Clinical Online Recommendation with Subgroup Rank Feedback

Yanan Sui
California Institute of Technology
1200 E California Blvd
Pasadena, CA, USA
ysui@caltech.edu

Joel Burdick
California Institute of Technology
1200 E California Blvd
Pasadena, CA, USA
jwb@robotics.caltech.edu

ABSTRACT

Many real applications in experimental design need to make decisions online. Each decision leads to a stochastic reward with initially unknown distribution. New decisions are made based on the observations of previous rewards. To maximize the total reward, one needs to solve the tradeoff between exploring different strategies and exploiting currently optimal strategies. This kind of tradeoff problems can be formalized as Multi-armed bandit problem. We recommend strategies in series and generate new recommendations based on noisy rewards of previous strategies. When the reward for a strategy is difficult to quantify, classical bandit algorithms are no longer optimal. This paper, studies the Multi-armed bandit problem with feedback given as a stochastic rank list instead of quantified reward values. We propose an algorithm for this new problem and show its optimality. A real application of this algorithm on clinical treatment is helping paralyzed patient to regain the ability to stand on their own feet.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous

Keywords

Clinical Recommendation, Exploration-Exploitation Tradeoff, Bandit Problem, Rank-Comparison

1. INTRODUCTION AND MOTIVATION

Our problem is motivated by clinical research which aims to recover motor function after severe spinal cord injury (SCI). Previous research [4] has shown that electrical stimulation applied to the spinal cord via electrodes arrays implanted in the epidural space over the lumbosacral area enables paralyzed patients to achieve full weight-bearing standing, improvements in stepping, and partial recovery of lost autonomic functions. The electrical stimulation must be coupled with physical therapy to realize the best outcome.

RecSys'14, October 6–10, 2014, Foster City, Silicon Valley, CA, USA.

ACM http://dx.doi.org/10.1145/2645710.2645773.

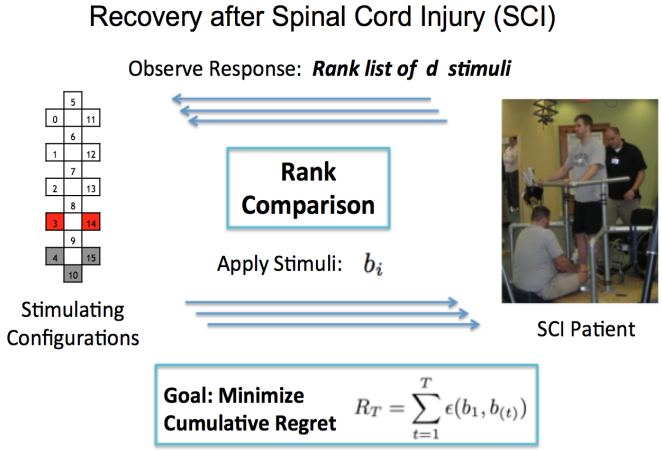

Figure 1: Clinical Treatment of Spinal Cord Injury

Stimulation consists of electrical pulse trains applied to selected electrodes. However, the optimal stimulus pattern (the choice of active electrodes and their polarity, the pulse amplitude and width, and the pulse train frequency) varies significantly across patients. And even for the same patient, the outcome of the same stimulus varies from trial to trial, and the optimal stimulation varies over time due to spinal cord plasticity. Hence, clinicians must determine the optimal stimulus for each patient under noisy and mildly non-stationary conditions. Currently, the search for the optimal stimulating parameters is a laborious and somewhat ad-hoc approach which consumes valuable clinician and patient time, and does not currently guarantee an optimal outcome.

Figure 1 shows the clinical treatment procedure for *stand-training*. During a treatment/optimization session, a new stimulus is recommended by our algorithm. The patient then attempts to stand using the given stimulus, and the observing clinicians then rank the patient's resulting performance. Using this noisy ranking as feedback, the algorithm continues to explore for the optimal stimulus while also exploiting currently good ones. The algorithm must spend significant time dwelling on good performing stimuli in order to provide the patient with a good therapeutic experience. Since clinical training has a fixed time horizon, we must also maximize total performance during the limited period within which we can search for the optimal solution.

This paper develops an algorithm to recommend optimal stimuli based on the general setting of multi-armed bandit problem. The classical bandit problem trades off between

exploration and multi-armed bandit problem exploitation among a number of different arms, each having a quantifiable, but stochastic, reward with initially unknown distribution. The goal of a bandit algorithm is to maximize the total reward. Since its introduction by Robbins [5], bandit problems have been widely studied in many situations [2]. Many efficient algorithms have been developed based on the work of Auer et al., [1].

