# IDENTIFYING OPINION INFLUENCERS OVER SOCIAL NETWORKS

*Valentina Shumovskaia, Mert Kayaalp, and Ali H. Sayed*

École Polytechnique Fédérale de Lausanne (EPFL)

## ABSTRACT

The adaptive social learning paradigm deals with the opinion formation process by a network of communicating agents in a dynamic environment. In this study, we show that a sequence of publicly exchanged beliefs allows users to discover rich information about the underlying model. In particular, it is shown that it is possible (i) to identify the influence of each individual agent to the objective of truth learning, (ii) to discover how well-informed each agent is, and (iii) to learn the underlying network topology.

***Index Terms***— Social learning, social influence, explainability, inverse modeling, online learning, graph learning.

## 1. INTRODUCTION AND RELATED WORK

The social learning paradigm is a popular non-Bayesian formulation that enables a group of networked agents to learn and track the state of nature. It has motivated several studies in the literature with many useful variations under varied modeling assumptions (see, e.g., [1, 2, 3, 4, 5, 6, 7]). Under this framework, agents observe streaming data and share information with their immediate neighbors. Through a process of localized cooperation, the agents continually update their *beliefs* about the underlying state. The main question in social learning is whether agents are able to learn the truth eventually, i.e., whether the beliefs on the wrong hypotheses will vanish. In this work, we adopt the *Adaptive Social Learning* (ASL) strategy from [8], which showed how to extend traditional non-Bayesian learning under *fixed* truth to *dynamic* scenarios where the state of nature is allowed to drift with time.

Given a collection of networked agents tracking the state of nature by means of the adaptive social learning (ASL) strategy, our main objective is to focus on two questions related to *explainability* and *inverse modeling*. In particular, by observing the sequence of publicly exchanged beliefs, we would like to discover the underlying graph topology (i.e., how the agents are connected to each other). We would also like to discover each agent's contribution (or influence) to the network's learning process.

Emails: {valentina.shumovskaia, mert.kayaalp, ali.sayed}@epfl.ch.
This work was supported in part by grant 205121-184999 from the Swiss National Science Foundation.

The question of explainability over graphs is a challenging task, and it has been receiving increasing attention — see, e.g., [9, 10, 11, 12], where explainability is considered from different perspectives and with different aims. Overall, higher transparency and a better understanding of the solutions are generally crucial for critical applications. Likewise, identifying the most influential users and their communication patterns can provide valuable insights in social network analysis [13]. Actually, the problem of identifying the most influential nodes in a network is also increasingly relevant [14, 15, 16, 17, 18, 19], especially following the rise of online social networks. Once identified, this information can be useful in many contexts. For example, it can be used to enhance recommendations for marketing purposes [14, 15, 16, 17, 18, 19], where the objective is to maximize the number of influenced nodes.

## 2. SOCIAL LEARNING MODEL

We consider a collection of agents $\mathcal{N}$ performing peer-to-peer exchanges of beliefs according to some combination matrix $A_\star$ with non-negative entries, $[A_\star]_{\ell,k} = a_{\ell k} \geq 0$. Agent $\ell$ is able to communicate with agent $k$ when $a_{\ell k}$ is positive; this scalar refers to the weight that agent $k$ assigns to the information received from agent $\ell$. We assume that the matrix $A_\star$ is *left-stochastic* and corresponds to a *strongly connected* graph [6, 2, 8]. It follows from the Perron-Frobenius theorem [20, Chapter 8], [21] that the power matrix $A_\star^t$ converges to $u\mathbb{1}^\mathsf{T}$ as $t \to \infty$ at an exponential rate, where $u$ is the Perron eigenvector that satisfies:

$$A_\star u = u, \qquad u_\ell > 0, \qquad \sum_{\ell \in \mathcal{N}} u_\ell = 1. \tag{1}$$

where the $u_\ell$ denote the individual entries of $u$. Each of these entries describes the centrality of the corresponding agent in the graph.

