

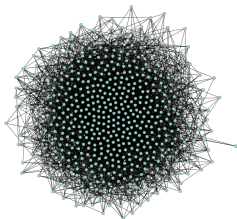
Bandits on Stochastic Blockmodel Graphs

Arthur Mensch, Michaël Weiss

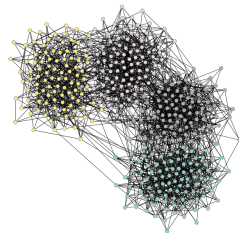
Reinforcement Learning

January 2015

Introduction



Erdos Renyi Graph



Other random graph

Problem

Multi arm bandit problem with side information to exploit

- 1 EXP3 on Erdős-Rényi Graph
- 2 EXP3 on Stochastic Block-Model Graphs
- 3 Results

Table of contents

- 1 EXP3 on Erdős-Rényi Graph
- 2 EXP3 on Stochastic Block-Model Graphs
- 3 Results

Problem definition

N arms, T steps.

- ① The environment chooses losses for every arm noted $l_{t,i}$ for the arm i at the step t .
- ② Following the algorithm we hope would minimize as much as possible the regret the player draws an arm I_t .
- ③ The player receives the loss l_{t,I_t} .
- ④ We define $(O_t)_{i \in [N]}$ as the indicative function of observed loss at step t . We have:

$$O_{t,I_t} = 1 \quad \forall i \neq I_t, O_{t,i} \sim B(r)$$

$(O_t)_{i \in [N]}$ corresponds to the value of the logic expression *i is neighbor of I_t* in the Erdős-Rényi random graph drawn at step t .

- ⑤ For all i such that $O_{t,i} = 1$ the player can observe the loss $l_{t,i}$.

Duplex tricks

Ideally,

$$\hat{l}_{t,i} = \frac{O_{t,i} l_{t,i}}{p_{t,i} + (1 - p_{t,i})r}.$$

Definition geometrical random variables

$$M_t^* = \min\{1 \leq i < N : O'_{t-1,i} = 1\} \cup \{N\}$$

$$G_{t,i} = \min(K_{t,i}, M_t)$$

Independence

$p_{t+2,i}$ estimated at the end of the step t .

So we can compute,

$$\hat{l}_{t,i} = G_{t,i} O_{t,i} l_{t,i}$$

Generalizing

ESTIMATE_R gives a safe lower bounding on r .

If $\underline{r} = 0$, we run vanilla EXP3.

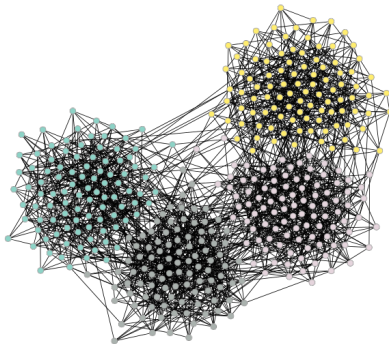
If $\underline{r} \geq \frac{\log T}{N}$, we run vanilla DUPLEX.

If $0 < \underline{r} < \frac{\log T}{N}$, $A = \left\lceil \frac{\log T}{N\underline{r}} \right\rceil$ and run DUPLEX grouping A steps.

Table of contents

- 1 EXP3 on Erdős-Rényi Graph
- 2 EXP3 on Stochastic Block-Model Graphs
- 3 Results

Random graph with Stochastic Block-Model



Neighbor communities

Probability of side information

- N clusters
- Labeled vertices
- $R = (r_{ij})_{1 \leq i, j \leq n}$ the matrix representing the probability of having side information.
- r_{ij} represents the probability that a vertex of the cluster j reveals his loss to a given vertex of the cluster i

Adapt M_t sampling

Unbiased estimator

$$\hat{l}_{t,i}^* = \frac{O_{t,i} l_{t,i}}{p_{t,i} + (1 - p_{t,i}) r_{C(l_t), C(i)}}$$

Using data...

$$\hat{l}_{t,i} = G_{t,i} O_{t,i} l_{t,i} \quad (1)$$

- Estimator $G_{i,t} \sim G(p_{t,i} + (1 - p_{t,i}) r_{l_t,i})$
 - M_t^i truncated geometric law of parameter $p_{t,i}$
- $G_{t,i} = \min(K_{t,i}, M_t^i)$

Sampling M_t^i

Algorithm adaptation

- Using previous observations
- Ensure independence of M_t and O_t
- Find the last $O_{t'}$, with $t' \equiv t - 1[2]$ in which $C(I(t')) = C(I(t))$
- Average the distance between two observation within cluster $C(j)$.

Keeping track of last time we picked I in cluster i

Lower bounding R

- We adapt `ESTIMATE_R` so that it returns \underline{R} where $\underline{r}_{ij} \leq r_{ij}$ with high probability.
- This is done at the relatively low cost

$$\mathbb{E}(\tau) \leq \sum_{i,j \in [N]^2} \frac{4 \log T}{N_i} \mathbf{1}_{r_{ij} > \frac{1}{N_j}} + \sqrt{T} \mathbf{1}_{r_{ij} < \frac{1}{N_j}} + \text{Cste}$$

Generalized algorithm with stochastic blockmodel graphs

If $\underline{R} = 0$,

we run vanilla EXP3.

