Supplementary material for Information Interaction Profile of Choice Adoption

Gael Poux-Medard[1] [0000−0002−0103−8778], Julien Velcin[1] [0000−0002−2262−045X], and Sabine Loudcher[1] [0000−0002−0494−0169]

[1] Université de Lyon, Lyon 2, ERIC UR 3083, 5 avenue Pierre Mendès France, F69676 Bron Cedex, France
gael.poux-medard@univ-lyon2.fr
julien.velcin@univ-lyon2.fr
sabine.loudcher@univ-lyon2.fr

1 Datasets

We provide details on the way datasets have been built from raw data. For each of the real-world datasets, we choose to consider only the order of apparition of the various entities instead of their absolute appearance times. This implies setting the time separating two successive exposures as constant, that we note δt. This choice is supported by state-of-the-art works [4] and we observed in our own experiments that it is more relevant than considering absolute times. Besides, we do not consider the first 10 pieces of information of any sequence to avoid boundary effects (the first 5 steps for the PD dataset): the history of exposures is incomplete in this case and could lead to biased results. For each dataset entities list, the number before the entity name is the key used in Fig.4 of the main article. The entities subsets have been chosen by computing the co-occurrence matrix of all the entities, and then select the ones that are part of a cluster using a K-mean algorithm. The datasets are:

- **Twitter** [3]: it consists in a collection of all the tweets containing URLs that have been posted on Twitter during October 2010, with the associated followers networks. A tweet read by a user in her feed is an exposition, and its possible retweet is a contagion. We consider only the URLs associated with the following URL shortening websites, the same as in [6]: {0: migre.me, 1: bit.ly, 2: tinyurl, 3: t.co}. The final dataset is made of 104,349 sequences of average length 53.5 steps (1 step = t_s), for 1,276,670,965 observed interactions.
- Prisoner's dilemma dataset (**PD**) [5, 1]: contains ordered sequences of repeated Prisoner's dilemma game between two players. From the dataset introduced in [5], we consider the sub-dataset noted BR-risk 0 (first entry of Tab.2 in the reference); we choose this subset in order to have decisions made in an homogeneous context, where players struggle in a dilemma that is hard to solve (which depends on the combination of the parameters T, R, S and P discussed further). Within each round, the players can either defect of cooperate. Each duel is made of 10 rounds. If both cooperate, the

reward R is high, if they both defect, the reward P is low, if one player co-operates while the other defects, this one gets a penalty S, while the other gets a reward T. To make the game a Prisoner dilemma, the variables have to obey T>R>P>S. We refer to the combination of players' actions ("the user cooperated and the opponent defected at time t") as exposures, and to the *defect* actions of the player in the following round as a contagion. We defined the action of cooperating as a non-contagion. We therefore have 4 possible situations ({0: Player cooperated and opponent defected, 1: Both players defected, 2: Both players cooperated, 3: Player defected and opponent cooperated}) and 2 possible outcomes (Player cooperates or defects). The final dataset is made of 2,337 sequences of average length 10.0 steps, for 189,297 observed interactions.
- Taobao dataset (**Ads**): contains all ads exposures for 1,140,000 randomly sampled users from the website of Taobao for 8 days (5/6/2017-5/13/2017) [2]. Taobao is one of the largest e-commerce websites, and is owned by Alibaba. To each exposure is associated the corresponding timestamp and the action of the user (click on the ad or not). A click is considered as a contagion. The subset of ads we consider is: {0: 4520, 1: 4280, 2: 1665, 3: 4282}. The resulting dataset is made 87,500 sequences of average length 23.9 steps, for 240,932,401 observed interactions.

2 Implementation of Clash of the Contagions

In this appendix, we provide technical details on the way the Clash of Contagions baseline is implemented. Following the directions given in the reference article [4], we implemented a Stochastic Gradient Descent (SGD) method for parameters inference. Given the small number of entities considered in the experiments, each iteration of the SGD is computed using the full dataset instead of slicing it into mini-batches.

Setup For each corpus, we run the SGD algorithm 100 times, from which we save the parameters maximizing the likelihood the most. At the beginning of each run, parameters M and Δ are randomly initialized. The stopping condition makes the algorithm ends when the relative variation of the likelihood according to the last iteration is been lesser than 10^{-6} for more than 30 times in a row; those numbers have been chosen empirically to maximize the performances of the algorithm. The hyper-parameters have been set to: T=5 (number of clusters) and K=20 (number of considered time steps).

Update rule In each iteration, we update the parameters in the direction of the gradient descent (noted G). However, a major problem when dealing with SGD is to choose the line step length η (the amplitude of the variation of the parameters in the direction of the gradient G). After each iteration, we compare several update rules, and we select the one maximizing the likelihood. Those rules are as follows:

– AdaGrad: $\eta^{AG} \times G$
– AdaDelta: $\eta^{AD} \times G$
– Line search in the direction of the gradient: $\eta^{LS} \times G$
– Line search in the direction of AdaDelta: $\eta^{LS} \times \eta^{AD} \times G$

The line search snippets consider 50 values of η^{LS} logarithmically distributed in the interval $[10^{-6}; 10^{3}]$.

