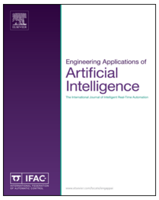
Engineering Applications of Artificial Intelligence 46 (2015) 82–92

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai



A priori-knowledge/actor-critic reinforcement learning architecture for computing the mean–variance customer portfolio: The case of bank marketing campaigns

Emma M. Sánchez [a], Julio B. Clempner [b,*], Alexander S. Poznyak [a]

[a] Center for Research and Advanced Studies, National Polytechnic Institute, Av. IPN 2508, Col. San Pedro Zacatenco, 07360 Mexico City, Mexico
[b] Center for Economics, Management and Social Research, National Polytechnic Institute, Lauro Aguirre 120, col. Agricultura, Del. Miguel Hidalgo, 11360, Mexico City, Mexico

ARTICLE INFO

Article history:
Received 10 December 2014
Received in revised form
27 August 2015
Accepted 31 August 2015
Available online 27 September 2015

Keywords:
Reinforcement learning
Preprocessing
Actor-critic
Mean–variance customer portfolio
Markov chains

ABSTRACT

In this paper we propose a novel recurrent reinforcement learning approach for controllable Markov chains that adjusts its policies according to a preprocessing and an actor-critic architecture. The preprocessing is proposed when learning a new task is needed from reinforcement based on a priori knowledge, in order to decrease computation time and not explore and not learn everything from scratch. The actor-critic architecture is based on an iterated quadratic/Lagrange programming maximization algorithm for computing the optimal strategies of the mean–variance customer portfolio. This process can be viewed as a specific form of asynchronous value iteration with optimized computational properties. The use of only the value-maximizing action at each state is unlikely in practice. Then, a specific selection of policies is used to ensure convergence. The reinforcement model proposed predicts a learning process that takes the risk of the customer portfolio into account. The resulting policies dynamically optimize the customer portfolio. We propose to apply three different learning rules, based on the transition matrices, the utilities and the costs, to estimate the objective function for the current policies. In particular, the learning rule related to estimate the real costs imposes restrictions over the formulation of the portfolio: costs cannot be underestimated or overestimated. The learning rules allow the process to make use of past experiences and decide on future actions to take in or around a given state of the Markov chain. We provide implementation details of the learning process and the complete algorithm. In addition, we illustrate our approach with a bank marketing application example for showing the viability of the model for solving realistic problems.

1. Introduction

1.1. Marketing campaigns

A company's marketing advantage lies in the capacity to predict the future buying behavior of the customers (before their rivals can), not just to respond to customers needs. Sectors such as industries, financial & insurance, telecommunications, energy, and health care realized that developing a relationship with the customers is a crucial factor in staying competitive. For achieving this goal companies need to allocate resources to build long-term relationships with their customers (Johnson and Selnes, 2004; Thomas et al., 2004). In this sense, it is relatively simple to define campaigns, adjust the campaign parameters, calculate the expenses, and assess the revenues of the campaigns. However, to assess the efficiency and effectiveness of the proper allocation of the marketing budget is not a trivial topic. A major challenge faced by marketing planners is to develop optimal campaigns for attracting customers and engaging them to patronize the products in the long run. The investment on the campaign and the periodicity are assessed based on lifetime profitability of the customer.

In response to the competitive pressures enforced by the customer demands and the constant changes on the conditions of the market, many companies are re-thinking the way they plan marketing campaigns and select the best marketing initiatives. Due to the fact that marketing is the " art" of keeping and attracting customers (Lilien et al., 1992), companies are looking for predictive models that can evolve and adapt to the changing marketing strategies in order to efficiently allocate marketing resources and to maximize the financial value generated by marketing investments.

Markov models are employed to predict the behavior of the customers (Ascarza and Hardie, 2013; Cao et al., 2012; Clempner and Poznyak, 2014; Ho et al., 2004; Kumar et al., 2011; Labbi and

[*] Corresponding author.
E-mail addresses: esanchez@ctrl.cinvestav.mx (E.M. Sánchez), julio@clempner.name (J.B. Clempner), apoznyak@ctrl.cinvestav.mx (A.S. Poznyak).

http://dx.doi.org/10.1016/j.engappai.2015.08.011
0952-1976/© 2015 Elsevier Ltd. All rights reserved.

Berrospi, 2007; Netzer et al., 2008; Pfeifer and Carraway, 2000). The segments of customers result from applying a segmentation process based on clusters techniques using different variables' criteria. The segmentation process is also used repeatedly to discretize the customer space into a finite number of states. States are refined up until a transition probability between any pair of states is possible to be established. The actions associated to the states correspond with the action of a marketing campaign. Strategies dictate how the customer in an interaction makes his decisions (choose an action). As a result of choosing a strategy the customer generates a utility (i.e., discount, points, gifts, and cash). As a result, the model can predict the behavior of the customer and the financial impact of a given strategy or the combination of several (optimal) marketing policies.