If $\min_{i,j, \underline{r}_{i,j} > 0} (\underline{r}_{ij} N_j) \geq \log T$,

we run adapted DUPLEX.

Else,

$$A = \left\lceil \frac{\log T}{r_{N^*}} \right\rceil \text{ where}$$

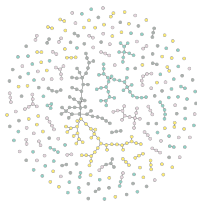
$$r_{N^*} = \min_{i,j, \underline{r}_{i,j} > 0} (\underline{r}_{ij} N_j)$$

and run DUPLEX grouping A steps.

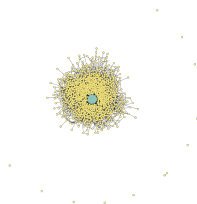
Table of contents

- 1 EXP3 on Erdős-Rényi Graph
- 2 EXP3 on Stochastic Block-Model Graphs
- 3 Results

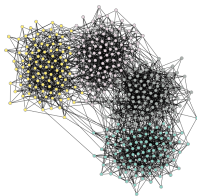
Test graphs



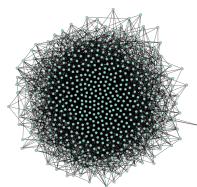
Weakly associative



Unbalanced

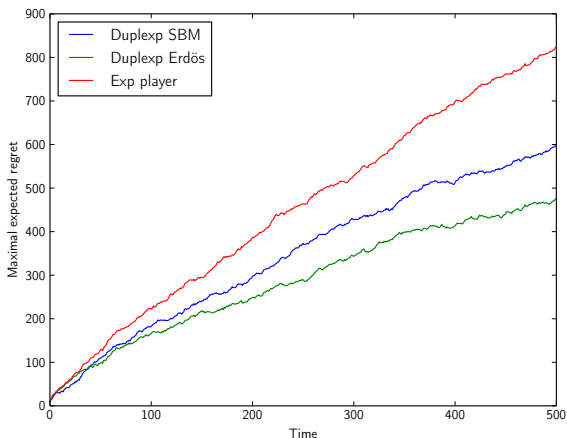


Neighbours



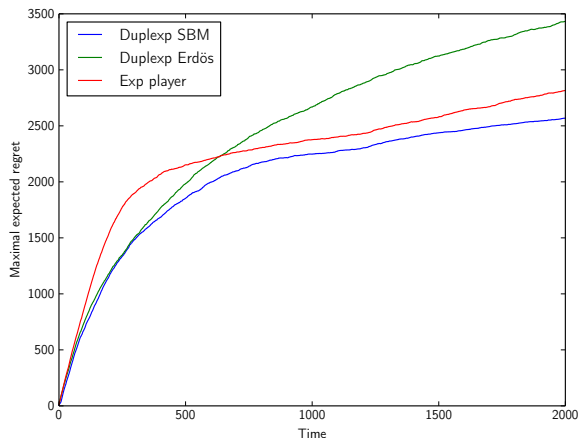
Erdos-Rényi

Unbalanced



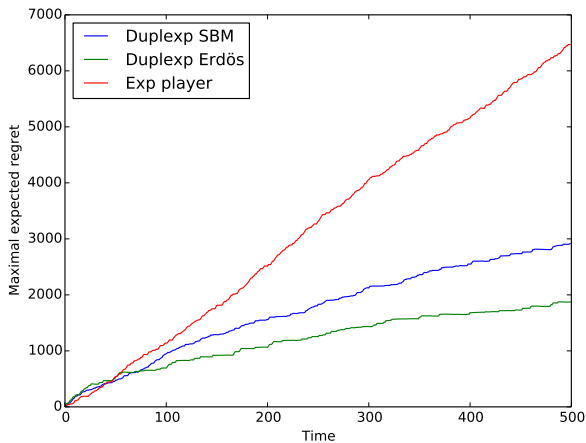
Easy problem

Unbalanced



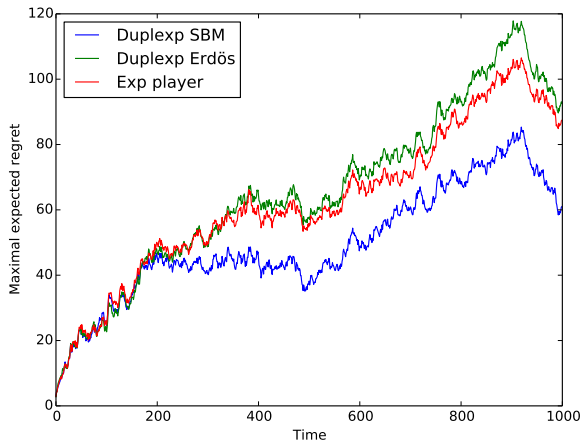
Hard problem

Neighbours



Easy problem

Neighbours



Hard problem