CS 181 Machine Learning
Practical 4 Report, Team *la Dernière Dame M*

(Jeremiah) Zhe Liu[1], (Vivian) Wenwan Yang[2], and Jing Wen[1]

[1]Department of Biostatistics, Harvard School of Public Health
[2]Department of Computational Science and Engineering, SEAS

May 6, 2015

1 Problem Description

Set in a *Flappy Bird*-type game *Swingy Monkey*, our current learning task is to estimate the optimal policy function $\pi : \mathcal{S} \to \mathcal{A}$ such that the expectation of reward function $R : \mathcal{S} \times \mathcal{A} \to \mathcal{R}$ is maximized, i.e. if define a stochastic process of game state $\{s_t\}$ with unknown transition probability $P(s_{t+1}|s_1, \dots, s_t, a_1, \dots, a_t)$, we aim to identify a π^* such that

$$\pi^* = \arg\max_{\pi} E\Big(\sum_{s \in \mathbf{p}} R(s, \pi(s))\Big|\mathbf{p}\Big)$$

where $\mathbf{p}$ a sample path of $\{S_t\}$.

In current setting, the state and action spaces are defined as:

$$\begin{aligned}
\mathcal{S} &= \begin{bmatrix}\mathsf{Tree}_{\text{dist}} & \mathsf{Tree}_{\text{top}} & \mathsf{Tree}_{\text{bot}} & \mathsf{Monkey}_{\text{vel}} & \mathsf{Monkey}_{\text{top}} & \mathsf{Monkey}_{\text{bot}}\end{bmatrix} \subset \mathbb{R}^6 \\
\mathcal{A} &= \begin{bmatrix}\mathsf{NoJump} & \mathsf{Jump}\end{bmatrix}
\end{aligned}$$

Note theta $\big[\mathsf{Monkey}_{\text{top}}, \mathsf{Monkey}_{\text{bot}}, \mathsf{Tree}_{\text{top}}, \mathsf{Tree}_{\text{bot}}\big]$ are in fact bounded by screen size (600 pxls).

The reward function can be partially described as:

$$R : \begin{bmatrix}\texttt{pass\_tree} \\ \texttt{hit\_trunk} \\ \texttt{hit\_edge} \\ \texttt{otherwise}\end{bmatrix} \to \begin{bmatrix}1 \\ -5 \\ -10 \\ 0\end{bmatrix}$$

where $\big[\texttt{pass\_tree}, \texttt{hit\_trunk}, \texttt{hit\_edge}\big]$ are unknown boolean functions of $s \in \mathcal{S}$.

2 Method

2.1 Rationale on Model Choice

In the previous section we identified below characteristics of the task at hand:

(1) Available Information:

   (a) Known $\mathcal{S}$, $\mathcal{A}$ spaces.
   (b) Unknown transition probability $P(s_{t+1}|\{s_i\}_{i=0}^t, \{a_i\}_{i=1}^t)$ and unknown reward function $R : \mathcal{S} \times \mathcal{A} \to \mathcal{R}$

(2) $|\mathcal{A}| = 2$, while $\mathcal{S} \subset \mathbb{R}^6$ is continous with $|\mathcal{S}| = \infty$.

(3) Outcome metric: $E\Big(\sum_{s \in \mathbf{p}} R(s, \pi(s))\Big|\mathbf{p}\Big)$ the expected number of total reward in each play.

If we are willing to assume Markovian property for the process $\{s_t\}$, i.e. $P(s_{t+1}|\{s_i\}_{i=0}^t, \{a_i\}_{i=1}^t) = P(s_{t+1}|s_t, a_t)$. The availabile information listed in (1) put us into a Reinforcement Learning setting.

2.1.1 Dimension selection and Discretization

markov assumption

2.1.2 Exploration/Exploitation Parameters

Learning rate
ϵ-greedy

2.2 Performance Evaluation

3 Result

3.1 State Exploration

3.2 Convergence Behavior

4 Discussion & Possible Directions

Reference

1. Ricci F, Rokach L, Shapira B et al. (2010) **Recommender Systems Handbook**. *Springer*.

2. Koren Y, Bell R, Volinsky C. (2009) **Matrix factorization techniques for recommender systems**. *IEEE Computer* Aug 2009, 42-49.

3. Srebro N, Jaakkola T.(2003) **Weighted low-rank approximations**. *Proceedings of the Twentieth International Conference* 720727.

4. R Salakhutdinov, A Mnih. (2008) **Probabilistic Matrix Factorization**. *Advances in Neural Information Processing Systems* Vol. 20

5. Koren, Y. (2008) **Factorization Meets the Neighborhood: a Multifaceted Collaborative Filtering Model**, *Proc. 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.