However, for our clinical problem, the patient's motor response to stimulation is hard to quantify. Neither video motion capture nor electromyograhic (EMG) recordings of muscle activity can yet provide a consistent and satisfactory measure of motor skill under stimulation. A good standing performance might map to numerous combinations of muscle activities, and it is not a stationary process. While the patient's performance under a specific stimulus is hard to quantify, it can be compared to others. In the clinical setting, we can obtain the ranking of a *group* of stimuli which are performed within the short time period of one training session. The *dueling bandit problem* [7] formalizes online learning problems with preference feedback instead of absolute rewards, and hence it can be used for problems with unquantifiable reward. The algorithm we propose in this paper is a variant of the dueling bandit problem which is dictated by the clinical demands of our application.

At the start of the optimization process, we have little information about the best stimulus for the patient, but we have often have a pool of possibly useful stimuli. Our approach is based on the idea of successively removing suboptimal arms [3] while keeping the optimal one(s) in the sample space. By setting proper confidence intervals, we can reach the optimal reward within the time horizon.

2. PROBLEM SETUP

The classical dueling bandit problem receives feedback in the form of a comparison between a pair of bandits in each test. When the size of the decision set, K, is large, it is unavoidable to carry out a very large number of tests before the algorithm converges to its optimal solution. In some applications like our clinical example, each test is expensive and time consuming. The number of tests - time horizon of an algorithm - is often predetermined by clinical conditions. It is infeasible to apply the dueling bandit algorithm directly.

However, our training and optimization procedure allows for patients to not only compare successive stimulations, but to also rank the performances for a modest-sized group of stimulations (the number which can be tested in one clinical session before the patient fatigues). Thus, feedback consists of a ranked list of at most d ($d < K$) chosen arms. More precisely, the feedback for each test consists of a combined scoring of 4 different standing criteria by the observing clinicians, and the combined score is used to rank the tests within one session. We show below that this feature helps us to reduce the total number of tests significantly, while also dovetailing well with current clinical practice.

Our procedure can be described as follows. There are K arms $\{b_1, \cdots, b_K\}$, and a total number of T tests to be performed. Each test physically corresponds to a ∼90-second stimulation period with a specific stimulus (arm) chosen from the K arms. T is determined before we run the algorithm, and is generally assumed to be an integer multiple of d: $T = d*G$, where G is the number of ranking sessions, with each session producing a noisy ranked list of d arms.

Algorithm 1 Rank-Comparisons

Input: $\{b_1, ..., b_K\}$, d, G // *Total tests* $T = d \cdot G$
Input: $c_\delta(n) = \sqrt{(1/n)log(1/\delta)}$
Run: [Parameters-Initialization]
Run: [Active-Elimination]
return b^* // *Optimal arm*

Algorithm 2 Parameters-Initialization

Input: $\{b_1, ..., b_K\}$, d, G
Input: $c_\delta(n) = \sqrt{(1/n)log(1/\delta)}$
$W_1 \leftarrow \{b_1, ..., b_K\}$ // *set of active arms*
$\ell \leftarrow 1$ // *rounds*
$\forall b \in W_\ell,\ n_b \leftarrow 0$ // *comparisons*
$\forall b \in W_\ell,\ w_b \leftarrow 0$ // *priorities*
$\forall b \in W_\ell,\ \hat{P}_b \equiv w_b/n_b$, or 1/2 if $n_b = 0$
$n^* \equiv min_{b \in W_\ell} n_b$
$c^* \equiv c_\delta(n^*)$, or 1 if $n^* = 0$ // *confidence radius*
$g \leftarrow 0$ // *total number of ranks*
$T \leftarrow d \cdot G$
return all new parameters

We follow the the original notation of the dueling bandit problem [7]. For two arms b_i and b_j, where $i, j \in \{1, \cdots, K\}$, we write the comparison factor as:

$$\epsilon(b_i, b_j) = P(b_i \succ b_j) - 1/2$$

where $P(b_i \succ b_j)$ is the probability that b_i dominates b_j and $\epsilon(b_i, b_j) \in [-1/2, 1/2]$ represents the priority between b_i and b_j. We define $b_i \succ b_j \Leftrightarrow \epsilon(b_i, b_j) > 0$. We use the notation $\epsilon_{i,j} \equiv \epsilon(b_i, b_j)$ for convenience. Note that $\epsilon(b_i, b_j) = -\epsilon(b_j, b_i)$ and $epsilon(b_i, b_i) = 0$. We assume the distribution of reward for each arm is stationary so that all comparison factors converge in [-1/2,1/2]. We also assume *w.l.o.g.* that the bandits are indexed in preferential order $b_1 \succ b_2 \succ \cdots \succ b_K$ so that there is one preferred arm.