At each time instant $i$, each agent $k$ observes a measurement $\boldsymbol{\zeta}_{k,i}$. We assume initially that each agent $k \in \mathcal{N}$ has access to private likelihood functions, $L_k(\zeta|\theta)$, which describe the distribution of the observation $\zeta$ conditioned on each potential model $\theta$. The observations $\boldsymbol{\zeta}_{k,i}$ are assumed to be independent and identically distributed (i.i.d.) over time. In order to be able to distinguish the true hypothesis $\theta^\star$ from any other hypothesis $\theta \neq \theta^\star$, we need to assume that there exists at least one clear-sighted agent $k \in \mathcal{N}$ that has strictly positive KL-divergences relative to the true likelihood, i.e., $D_{KL}\left(L_k\left(\theta\right)||L_k\left(\theta^\star\right)\right) > 0$, for all $\theta \neq \theta^\star \in \Theta$. The following boundedness assumption on the likelihood is common in the literature [22, 12]; it essentially amounts to assuming that the likelihoods share support regions.

**Assumption 1 (Bounded likelihoods).** *There exists a finite constant $b > 0$ such that, for all $k \in \mathcal{N}$:*

$$\left|\log \frac{L_k(\boldsymbol{\zeta}|\theta)}{L_k(\boldsymbol{\zeta}|\theta')}\right| \leq b \tag{2}$$

*for all $\theta,\ \theta' \in \Theta$, and $\boldsymbol{\zeta}$.* ■

Next, we describe the ASL strategy from [8]. At each time step $i$, each agent $k$ performs a local update based on the newly received observation and forms the *intermediate (public)* belief:

$$\boldsymbol{\psi}_{k,i}(\theta) = \frac{L_k^\delta(\boldsymbol{\zeta}_{k,i}|\theta)\boldsymbol{\mu}_{k,i-1}^{1-\delta}(\theta)}{\sum_{\theta'\in\Theta} L_k^\delta(\boldsymbol{\zeta}_{k,i}|\theta')\boldsymbol{\mu}_{k,i-1}^{1-\delta}(\theta')}, \quad k \in \mathcal{N}. \tag{3}$$

Here, $\delta \in (0,1)$ is a step-size parameter that controls the adaptation capacity. Subsequently, agent $k$ fuses the beliefs received from its neighbors, i.e., from all agents for which $a_{\ell k} > 0$:

$$\boldsymbol{\mu}_{k,i}(\theta) = \frac{\prod_{\ell\in\mathcal{N}_k} \boldsymbol{\psi}_{\ell,i}^{a_{\ell k}}(\theta)}{\sum_{\theta'\in\Theta}\prod_{\ell\in\mathcal{N}_k} \boldsymbol{\psi}_{\ell,i}^{a_{\ell k}}(\theta')}, \quad k \in \mathcal{N}. \tag{4}$$

Once (4) is performed, the true state $\theta^\star$ can be estimated by agent $k$ at time $i$ using the maximum a-posteriori construction over either the private ($\boldsymbol{\mu}_{k,i}$) or public ($\boldsymbol{\psi}_{k,i}$) beliefs.

Following [12], we introduce two matrices $\boldsymbol{\Lambda}_i$ and $\boldsymbol{\mathcal{L}}_i$ in order to represent the recursions (6) in a more compact matrix form as follows:

$$\boldsymbol{\Lambda}_i = (1-\delta)A_\star^\mathsf{T}\boldsymbol{\Lambda}_{i-1} + \delta\boldsymbol{\mathcal{L}}_i. \tag{5}$$

The matrices are of size $|\mathcal{N}| \times (|\Theta|-1)$, and their entries are log-belief and log-likelihood ratios given by:

$$[\boldsymbol{\Lambda}_i]_{k,j} \triangleq \log\frac{\boldsymbol{\psi}_{k,i}(\theta_0)}{\boldsymbol{\psi}_{k,i}(\theta_j)}, \quad [\boldsymbol{\mathcal{L}}_i]_{k,j} \triangleq \log\frac{L_k(\boldsymbol{\zeta}_{k,i}|\theta_0)}{L_k(\boldsymbol{\zeta}_{k,i}|\theta_j)} \tag{6}$$

where the reference state $\theta_0 \in \Theta$ can be chosen at will by the designer.

