With a Little Help from the Computer: Hybrid Human-Machine Systems on Bandit Problems

Bryan R. Gibson, Kwang-Sung Jun, Xiaojin Zhu

University of Wisconsin-Madison, USA

The Question

Is it possible for a machine-human hybrid system to perform better on a Multi-Arm Bandit task that a human would alone?

Multi-Armed Bandit

Deciding which of two slot machines (arms) to play when you don't know the distributions they are drawing from is an example of a Multi-Arm Bandit (MAB) problem. If the objective is win as much as possible, people often make suboptimal decisions in this task.

What if a machine learner (ML) was able to help? Would the collaboration together perform better than the human alone?

Problem Setup

The Setting:

A person performs a set of pulls on 2 arms $\{A, B\}$ with unknown distributions. On each iteration, ML calculates the optimal pull and provides a suggestion. The person performs a pull and both the person and ML see the reward. This process repeats many times.

The Knowns:

j_i : Arm pulled on pull i, $i = 1, \ldots, n$
n_j : Number of pulls made so far on arm j
n : Number of pulls made so far total $(n_A + n_B)$
r_i : Reward received at pull i, $r \in [1, 100]$

The Unknowns:

P_A : Distribution of A with mean μ_A
P_B : Distribution of B with mean μ_B
μ_* : True best expected value

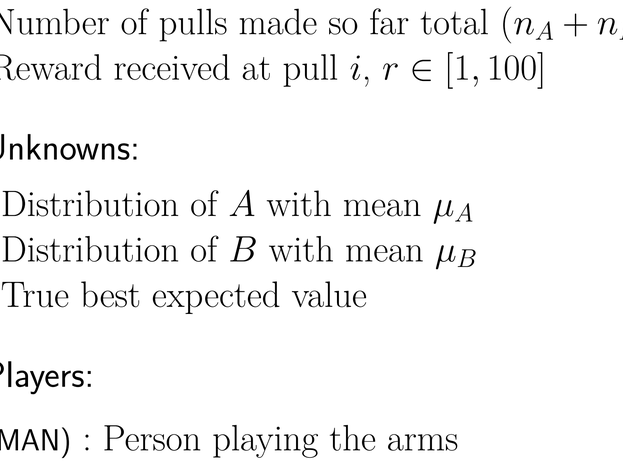
The Players:

(HUMAN) : Person playing the arms
(MACHINE) : ML learning the optimal arm

The Constraints:

- Only (HUMAN) can pull the arms
- Both (HUMAN) and (MACHINE) can see the reward

The Goal: To maximize the reward received



An iteration consists of an actual pull, reward presentation, calculation of the next optimal pull and conversion into a suggestion.

Machine Learner

The UCB1 algorithm [1] solves this optimally, where optimal is defined as choosing the arm corresponding to $\max\{\mu_A, \mu_B\}$.

$$\text{Per-Trial Regret} : \mu_* - \frac{1}{n}\sum_{i=1}^{n} r_i$$

At each pull, UCB1 finds the arm with the highest expected upper bound on its reward.

$$\text{UCB1} : \arg\max_j \bar{r}_j + \sqrt{\frac{2\ln n}{n_j}}$$

Instead of using UCB1 we use UCB1-Tuned (UCB1t) which was found to work better empirically. UCB1t uses a different method to calculate the upper bound.

$$\text{UCB1t} : \arg\max_j \bar{r}_j + \sqrt{\frac{\ln n}{n_j}\min\left(\frac{1}{4}, V_j(n_j)\right)}$$

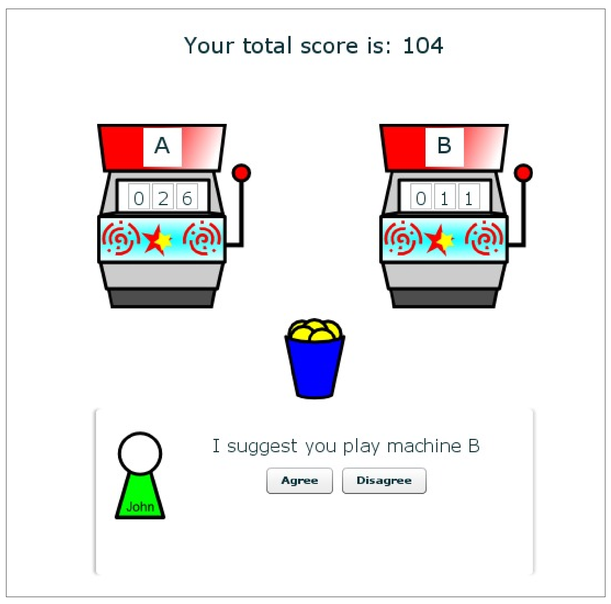
$$V_j(n_j) = \left(\frac{1}{n_j}\sum_{\tau=1}^{n_j} r_{j,\tau}^2\right) - \bar{r}_{j,n_j} + \sqrt{\frac{2\ln i}{s}}$$

The Experiment

112 participants played two arms for 150 pulls.

