derniers. Un *Test A/B* consiste à évaluer concrètement ces différentes alternatives par rapport à un objectif défini a priori. Pour cela, dans une première phase, bornée dans le temps, appelée *phase d'exploration*, les items (objets, personnes visées par test,...) sont soumis à l'une des alternatives de façon irrévocable. Ainsi, dans le cadre d'un test A/B sur une page web, un certain nombre fixé de visiteurs verront exclusivement la version A de la page, et l'autre partie des visiteurs la version B, jusqu'à la fin du test, et ce même en cas de revisite. À la fin de cette période, les gains cumulés des différentes alternatives (par exemple la somme des conversions ...) sont comparés. L'estimation du gain moyen associé à chaque alternative se compare en général avec le *regret cumulé*, défini comme la différence entre l'alternative optimale (celle qui maximise le gain total) et les alternatives proposées. La meilleure alternative est alors choisie pour la *phase d'exploitation*, c'est-à-dire celle mise en production pour les nouveaux visiteurs.

Une première approche dite fréquentiste nécessite d'anticiper correctement le temps nécessaire pour conclure sur la différence entre les variations et donc déterminer l'alternative optimale. Elle impose aussi de choisir un ratio fixe d'affectation des items à chaque alternative. Or il peut être intéressant de changer ce ratio au cours du test. En effet, il se peut qu'une alternative réalise très rapidement un gain cumulé important : il convient alors de terminer la phase

de test (si cette alternative est optimale) ou de favoriser l'affectation des items à cette alternative. A contrario, une alternative présentant des résultats très médiocres, c'est-à-dire s'avérant sous-optimale, devra être désavantagée afin de limiter les pertes ou le manque à gagner.

Pour contourner ce problème, de nombreuses méthodes adoptent un mécanisme d'*allocation dynamique*. Ce mécanisme consiste à adapter le ratio d'affectations afin de basculer automatiquement l'affectation des items vers l'alternative optimale lorsqu'elle est identifiée.

Dans le domaine du test A/B, les modèles de bandits sont des stratégies d'allocation dynamique très utilisées.

Cependant, la quantité d'informations disponibles, autrement dit la taille du vecteur contextuel, influe fortement sur la performance des modèles de bandits contextuels (Chu et al., 2011). Par ailleurs, certaines caractéristiques peuvent être bruitées voire inutiles ou au contraire avoir un fort impact sur le gain généré par un item. Enfin, l'information peut être quantitative ou bien qualitative et dans ce dernier cas augmenter fortement la complexité.

Dans cet article, nous proposons un algorithme sélectionnant les informations les plus pertinentes pour identifier des sous-populations homogènes sur chacune desquelles un modèle de bandits indépendants est alors appliqué.

La section 2 de cet article introduit le principe du modèle de bandits contextuels et présente les principales méthodes existantes sur lesquelles nous nous appuyons. La section 3 donne notre proposition d'algorithme `CTREE-UCB`. La section 4 présente les résultats expérimentaux. Enfin, la section 5 conclue et donne les perspectives de nos travaux de recherche.

2 État de l'art sur l'allocation dynamique

Une première stratégie, ϵ-greedy, (Auer et al., 2002a) consiste à allouer aléatoirement une proportion ϵ du trafic à la meilleure alternative et le reste aux autres alternatives existantes.

D'autres approches comme `EXP3` (Auer et al., 2002b) ou `UCB` (Auer et al.a, 2002) utilisent la borne supérieure du gain moyen estimé de chaque bras pour affecter les items à la meilleure alternative. Après plusieurs itérations, l'intervalle de confiance converge pour chaque bras vers le gain moyen.

Les bandits *contextuels* estiment la moyenne du gain de chaque bras selon des informations caractérisant un item. Le plus utilisé `LinUCB` construit une régression linéaire à partir du contexte pour estimer les moyennes des gains obtenus par chaque bras.

Cependant, l'utilisation de ces bandits reste difficile pour plusieurs raisons.

En effet, la complexité des algorithmes de bandits contextuels augmente avec la dimension du vecteur contextuel. De plus, la fonction de gain réel n'est pas nécessairement une fonction linéaire du vecteur contextuel. Enfin, identifier les données contextuelles qui ont un réel impact sur la récompense est une problème en soi.

En pratique, ces méthodes nécessitent donc une analyse préalable de la pertinence des données contextuelles qui s'avère, dans la plupart des cas, difficile à réaliser.