The total reward is defined in terms of regret as in the classical bandit problem setting. In the online setting, let $b_{(t)}$ be the arm chosen at test t. We define total regret as follows:

$$R_T = \sum_{t=1}^{T} \epsilon(b_1, b_{(t)})$$

The total regret $R_T = 0$ if we constantly choose $b_{(t)} = b_1$ during the experiment. $R_T = \Theta(T)$ is linear *w.r.t.* T if we constantly choose $b_{(t)} \in \{b_1, \cdots, b_K\}$.

We also inherit two important properties of the comparison factors from the original dueling bandit problem:

Strong Stochastic Transitivity. For any triplet of arms $b_i \succ b_j \succ b_k$, we assume $\epsilon_{i,k} \geq \max\{\epsilon_{i,j}, \epsilon_{j,k}\}$.

Stochastic Triangle Inequality. For any triplet of arms $b_i \succ b_j \succ b_k$, we assume $\epsilon_{i,k} \leq \epsilon_{i,j} + \epsilon_{j,k}$. This can be viewed as a diminishing returns property.

An optimal method is proposed for our problem which has a finite-time regret bound of order $O(\frac{K}{d} logT)$ where T is the time horizon.

3. ALGORITHM

Our *Rank-Comparison* algorithm (Algorithm 1), which is a modified version of "Beat-the-Mean" [8], is based on the idea of successively removing suboptimal arms while keeping

Algorithm 3 Active-Elimination

1: **Input:** $\{b_1, ..., b_K\}$, d, G
2: **Input:** parameters generated in [Parameters-Initialization]
3: **while** $|W_\ell| > 1$ and $g \leq G$ **do**
4: **if** $|W_\ell| \geq d$ **then**
5: select $b'_1, ..., b'_d \in W_\ell$ at random with no repeats
6: **else**
7: $r \leftarrow d\%|W_\ell|$
8: $p \leftarrow (d-r)/|W_\ell|$
9: select $b'_1, ..., b'_r \in W_\ell$ at random with no repeats. In addition, select each arm in W_ℓ p times
10: **end if**
11: test selected arms and get rank of the selection
12: **for** all commutable pairs (b'_i, b'_j) in the selection **do**
13: if $b'_i \succ b'_j$, $w_{b'_i} \leftarrow w_{b'_i} + 1$
14: $n_{b'_i} \leftarrow n_{b'_i} + 1$
15: **if** $\min_{b' \in W_\ell} \hat{P}_{b'} + c^* \leq \max_{b \in W_\ell} \hat{P}_b - c^*$ **then**
16: $b' \leftarrow \arg\min_{b \in W_\ell} \hat{P}_b$
17: $\forall b \in W_\ell$, delete comparisons with b' from w_b, n_b
18: $W_{\ell+1} \leftarrow W_\ell \backslash \{b'\}$ // *update working set*
19: $\ell \leftarrow \ell + 1$ // *new round*
20: **end if**
21: **end for**
22: **end while**
23: **return** $b^* = \arg\max_{b \in W_\ell} \hat{P}_b$

the optimal one(s) in the sample space. The inputs to *Rank-Comparison* are the K arms, the largest group size d, and total number of groups G: $T = d \cdot G$.

Parameters-Initialization (Algorithm 2) defines the set of active arms W_ℓ, whose size shrinks as more tests are completed. For each arm b, let n_b be the total number of comparisons between b and other arms, and let w_b be the total number of wins against all other arms. Let $\hat{P}_b$ be the empirical average of $P(b \succ b')$ for all b' in W_ℓ, and let $\hat{P}_{b,n}$ be the value of $\hat{P}_b$ after n comparisons between arm b and any other arms. Set the confidence interval of $P(b \succ b')$ as:

$$\hat{C}_{b,n} = (\hat{P}_{b,n} - c_\delta(n), \hat{P}_{b,n} + c_\delta(n))$$

where $c_\delta(n) = \sqrt{(1/n)log(1/\delta)}$, and δ is the confidence that $P(b \succ b')$ lies in $\hat{C}_{b,n}$. The function $c_\delta(n)$ decreases as the number of comparisons n increases. By properly setting parameter δ, the optimal reward can be reached within the fixed time horizon.