After a sufficient number of iterations $i$ (i.e., for $i > M \gg 1$), the average of the log-belief matrix converges in probability to a limit value given by [12, Lemma 1] as follows:

$$\frac{1}{M}\sum_{j=i-M}^{i-1}\boldsymbol{\Lambda}_j \xrightarrow{M\to\infty} \mathbb{E}\boldsymbol{\Lambda}. \tag{7}$$

The matrices $\boldsymbol{\mathcal{L}}_i$ are i.i.d. over time due to the assumed i.i.d. assumption on the observations. We introduce the following condition on the higher-order moments of $\boldsymbol{\mathcal{L}}_i$ [8, 12].

**Assumption 2 (Positive-definite covariance matrix).** *The covariance matrix $\mathcal{R}_{\boldsymbol{\mathcal{L}}}$ is uniformly positive-definite for all $i \geq 0$, i.e., there exists $\tau > 0$ such that:*

$$\mathcal{R}_{\boldsymbol{\mathcal{L}}} \triangleq \mathbb{E}\left(\boldsymbol{\mathcal{L}}_i - \bar{\mathcal{L}}\right)\left(\boldsymbol{\mathcal{L}}_i - \bar{\mathcal{L}}\right)^\mathsf{T} \geq \tau I. \tag{8}$$

■

## 3. INVERSE LEARNING FROM PUBLIC BELIEFS

### 3.1. Problem Statement

The data available from the social network might be limited for various reasons, including privacy. Therefore, in this work, we assume that we can only observe the evolution of the public beliefs over time:

$$\left\{\boldsymbol{\psi}_{k,i}(\theta)\right\}_{i\gg1}, \quad \forall k \in \mathcal{N} \tag{9}$$

Here, by $i \gg 1$ we underline that we are observing $\boldsymbol{\Lambda}_i$ after a sufficient amount of iterations, i.e., after $\boldsymbol{\Lambda}_i$ has reached its steady-state distribution.

A good illustration for this setting is the social network of Twitter users. From each post (or tweet) that a user posts to their followers, we can extract an intermediate belief $\boldsymbol{\psi}_{k,i}$ based on sentiment analysis, a.k.a., opinion mining. After that, each user $k$ reads the posts of its followers and constructs the private belief $\boldsymbol{\mu}_{k,i}$ according to (4).

### 3.2. Algorithm Development

The previous work on learning the combination matrix $A_\star$ in [12] assumes that the expected log-likelihood matrix $\bar{\mathcal{L}} \triangleq \mathbb{E}\boldsymbol{\mathcal{L}}_i$ is known, where it can be verified that:

$$\begin{aligned}&[\bar{\mathcal{L}}]_{k,j} \triangleq [\mathbb{E}\boldsymbol{\mathcal{L}}_i]_{k,j}\\ &= D_{\text{KL}}(L_k(\theta^\star)||L_k(\theta_j)) - D_{\text{KL}}(L_k(\theta^\star)||L_k(\theta_0)).\end{aligned} \tag{10}$$

In this work, we do not assume that $\bar{\mathcal{L}}$ is known beforehand, and will instead estimate $\bar{\mathcal{L}}$ at each iteration $i$ by using

$$\widehat{\boldsymbol{\mathcal{L}}}_{i-1}(A) = \frac{1}{\delta M}\sum_{j=i-M}^{i-1}\left(\boldsymbol{\Lambda}_j - (1-\delta)A^\mathsf{T}\boldsymbol{\Lambda}_{j-1}\right), \tag{11}$$

where $A$ will be the estimate that is available for $A_\star$ at that point in time. Accordingly, the risk function is strongly convex and has Lipschitz gradients, and is given by:

$$J(A) \triangleq \frac{1}{2(N-M)}\sum_{i=M}^{N-1}\mathbb{E}\|\boldsymbol{\Lambda}_i - (1-\delta)A^\mathsf{T}\boldsymbol{\Lambda}_{i-1} - \delta\bar{\mathcal{L}}\|_\text{F}^2 \tag{12}$$

It can be shown that the unique minimizer of this risk function, denoted by $A_{\min}$, gets closer to the true combination matrix as $M$ grows:

$$\|A_{\min} - A_\star\|_{\mathrm{F}}^2 = O\left(1/\delta^2 M^2\right) \tag{13}$$