L'amélioration apportée par des modèles de bandits lorsqu'ils sont appliqués indépendamment sur différents sous-groupes a déjà été prouvée par Maillard et Mannor (2014).

Soient $\mathcal{A}$ un ensemble fini de $N_{\mathcal{A}}$ bras et $\mathcal{B}$ un ensemble fini de $N_{\mathcal{B}}$ sous-groupes de N items avec $N_{\mathcal{B}} \leqslant N$. Chaque item n est défini par un vecteur contextuel f_n.

Nous définissons $\{\nu_{a,b}\}_{a\in\mathcal{A},b\in\mathcal{B}}$ comme une distribution de probabilités réelles dans l'intervalle $[0;1]$ de moyenne $\mu_{a,b}\in\mathbb{R}$ telle que $\nu_{a,b}$ soit R-sous-gaussienne :

$$\forall\lambda\in\mathbb{R},\qquad \log\mathbb{E}_{\nu_{a,b}}\exp(\lambda(X-\mu_{a,b}))\leqslant R^2\lambda^2/2 \tag{1}$$

où R est une constante positive

et X le gain produit par un item.

La première idée de diviser les visiteurs en différents sous-ensembles provient de Agrawal et al. (1989).

Intuitivement, le bras choisi pour l'exploitation n'est pas nécessairement optimal pour un sous-ensemble de la population testée. Cela conduit à un regret linéairement croissant dépendant du gap $\Delta^2_{a,b'}$.

Pour identifier et construire des groupes homogènes, de nombreuses techniques existent : le lecteur trouvera une liste détaillée dans Kotsiantis (2007). L'une des méthodes qui nous a semblé la plus prometteuse et la plus adaptée à notre problème est basée sur l'apprentissage d'un arbre construit par partitionnements récursifs. Cette technique permet d'estimer différentes moyennes pour des variables explicatives spécifiques et de les identifier à travers une analyse de régression (Strasser et Weber, 1999). Des algorithmes comme `CART` (Breiman et al., 1999) et `C4.5` (Salzberg, 1994) utilisent cette technique.

Enfin, `CTree` (conditional inference Tree), dérivé de `C.A.R.T` (Hothorn et al., 2006), est un algorithme non paramétrique intégrant des modèles de régression arborescents à travers des procédures d'inférences conditionnelles.

3 Contribution

Maillard et Mannor (2014) ont proposé l'algorithme `Single-K-UCB` qui affecte les items à différents sous-groupes et associe des modèles de bandits indépendants pour chacun de ses sous-ensembles. Ils supposent ainsi défini a priori l'ensemble $\mathcal{B}$ des sous groupes. De plus, ils supposent que les sous-groupes sont homogènes et que les distributions $\{\nu_{a,b}\}_{a\in\mathcal{A},b\in\mathcal{B}}$ sont connues. Dans ce cas, ils ont montré que le regret cumulé est borné par le regret d'une sous-population pour laquelle le bras choisi n'est pas optimal.

Les auteurs ont aussi prouvé que si des sous-groupes pouvaient être associés à des bras optimaux différents, l'utilisation de modèles de bandits indépendants pour chaque sous-groupe permet de réduire le regret cumulé.

Néanmoins, si ces méthodes ont prouvé leur efficacité dans certains cas, elles restent fortement dépendantes de données contextuelles utilisées. Un mauvais choix de celles-ci peut amener à des sous-groupes non pertinents et donc à conserver un regret linéaire continu dû à un ou plusieurs sous-groupes non identifiés.

Dans leurs approches la distribution des sous-groupes doit être connue à priori. Une autre hypothèse est que les sous-groupes aient des probabilités de gain différentes. Cependant, un utilisateur peut proposer des sous groupes ne respectant pas ces propriétés. Un tel algorithme ne sera alors pas plus performant qu'un algorithme non contextuel.

Ainsi, nous nous proposons de baser la construction de ces groupes sur différentes estimations de la récompense : l'idée est que des items produisant un même gain, indépendamment des alternatives possibles, doivent faire partie d'un même sous-groupe. Pour réaliser cette opération, nous nous proposons d'utiliser une méthode d'apprentissage supervisée.

Nous proposons donc d'identifier automatiquement les variables les plus discriminantes pour ainsi identifier des sous-groupes homogènes en utilisant uniquement une distribution de gain issue d'un bras (qui peut par exemple être obtenue à partir de l'alternative originale au préalable du test).