Active-Elimination (Algorithm 3) is the key part of *Rank-Comparison*. For each group of tests, d arms are randomly chosen from W_ℓ with no repeats when $d < |W_\ell|$. Otherwise, we pick each arm equally and pick the rest arms randomly according to lines 7-9 in Algorithm 3. The randomized selection method provides low-variance total regret. Each group of tests results in a ranking of d arms, which can be regarded as $d(d-1)/2$ comparisons among the d arms. For each arm b, the values of w_b, n_b and $\hat{P}_b$ are updated, as is the corresponding confidence radius c^*. For any pair of arms b and b', one dominates the other if their confidence intervals do not overlap, and the less superior arm is eliminated from W_ℓ. The algorithm runs until the time horizon $T = d \cdot G$ is reached, or only one active arm remains.

4. THEORETICAL RESULTS

The patients can rank performances d stimuli at most. For fixed time horizon T, choose the size of groups equals the maximum group size d. It will maximize the number of total comparisons extracted from the ranks, which is $d(d-1)/2$.

Let $\epsilon = \epsilon_{1,2}$ to be the comparison factor between the best and second best arms. Obviously, we have $\epsilon \leq \epsilon_{1,j}$ for all j. The upper bound of the expected total regret for *Rank-Comparison* is given in the theorem below.

THEOREM 1. *The expected regret generated by running Algorithm 1 is bounded from above by* $O(\frac{K}{\epsilon \cdot d} logT)$.

As compared to the classical dueling bandit regret bound of $O(\frac{K}{\epsilon} logT)$, *Rank-Comparison* has an extra divisor factor of d. This tighter bound is realized because for each group of d tests, order $O(d^2)$ comparisons are extracted from the ranking test. Recall that $R_T = 0$ if the optimal arm $b_{(t)} = b_1$ is constantly chosen, and $R_T = \Theta(T)$ is linear *w.r.t.* T if we constantly choose $b_{(t)} \in \{b_1, \cdots, b_K\}$. The factor $O(\frac{K}{\epsilon \cdot d} logT)$ lies in the region between 0 and $\Theta(T)$. As T increases, $O(\frac{K}{\epsilon \cdot d} logT)$ is significantly less than $\Theta(T)$.

By extending Theorem 4 of [7], we can form a lower bound on regret in expectation, as stated in Theorem 2, for any algorithm which solves the rank comparison problem. Which means no algorithm can achieve lower regret than *Rank-Comparison* in expectation.

THEOREM 2. *Any algorithm for the rank comparison problem has a regret bounded from below by* $\Omega(\frac{K}{\epsilon \cdot d} logT)$.

Notice that Theorem 2 lower bounds total regret on the same order as the upper bound in Theorem 1. So we have $\Omega(\frac{K}{\epsilon \cdot d} logT) = E[R_T] = O(\frac{K}{\epsilon \cdot d} logT)$, from which we can conclude that total regret is order $\Theta(\frac{K}{\epsilon \cdot d} logT)$ for *Rank-Comparison*.

Theorems 1 and 2, whose detailed proofs can be found in the supplementary [6], show that our algorithm is optimal in terms of the expected total regret.

Unlike the classical multi-armed bandit problem, which only focuses on expected total regret, many applications must constrain the regret's variation. In our context, if a stimulus optimization algorithm provides good results in the majority of patients, but bad results in a few, the variation is large. Such an algorithm is not practically useful, even if total regret is small. By randomizing the choice of arms within each test group, the randomized comparison strategy of *Rank-Comparison* provides low-variance regret in expectation.

5. EXPERIMENTS

We first evaluate the algorithm by simulation. The reward for each arm b_i is modeled as a Gaussian distribution with mean μ_i and standard deviation σ_i. All arms are independent with each other. Obviously, the distributions generated in this way satisfies the Strong Stochastic Transitivity and Stochastic Triangle Inequality. Then we sample the arms for each group and rank them by using the Rank-Comparison algorithm. We calculated the expected regret $r_t = R_t/t$ (instead of total regret R_t) where t is the number of tests. In the simulation, we consider the total number of arms is 10 and we can get rank list with dimension no larger than 5. The reward of each arm b_i follows a Gaussian distribution

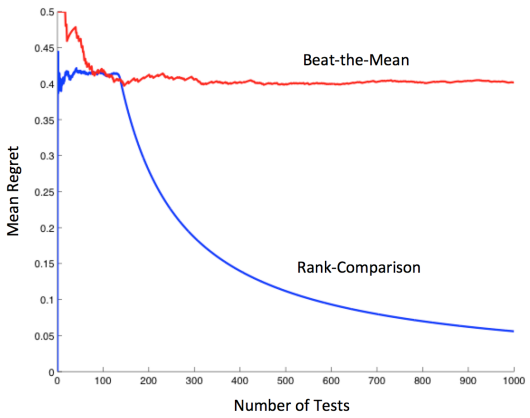

Figure 2: Mean Regret against Number of Tests

with mean $\mu_i \in [0, 1]$ and standard deviation $\sigma_i = 0.2$. Set the confidence parameter $\delta = 10^{-2}$.