To minimize $J(A)$, we apply stochastic gradient descent (SGD) with constant step-size $\mu > 0$. At each iteration $i$, the estimate $\boldsymbol{A}_i$ for the combination matrix is updated via:

$$\begin{aligned}\boldsymbol{A}_i = \boldsymbol{A}_{i-1} + \mu(1-\delta)\left(\boldsymbol{\Lambda}_{i-1} - \frac{1}{M}\sum_{j=i-M}^{i-1}\boldsymbol{\Lambda}_{j-1}\right)\\ \times\left(\boldsymbol{\Lambda}_i^\mathsf{T} - (1-\delta)\boldsymbol{\Lambda}_{i-1}^\mathsf{T}\boldsymbol{A}_{i-1} - \delta\widehat{\boldsymbol{\mathcal{L}}}_{i-1}^\mathsf{T}\right)\end{aligned} \tag{14}$$

## 4. GLOBAL INFLUENCE IDENTIFICATION

In this section, we establish a strong connection between the probability of error for truth learning and the *network divergence*. The network divergence is defined in terms of the Perron eigenvector of $A_\star$, and the KL-divergences between the likelihoods:

$$K(\theta^\star, \theta) \triangleq \sum_{k\in\mathcal{N}} u_k D_{\mathrm{KL}}(L_k(\theta^\star)||L_k(\theta)) > 0. \tag{15}$$

The probability of truth learning error is defined as the probability of picking a wrong hypothesis $\theta \neq \theta^\star$:

$$p_{k,i} = \mathbb{P}\left(\exists\theta\neq\theta^\star\colon\ \log\frac{\boldsymbol{\psi}_{k,i}(\theta^\star)}{\boldsymbol{\psi}_{k,i}(\theta)} \leq 0\right). \tag{16}$$

For this subsection, we additionally assume that the observations $\{\boldsymbol{\zeta}_{k,i}\}$ are independent between different agents. Now, we know from [8, Theorem 3] that each random variable (for any $k$) in (16) can be approximated by a Gaussian random variable in the steady state with the following moments:

$$\log\frac{\boldsymbol{\psi}_{k,i}(\theta^\star)}{\boldsymbol{\psi}_{k,i}(\theta)} \approx \mathcal{G}\Big(K\left(\theta^\star,\theta\right) + O\left(\delta\right), \delta C + O\left(\delta^2\right)\Big) \tag{17}$$

for some finite and constant covariance matrix, $C$. Thus, the probability of error (16) becomes the probability of the Gaussian random variable (17) assuming negative values for at least one $\theta\in\Theta$. This Gaussian random variable concentrates around its mean (i.e., the network divergence in (15)), which is positive.

The larger the contribution of agent $k$ to the network divergence (15) is, the stronger its influence will be towards moving the network away from an erroneous decision. We therefore say that

$$u_k D_{\mathrm{KL}}(L_k(\theta^\star)||L_k(\theta)) \tag{18}$$

determines the amount of information that agent $k$ has about $\theta$ agreeing with $\theta^\star$. We define the level of informativeness of agent $k$ to the learning process by considering the aggregate of its contributions for all $\theta$:

$$I_k \triangleq u_k \sum_{\theta\in\Theta} D_{\mathrm{KL}}(L_k(\theta^\star)||L_k(\theta)). \tag{19}$$

This quantity serves as a measure of *influence*, since agents with large $I_k$ contribute the most to learning the truth by the network.

In what follows, we describe how to estimate the quantities $I_k$ by the learning algorithm. First, to obtain the Perron eigenvector for $\boldsymbol{A}_i$, we need to normalize any of its eigenvectors corresponding to the eigenvalue at 1. We let $j'$ denote the index within the hypothesis set $\Theta$ that maximizes the public belief after a sufficient number of iterations $i_N$:

$$\widehat{\boldsymbol{\theta}}_{j'} = \arg\max_{\theta\in\Theta}\boldsymbol{\psi}_{k,i_N}(\theta). \tag{20}$$

Returning to (10), we approximate the KL-divergences by

$$D_{\mathrm{KL}}\left(L_k(\theta^\star)||L_k(\theta_0)\right) \approx -[\widehat{\boldsymbol{\mathcal{L}}}_{i_N}]_{k,j'}, \tag{21}$$

$$D_{\mathrm{KL}}\left(L_k(\theta^\star)||L_k(\theta_j)\right) \approx [\widehat{\boldsymbol{\mathcal{L}}}_{i_N}]_{k,j} + [\widehat{\boldsymbol{\mathcal{L}}}_{i_N}]_{k,j'}, \tag{22}$$

where $\widehat{\boldsymbol{\mathcal{L}}}_{i_N}$ is an estimate for $\bar{\mathcal{L}}$.