Nous appliquerons ensuite un modèle de bandit à chaque sous-groupe lors de la phase d'exploration.

Pour cela, nous proposons d'identifier toutes les covariables pertinentes des items pour créer automatiquement ces sous-groupes optimaux sous un critère α de confiance.

Dans la suite, nous examinons comment la performance d'un algorithme de bandit peut être améliorée si elle s'applique à un sous-groupe où la probabilité de distribution du gain (indépendamment d'un bras) est très similaire entre chaque item de ces sous-groupes.

Soit une fonction de gain suivant une distribution de Bernoulli pour différents sous-groupes $b \in \mathcal{B}$ avec un bras optimal $\star_b$ existant, qui peut-être possiblement différent, pour chaque sous groupes.

Dans notre approche, nous identifions des sous-groupes qui maximisent la divergence de Kullback Leibler entre deux distributions $\nu_{a,b}$ et $\nu_{a,b'}$: $\mathrm{KL}(\nu_{a,b}, \nu_{a,b'})$. Ces groupes sont basés sur les covariables f observables pour des items soumis à au moins un seul bras.

En pratique, cette approche, contrairement à d'autres algorithmes tels que `LinUCB`

— permet d'identifier les covariables qui influencent la distribution, et n'utilise pas les variables non pertinentes,

— présente une complexité moindre dans la mesure où elle utilise un nombre réduit de covariables et où elle estime l'intervalle de confiance du gain de chaque bras [1],

— est parallélisable du fait de l'utilisation de modèles de bandits indépendants.

Dans certains domaines comme le marketing, un léger changement sur une page web peut avoir en pratique un très faible impact sur le désir d'achat d'un visiteur. Si une grande partie des visiteurs ne n'est pas affectée par le test, dans le meilleur des cas, les visiteurs seront tous classés dans un sous-groupe, dans le pire des cas, notre algorithme ne sera pas moins performant que les algorithmes de bandits existants, soumis à un regret linéaire. En revanche, nous supposons que l'environnement est stationnaire, ce qui signifie qu'un bras optimal ne peut devenir sous-optimal au cours du temps.

4 Expérimentations

Dans cette section, nous comparons `CTREE_UCB` à `Lin-UCB` et `UCB`.

Nous avons testé avec différentes configurations (différents bras, nombre d'itérations, contextes et fonctions de gain) avec une version de notre approche développée en langage libre `R`. Nous avons utilisé notamment, le jeu de données UCR (Car Evaluation Database).

Une première série de tests a été effectuée sur la base de données The Car Evaluation Database qui contient six attributs d'entrée : achat, maintenance, portes, personnes, ergot, sécurité. Nous avons sélectionné 100, 500, 700 et 1000 premiers éléments pour construire les différents jeux d'entraînement.

1. pour `LinUCB` il est nécessaire d'inverser la matrice M de co-variance $O(M^3)$, ce qui peut aussi être très coûteux lorsque nous avons un nombre élevé de covariables

taille du learn set	nb de sous groupes [2]	$\frac{\text{Regret cumulé}}{\text{Iterations}}$		
		Ctree_UCB	UCB	LinUCB
100	1	0.26 [0.23 $\pm$ 0.29]	0.26 [0.23 $\pm$ 0.29]	**0.25** [0.22 $\pm$ 0.28]
500	3	**0.27** [0.24 $\pm$ 0.30]	0.36 [0.32 $\pm$ 0.40]	0.33 [0.29 $\pm$ 0.37]
700	3	**0.22** [0.19 $\pm$ 0.24]	0.32 [0.29 $\pm$ 0.35]	0.30 [0.27 $\pm$ 0.33]
1000	4	**0.23** [0.20 $\pm$ 0.26]	0.45 [0.39 $\pm$ 0.51]	0.41 [0.35 $\pm$ 0.47]

TAB. 1 – *Expérimentation avec quatre variables contextuelles pour le dataset car*

Nous associons à l'acceptabilité de la voiture la récompense à maximiser. L'accessibilité des voitures présente 4 niveaux associés à la valeur différente du gain possible : *unacc*, *acc*,*good*, *vgood* : (-1,0,1,2).

Nous cherchons ici à proposer la taille du coffre à bagages (*lug_boot*) qui maximise l'acceptabilité de la voiture. Il y a 3 bras possibles, représentant 3 tailles de coffre différentes (small, med, big).