Constraints on the parameters The membership vectors entries $M_{i,t}$ (membership of i to cluster t) must be positive and sum to 1 over all the clusters ($\sum_t M_{i,t} = 1$). In order to enforce this constraint, we consider the following variable change: $M_{i,t} \rightarrow \frac{\phi_{i,t}^2}{\sum_{t'} \phi_{i,t'}^2}$. This transformation guarantees the membership vector properties with no need for penalty methods in the implementation.

Besides, as stated in [4], it can happen that a probability is larger then 1 or lesser than 0. In the absence of complementary details in the main paper, we implemented our own method to force the probabilities to stay within reasonable bounds. Here it is impossible to make a simple variable change to enforce this constraint, since the probability results of a non-linear combination of the model's parameters. We added to the likelihood an exponential penalty term. Let P denote a quantity we want to constrain between 0 and 1. The penalty term equals $e^{-\lambda P} + e^{\lambda(P-1)}$. λ here is a parameter that tunes the intensity of the penalty, and is empirically set to $\lambda = 75$. This penalty function has the form of a well with very steep walls in x=0 and x=1. In this way, it seldom happens that probabilities are larger than 1 or lesser than 0, as said in the main article. When such cases happen, we simply set it back to the closest bound for methods comparisons.

3 Experimental results with standard deviation

The numerical results along with the associated standard deviation are presented Table 1.

Table 1. Experimental results with associated standard deviation

		RSS	JS div.	BCF1	MSE β
Synth-20	IR-RBF	18.42(25)	0.002 284(45)	0.9188(1)	0.005 03(4)
	ICIR	139.59(77)	0.009 980(49)	0.8271(16)	0.015 88(0)
	Naive	145.51(75)	0.010 379(59)	0.8221(2)	
	CoC	123.06(75)	0.009 384(70)	0.8220(15)	
	IMMSBM	222.06(139)	0.017 288(51)	0.7265(7)	
Synth-5	IR-RBF	0.117(4)	0.000 217(6)	0.9742(4)	0.005 30(3)
	ICIR	8.266(36)	0.008 117(38)	0.8499(11)	0.019 21(1)
	Naive	10.026(48)	0.009 956(48)	0.8214(17)	
	CoC	0.115(23)	0.000 197(31)	0.9763(10)	
	IMMSBM	11.694(274)	0.013 622(489)	0.7693(50)	
Twitter	IR-RBF	0.0015(4)	0.000 058(8)	0.9832(8)	
	IR-EXP	0.0011(2)	0.000 049(5)	0.9862(6)	
	ICIR	0.0137(8)	0.000 629(48)	0.9614(12)	
	Naive	0.0161(9)	0.000 725(60)	0.9379(38)	
	CoC	0.0017(2)	0.000 067(4)	0.9572(263)	
	IMMSBM	0.0147(13)	0.000 683(73)	0.9543(29)	
PD	IR-RBF	1.13(18)	0.007 583(470)	0.9789(27)	
	IR-EXP	1.55(31)	0.008 669(1137)	0.9661(22)	
	ICIR	3.54(31)	0.018 225(1286)	0.9381(47)	
	Naive	3.65(43)	0.019 147(1538)	0.9455(23)	
	CoC	1.24(28)	0.008 088(1266)	0.9736(56)	
	IMMSBM	20.38(297)	0.087 010(12172)	0.7672(204)	
Ads	IR-RBF	0.0043(6)	0.000 043(6)	0.9814(19)	
	IR-EXP	0.0030(3)	0.000 030(2)	0.9852(7)	
	ICIR	0.0983(70)	0.000 848(63)	0.9659(11)	
	Naive	0.1453(106)	0.001 263(95)	0.9126(59)	
	CoC	0.0045(2)	0.000 045(2)	0.9741(40)	
	IMMSBM	0.0155(18)	0.000 153(15)	0.9543(16)	

References

1. Bereby-Meyer, Y., Roth, A.E.: The speed of learning in noisy games: Partial reinforcement and the sustainability of cooperation. American Economic Review **96**(4), 1029–1042 (2006)
2. Cao, J., Sun, W.: Sequential choice bandits: learning with marketing fatigue. AAAI-19 (2019)
3. Hodas, N.O., Lerman, K.: The simple rules of social contagion. Scientific Reports **4**(4343) (2014)

4. Myers, S., Leskovec, J.: Clash of the contagions: Cooperation and competition in information diffusion. IEEE pp. 539–548 (2012)
5. Nay, J.J., Vorobeychik, Y.: Predicting human cooperation. PLoS One **11**(5) (2016)
6. Zarezade, A., Khodadadi, A., Farajtabar, M., Rabiee, H.R., Zha, H.: Correlated cascades: Compete or cooperate. AAAI (2017)