## 5. THEORETICAL RESULTS

To investigate the steady-state performance of recursion (14), we adopt the following independence assumption, which is typical in the study of adaptive systems [23, 24].

**Assumption 3 (Separation principle).** *We denote the estimation error by $\widetilde{\boldsymbol{A}}_i \triangleq A_\star - \boldsymbol{A}_i$, and assume the step-size $\mu$ is small enough to allow $\|\widetilde{\boldsymbol{A}}_i\|_F^2$ to attain a steady-state distribution. The separation principle states that the error $\widetilde{\boldsymbol{A}}_i$ is independent of the observations $\boldsymbol{\Lambda}_i, \dots, \boldsymbol{\Lambda}_{i-M}$, conditioned on the history of previous observations.* ■

The following results quantify the performance of the proposed algorithm. Proofs are omitted due to space limitations.

**Theorem 1 (Steady-state performance).** *In the limit, the mean-square deviation (MSD) satisfies:*

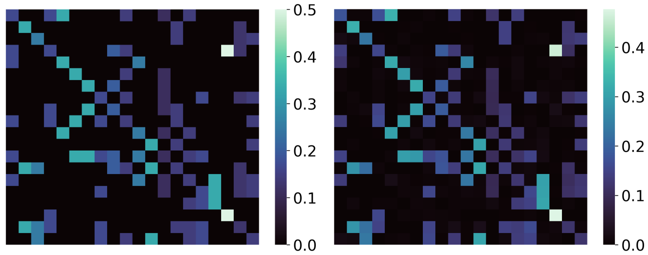
$$\limsup_{i\to\infty}\mathbb{E}\|\widetilde{\boldsymbol{A}}_i\|_{\mathrm{F}}^2 \leq O(\mu) + O(1/\delta^3 M^2). \tag{23}$$

**Theorem 2 (Steady-state log-likelihood learning).** *The MSD converges exponentially fast with*

$$\begin{aligned}&\limsup_{i\to\infty}\mathbb{E}\|\widehat{\boldsymbol{\mathcal{L}}}_i - \bar{\mathcal{L}}\|_F^2\\ &\leq \frac{1}{M}\mathrm{Tr}\left(\mathcal{R}_{\boldsymbol{\mathcal{L}}}\right) + O(\mu/\delta^2) + O\left(1/\delta^5 M^2\right)\end{aligned} \tag{24}$$

*where* $\mathcal{R}_{\boldsymbol{\mathcal{L}}} = \mathbb{E}\left(\boldsymbol{\mathcal{L}}_i - \bar{\mathcal{L}}\right)\left(\boldsymbol{\mathcal{L}}_i - \bar{\mathcal{L}}\right)^\mathsf{T}$.

We can reduce the limiting MSD for both problems by using arbitrary small step-size $\mu \ll \delta^2$ and by the number of samples $M$.




(a) True combination matrix. (b) Learned combination matrix.

**Fig. 1**: True combination matrix and learned combination matrices using $M = 50$.

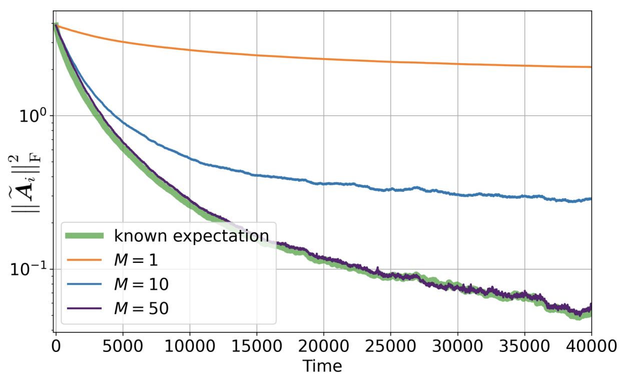



**Fig. 2**: Algorithm performance when $\bar{\mathcal{L}}$ is known and when estimated by (11) for different $M \in \{1, 10, 50\}$.