Nous sélectionnons uniquement les items ayant $lug\_boot = small$ issus du jeu d'entraînement. Les items des autres bras sur le jeu d'entraînement sont alors ignorés pour l'apprentissage.

Le tableau 1 renvoie les résultats selon la taille du jeu d'apprentissage. Le regret est calculé à partir de la moyenne des récompenses réalisées par les items ayant les mêmes caractéristiques.

Des expérimentations complémentaires sont actuellement réalisées pour déterminer la performance de l'algorithme `CTree_UCB` en faisant varier la taille du jeu de d'apprentissage, le nombre de bras, le type de variables explicatives, l'intervalle de confiance (I.C.) utilisé pour la génération de l'arbre, le type de gain et la différence entre les bras sur la fonction de gain.

Dans de nombreuses expériences, nous voyons que `CTree_UCB` concurrence `Lin-UCB` et `UCB`. De plus, il est possible de savoir quel bras est le plus adapté à chaque sous-groupe et de poursuivre l'exploration si le meilleur bras n'a pas encore été identifié.

5 Conclusion

Nous avons proposé un nouvel algorithme de bandits et montré comment il peut être utilisé pour résoudre certains problèmes concernant les tests personnalisés. Notre méthode propose d'explorer un nouveau type de compromis pour l'exploration contextuelle lorsque le gain n'est pas forcement un fonction linéaire du contexte.

Pour T itérations, K bras, et des vecteurs contextuels à d dimensions, le regret de notre algorithme est borné par O($\log(KT)$) contrairement au O($\sqrt{T * N_f * \ln^3(K * T * \ln T/\delta)}$) de `LinUCB` avec une constante δ. En pratique, la récompense n'est pas toujours une fonction linéaire du contexte. Dans certains cas, le regret sera linéairement croissant. Cependant, en utilisant un modèle pour chaque sous-groupe, nous obtenons un regret cumulé global inférieur à `LinUCB` et `UCB`.

À partir d'observations issues d'un seul bras sur un jeu d'apprentissage, il est possible d'identifier des sous-groupes plus homogènes que la population globale. À court terme, il nous semble intéressant de voir si des résultats de meilleure qualité peuvent être obtenus avec différents types d'arbres.

Enfin, d'un point de vue applicatif, pour montrer son efficacité, nous prévoyons d'appliquer notre méthode à des cas réels de tests *A/B* appliqués au webmarketing.

Références

Agrawal, R., D. Teneketzis, et V. Anantharam (1989). Asymptotically efficient adaptive allocation schemes for controlled markov chains : finite parameter space. *IEEE Transactions on Automatic Control 34*(12), 1249–1259.

Auer, P., N. Cesa-Bianchi, et P. Fischer (2002a). Finite-time analysis of the multiarmed bandit problem. *Machine Learning 47*(2), 235–256.

Auer, P., N. Cesa-Bianchi, Y. Freund, et R. E. Schapire (2002b). The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing 32*(1), 48–77.

Breiman, L., J. H. Friedman, R. A. Olshen, et C. J. Stone (1984). *Classification and Regression Trees*. Monterey, CA : Wadsworth and Brooks.

Chu, W., L. Li, L. Reyzin, et R. Schapire (2011). Contextual bandits with linear payoff functions. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pp. 208–214.

Hothorn, T., K. Hornik, et A. Zeileis (2006). Unbiased recursive partitioning : A conditional inference framework. *Journal of computational and graphical statistics 15*(3), 651–674.

Kotsiantis, S. B. (2007). Supervised machine learning : A review of classification techniques. In *Proceedings of the 2007 Conference on Emerging Artificial Intelligence Applications in Computer Engineering*, pp. 3–24.

Maillard, O.-A. et S. Mannor (2014). Latent bandits. In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ICML'14, pp. I–136–I–144. JMLR.org.

Salzberg, S. L. (1994). C4.5 : Programs for machine learning. *Machine Learning 16*(3), 235–240.

Strasser, H. et C. Weber (1999). The asymptotic theory of permutation statistics. *Mathematical Methods of Statistics 8*, 220–250.

Summary

In this work we devise a principled approach which mixes the contextual bandit framework with the learning of a stratification procedure. The proposed algorithm is able to balance contextual exploration and exploitation more efficiently than state-of-the-art bandit algorithms for finite time at cost of a controlled probability for a linear regret. Finally, the learned structure is easily interpretable by a human.