Participants were told:

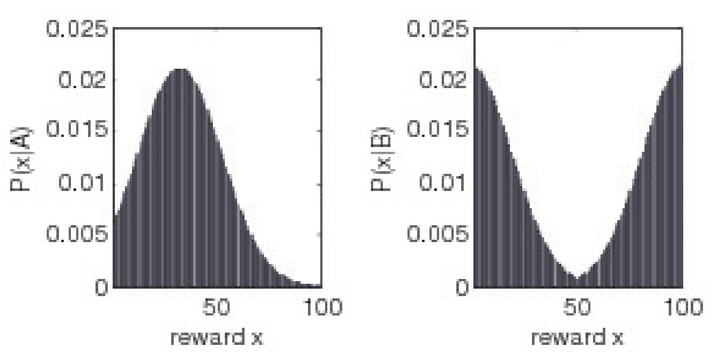
- The range of the reward
- One arm might give better results than the other
- They might receive suggestions
- That their goal was to maximize their reward



The task interface showing a simple suggestion (**S**)

Arm distributions were designed to confuse the human learner into choosing the suboptimal arm.

A $(\mu_A = 35)$: consistently mediocre reward
B $(\mu_B = 50)$: high and low rewards (optimal)



The distributions P_A $(\mu_A = 35)$ and P_B $(mu_B = 5)$

Conditions

H : No suggestions given (28 participants)

S : Simple suggestion (27 participants)
"I suggest you play machine A"

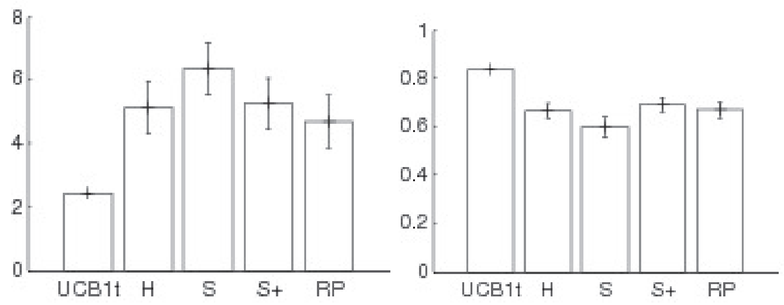
S+ : Authoritative suggestions (28 participants)
"You have played machine A (B respectively) 3 (5) times, the sample mean is 45 (72), while the upper confidence bound of the true mean can be as high as 87 (100). I suggest you play machine B."

RP : Reverse psychology (29 participants)
The probability that the participant will agree with the suggestion is calculated using the agreement history: $p(g_i|g_1, \ldots, g_{i-1})$. Instead of the full history, this is approximated using only the previous iteration: $p(g_i|g_{i-1})$. If agreement is unlikely ($< .5$) the suggestion is flipped to the other arm. Otherwise the suggestion is identical to the simple suggestion type.

Results

Results were measured using Per-Trial Regret and Best Arm Percentage, the percentage of iterations where the optimal arm was pulled.

As a comparison, UCB1t was run on its own for 5000 sessions, each session 29 trials of 150 iterations.



Per-Trial Regret | Best Arm Percentage

Differences were not found to be statistically significant, however the trend is surprising in that suggestions do not seem to have helped and, in fact, have hurt in **S**.

Hybrid system performance was found to be at best equivalent to human performance without suggestions.

Future Work

- Expand suggestions available
- Learn the best suggestion type for a participant
- Create a more complex model of participant agreement taking into account the full history
- Rerun using Gittins' Dynamic Allocation Process as the ML algorithm (direct reward maximization)

References

[1] P. Auer, N. Cesa-Bianchi, and P. Fisher. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2):235-256, 2002.

Research supported in part by
NSF IIS-0953219 and AFOSR FA9550-09-1-0313

NIPS 2010 Vancouver, BC {bgibson,deltakam,jerryzhu}@cs.wisc.edu