We use a reinforcement learning (RL) approach to solve the marketing problem seeing as previous works have confirmed its efficiency and effectiveness in this area (Oliveira, 2014). Abe et al. (2002) propose and evaluate a progression of reinforcement learning methods, ranging from the "direct" or "batch" methods to "indirect" or "simulation based" methods, and those that we call "semidirect" methods that fall between them. Pednault et al. (2002) present a method that attempts to learn decision rules that optimize a sequence of cost-sensitive decisions so as to maximize the total benefits accrued over time. Sun (2003) introduces a RL model that allows firms to design optimal dynamic mailing policies using their own business data by studying how an action in time influences actions in following times. Abe et al. (2004) suggest to optimize cross channel marketing (lack of explicit linking between the marketing actions taken in one channel and the customer responses obtained in another) providing a solution for this problem based on old and new techniques in RL. Gomez-Perez et al. (2009) present two approaches for finding an optimal policy for a marketing campaign: (a) the first approach is based on the self-organizing map, which is used to aggregate states, and (b) the second approach uses a multilayer perceptron to carry out a regression of the action-value function.

1.2. *Reinforcement learning*

Reinforcement learning (Kaelbling et al., 1996) is a computational technique developed to learn by receiving reinforcement signals (reward or punishment) from interaction with the environment (Ayesh, 2004; Gadanho, 1999). A reinforcement learning agent learns from its own experience interacting alone with the environment (Ribeiro, 2002). The environment is everything outside the agent. The states of a RL system describe the characteristics of the environment. The dynamics of the agent is as follows: it senses the environment and learn optimal policies by taking actions trying to maximize the reward or minimize the punishment (Ayesh, 2004; Gadanho, 1999; Sutton and Barto, 1998). One of the classical approaches of reinforcement learning is temporal difference which does not require a model of the environment and is based on step-by-step, incremental computation. RL algorithms take advantage from methods that decrease their computation time in the case of large state spaces (Sutton and Barto, 1998).

An agent will eventually converge to an optimal strategy and utility (Sutton and Barto, 1998). One of the theoretical conditions for convergence of reinforcement learning implies that each state–action pair must be visited infinite times. However, the use of the optimal strategy to maximize the value of an action at each state is unlikely to result in thorough exploration in practice. The basic problem is that the agent will know the reward until it takes an action. For this reason, an important challenge in reinforcement learning is the trade off between exploration and exploitation. On one hand, the exploration policies are used to ensure that each action is tried at each state sufficiently often. But on the other hand, by acting randomly an agent is assured of sampling each action at each state infinitely often in the limit. Then, the agent has to exploit the already known optimal actions to obtain rewards, and it has also explored the actions not taken to improve the action choices in the future.

There are several solutions presented in the literature that consider conditions for convergence of reinforcement learning where the agent faces a very large state or action space (Kaelbling et al., 1996). Ribeiro (1998) proposes methods for embedding *a priori knowledge* in a reinforcement learning technique using the Q-learning algorithm in such a way that both the mathematical structure of the basic learning algorithm and the capacity to generalize experience across the state–action space are kept. Considering a priori knowledge Maclin et al. (2005) present the Preference Knowledge-Based Kernel Regression algorithm that employs human advice as a policy based on if-then rules. Mahadevan and Kaelbling (1996) introduce the *hierarchical model* approach that involves increasing the speed of learning by decomposing the task into a collection of simpler subtasks. For hierarchical reinforcement learning, Dietterich (2000) proposes a method based on decomposing the target Markov decision process (MDP) into a hierarchy of smaller MDPs and decomposing the value function of the target MDP into an additive combination of the value functions of the smaller MDPs. Goel (2003) describes a method of sub-goal discovery based on learned policy. Kaelbling (1993) suggests the hierarchical distance to goal learning by using decomposition of the state space. Shapiro et al. (2001) bring together the hierarchical RL with background knowledge. McGovern et al. (1997) for *increasing the learning speed* propose macro-actions, a set of actions executed in sequences, that are closed-loop policies with termination conditions that can be chosen at the same level as primitive actions. Sutton et al. (1998) accelerate reinforcement learning process presenting options, that extend the notion of macro-action (McGovern et al., 1997), consisting of three components: a policy, a termination condition, and an initiation set. Sutton et al. (1999) introduce intra-option, learning methods that learn about an option from small fragments of experience consistent with that option (Sutton et al., 1998), even if the option is not executed. Tizhoosh et al. (2005) propose that the demands and needs of a user can be learned *offline* with the purpose to minimize the real time. Potapov and Ali (2003) show that appropriate choices of certain parameters may influence drastically the convergence rate of the RL techniques. Berenji (1994) introduces *fuzzy reinforcement learning* (FRL) developing a new algorithm called fuzzy Q-learning (or FQ-Learning) used for decision processes in which the goals and/or the constraints, but not necessarily the system under control, are fuzzy in nature. Berenji (1996) extends his previous work introducing GARIC-Q, a method for incremental dynamic programming using a society of intelligent agents which are controlled: (1) at the top level by fuzzy Q-learning, and (2) at the local level each agent learns and operates based on GARIC. Driessens and Dzeroski (2004) introduce *relational reinforcement learning,* that makes Q-learning feasible in structural domains by incorporating a relational learner into Q-learning, presenting a solution based on the use of "reasonable policies" to provide guidance. Morales (2004) shows how a relational representation can be used to produce powerful abstractions which can significantly reduced the search space, and that learning over this abstracted space can help us to re-use previously learned policies on new problems. Tizhoosh (Tizhoosh, 2005; Tizhoosh et al., 2008) presents *opposition-based learning* as a scheme for machine intelligence that estimates and counter-estimates, weights and opposite weights, and actions versus counter-actions which are the foundation of this new approach. Shokri (Shokri et al., 2008; Shokri, 2011) proposes the concept of opposition for each action within reinforcement learning techniques to