Under this setting, the arms are hard to be distinguished from each other due to the large variances. Figure 2 shows the mean regret r_t vs. time t for *Rank-Comparison* (blue curve) and Beat-the-mean [8] (red curve) with fixed horizon $T = 1000$. Blue curve is the mean regret of *Rank-Comparison*, while the red curve is the mean regret of Beat-the-Mean algorithm. For both algorithms, the mean regret is high during exploration, and then drops quickly after the algorithms converge to the optimum. We can see that *Rank-Comparison* finds the optimum within 150 tests, and thereafter exploits it to reduce the mean regret. However, Beat-the-Mean did not converge to the optimum within the time horizon for the same parameter settings. We hypothesize that *Rank-Comparison* outperforms Beat-the-Mean because of the utility of finer feedback information.

We have applied *Rank-Comparison* to a SCI patient implanted with Medtronic electrode arrays (16 electrodes) driven by a Restore Advanced impulse generator. This system can apply more than 10^9 unique stimuli. Searching through the whole space of possible stimuli is neither feasible nor necessary for the clinical experiment.

For the first clinical treatment, the initial space for exploration is composed of around 20 stimuli. We have run *Rank-Comparison* for the stimuli recommendation. The algorithm has not converges to a single arm but has eliminated the majority of them. Since the number of current clinical tests is small, we have not seen the logarithmic convergence of the regret. The reason is that elimination precess is still ongoing and we are exploring more for the early experiments. We will keep running the *Rank-Comparison* algorithm on new clinical treatments.

6. DISCUSSION AND CONCLUSION

This paper proposed a *Rank-Comparison* algorithm to efficiently solve a specific bandit problem using subgroup rank feedback. This optimal strategy (Theorems 1 and 2) provides clinical recommendation which explore for optimal stimuli while exploiting high performing stimuli for SCI therapy. The main advantages of *Rank-Comparison* are:

- Fast convergence, which is a necessity for applications which are characterized by expensive explorations.
- low variance of the reward/regret (R_T), which guarantees that the approach performs uniformly on the majority of patients.

Rank-Comparison decomposes test group rankings into equally weighted comparisons. One might reasonably assume that arms far apart in rank may be more distinguishable than adjacent ones, and thus employ different confidence parameters as appropriate. This feature can reduce total regret under the same problem setting. From the clinical point of view, this method avoids the varying effect of human judgement by using robust comparisons instead of volatile quantitative values, which may be non-stationary in our application. However, the time varying characteristics of human motor performance due to fatigue in the short term, and spinal plasticity over the long term, is a real theoretical and clinical issue we must address.

Additionally, the classical bandit problem's assumption of independent arms does not hold for the spinal cord stimulation where anatomical principles and electrical properties suggest a coupling occurs. Using a measure of similarity between stimuli based on the physical properties, we can build a prior distribution on unknown arms to guide our search.

7. ACKNOWLEDGMENTS

This work was supported by the the Helmsley Foundation, the Christopher and Dana Reeve Foundation, and the National Institutes of Health (NIH).

8. REFERENCES

[1] P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.

[2] S. Bubeck and N. Cesa-Bianchi. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends in Machine Learning*, 5:1–122, 2012.

[3] E. Even-Dar, S. Mannor, and Y. Mansour. Pac bounds for multi-armed bandit and markov decision processes. In *Computational Learning Theory*, pages 255–270. Springer, 2002.

[4] S. Harkema, Y. Gerasimenko, J. Hodes, J. Burdick, C. Angeli, Y. Chen, C. Ferreira, A. Willhite, E. Rejc, R. G. Grossman, et al. Effect of epidural stimulation of the lumbosacral spinal cord on voluntary movement, standing, and assisted stepping after motor complete paraplegia: a case study. *The Lancet*, 377(9781):1938–1947, 2011.

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

[6] Y. Sui and J. Burdick. Bandit problem with subgroup rank feedback. *arXiv preprint arXiv:0971348*, 2014.

[7] Y. Yue, J. Broder, R. Kleinberg, and T. Joachims. The k-armed dueling bandits problem. *Conference on Learning Theory (COLT)*, 2009.

[8] Y. Yue and T. Joachims. Beat the mean bandit. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 241–248, 2011.