## 6. COMPUTER SIMULATIONS

We generate a graph with $|\mathcal{N}| = 20$ agents according to the Erdos-Renyi model with an edge probability of $p = 0.2$. We set the adaptation hyperparameter to $\delta = 0.05$. Then, we generate the combination weights (see Fig. 1a) with uniform weights in the column, such that the resulting matrix is left-stochastic. We consider $|\Theta| = 5$ states, where the likelihood models $L_k(\theta)$ for each agent $k \in \mathcal{N}$ are assumed to follow a binomial distribution with randomly generated parameters. We generate likelihood models such that we observe only 3 agents with high informativeness.

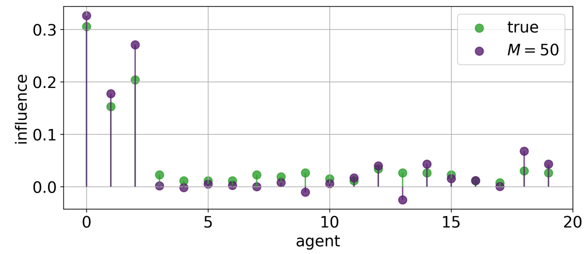
First, we consider how well the combination matrix is learned for different $M \in \{1, 10, 50\}$. We additionally compare with [12], where the expectation $\bar{\mathcal{L}}$ was assumed to be known beforehand. For $M = 50$, we use $\mu = 0.1$, for $M = 10$, we use $\mu = 0.01$, and for $M = 1$, we use $\mu = 0.001$ for better convergence. In Figure 2, we plot the reconstruction error with respect to the iteration number:

$$\|\widetilde{\boldsymbol{A}}_i\|_{\mathrm{F}}^2 = \|\boldsymbol{A}_i - A_\star\|_{\mathrm{F}}^2 \tag{25}$$

We notice that the higher $M$ improves the limiting MSD as reflected in Theorem 1.

Figure 3 illustrates how well the learned KL-divergences

(a) 3 influential agents.




(b) 1 influential agent.

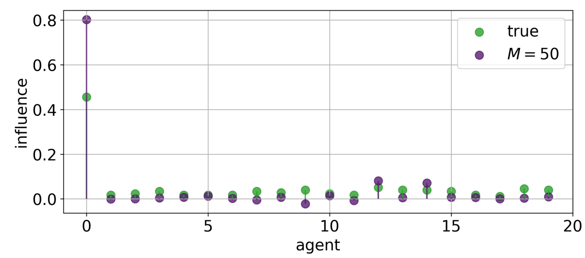



**Fig. 3**: Agents' influences (19) based on the learned graph and KL-divergences.

and combination matrix can recover the global influences (19). For better interpretability, we normalize the values so that they add up to one. We see that for some agents, the algorithm does not perfectly recover these components, but yet allows us to identify that the first agents are driving the learning the most. This property allows us to search for agents that are the most contributing to learning the true state.

## 7. CONCLUSIONS

In this study, we show that a sequence of publicly exchanged beliefs in the adaptive social learning protocol contains rich information about the underlying model. We present an algorithm for learning the agents' informativeness in terms of KL-divergences between likelihood models, and for identifying a combination graph. We demonstrate that these quantities determine the probability of error of the true hypothesis estimator, and we introduce a notion of a global agent influence, which quantifies the individuals' contribution to learning. As a result, the suggested approach enables us to determine the most influential agents in the opinion formation process. Our experiments illustrate that we can accurately find global influencers and learn the underlying graph.

## 8. REFERENCES


[1] A. Jadbabaie, P. Molavi, A. Sandroni, and A. Tahbaz-Salehi, "Non-Bayesian social learning," *Games and Economic Behavior*, vol. 76, no. 1, pp. 210–225, 2012.

[2] A. Nedić, A. Olshevsky, and C. A. Uribe, "Fast convergence rates for distributed non-Bayesian learning," *IEEE Transactions on Automatic Control*, vol. 62, no. 11, pp. 5538–5553, 2017.

[3] P. Molavi, A. Tahbaz-Salehi, and A. Jadbabaie, "Foundations of non-Bayesian social learning," *Columbia Business School Research Paper*, no. 15-95, 2017.

[4] X. Zhao and A. H. Sayed, "Learning over social networks via diffusion adaptation," in *Proc. Asilomar Conference on Signals, Systems and Computers*, 2012, pp. 709–713.

[5] P. Molavi, A. Tahbaz-Salehi, and A. Jadbabaie, "A theory of non-Bayesian social learning," *Econometrica*, vol. 86, no. 2, pp. 445–490, 2018.

[6] A. Lalitha, T. Javidi, and A. D. Sarwate, "Social learning and distributed hypothesis testing," *IEEE Transactions on Information Theory*, vol. 64, no. 9, pp. 6161–6179, 2018.

[7] Y. Inan, M. Kayaalp, E. Telatar, and A. H. Sayed, "Social learning under randomized collaborations," in *2022 IEEE International Symposium on Information Theory (ISIT)*, 2022, pp. 115–120.

[8] V. Bordignon, V. Matta, and A. H. Sayed, "Adaptive social learning," *IEEE Transactions on Information Theory*, vol. 67, no. 9, pp. 6053–6081, 2021.

[9] C. Rudin, C. Chen, Z. Chen, H. Huang, L. Semenova, and C. Zhong, "Interpretable machine learning: Fundamental principles and 10 grand challenges," *Statistics Surveys*, vol. 16, pp. 1–85, 2022.

[10] A. Heuillet, F. Couthouis, and N. Díaz-Rodríguez, "Collective explainable AI: Explaining cooperative strategies and agent contribution in multiagent reinforcement learning with shapley values," *IEEE Computational Intelligence Magazine*, vol. 17, no. 1, pp. 59–71, 2022.

[11] J. J. Ohana, S. Ohana, E. Benhamou, D. Saltiel, and B. Guez, "Explainable AI (XAI) models applied to the multi-agent environment of financial markets," in *Proc. Workshop on Explainable and Transparent AI and Multi-Agent Systems*, 2021, pp. 189–207.

[12] V. Shumovskaia, K. Ntemos, S. Vlaski, and A. H. Sayed, "Explainability and graph learning from social interactions," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 8, pp. 946–959, 2022.

[13] D. Camacho, Á. Panizo-LLedot, G. Bello-Orgaz, A. Gonzalez-Pardo, and E. Cambria, "The four dimensions of social network analysis: An overview of research methods, applications, and software tools," *Information Fusion*, vol. 63, pp. 88–120, 2020.

[14] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proc. ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2003, pp. 137–146.

[15] ——, "Influential nodes in a diffusion model for social networks," in *Proc. Intern. Colloquium on Automata, Languages, and Programming*, 2005, pp. 1127–1138.

[16] Y. Li, J. Fan, Y. Wang, and K.-L. Tan, "Influence maximization on social graphs: A survey," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 10, pp. 1852–1872, 2018.

[17] M. Gomez-Rodriguez, L. Song, N. Du, H. Zha, and B. Schölkopf, "Influence estimation and maximization in continuous-time diffusion networks," *ACM Transactions on Information Systems (TOIS)*, vol. 34, no. 2, pp. 1–33, 2016.

[18] J. Qiu, J. Tang, H. Ma, Y. Dong, K. Wang, and J. Tang, "Deepinf: Social influence prediction with deep learning," in *Proc. ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2018, pp. 2110–2119.

[19] W. Karoui, N. Hafiene, and L. Ben Romdhane, "Machine learning-based method to predict influential nodes in dynamic social networks," *Social Network Analysis and Mining*, vol. 12, no. 1, pp. 1–18, 2022.

[20] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge University Press, NY, 2013.

[21] A. H. Sayed, *Inference and Learning from Data*. Cambridge University Press, 2022, vols. 1–3.

[22] V. Bordignon, V. Matta, and A. H. Sayed, "Social learning with partial information sharing," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Barcelona, Spain, 2020, pp. 5540–5544.

[23] A. H. Sayed, *Adaptive Filters*. Wiley, 2008.

[24] ——, "Adaptation, learning, and optimization over networks," *Foundations and Trends® in Machine Learning*, vol. 7, no. 4-5, pp. 311–801, 2014.