Active Online Learning of Trusts in Social Networks

Hoi-To Wai[†], Anna Scaglione[†] and Amir Leshem[‡]

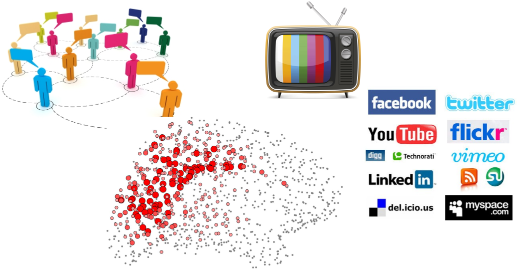
[†]School of Electrical, Computer and Energy Engineering, Arizona State University, USA.
[‡]Faculty of Engineering, Bar-Ilan University, Israel.

Acknowledgement: This work is supported by NSF CCF-1011811 and ISF 903/13.

Agenda

Learning trusts in social networks

- Nowadays, online social networks (OSNs) are dominating the way that human beings communicate $\Longrightarrow$ TONs of data for computational studies.



This work —

- *Learn trusts in social networks* from opinions expressed by individuals.
- Devise an *online algorithm* that learns the trusts from streaming data.

Opinion Dynamics & Notations

- A *social network* — $S = (V, E, \overline{\boldsymbol{W}})$, $V = [n] = \{1, ..., n\}$, $E \subseteq V \times V$.
- Let $x_i(t; s)$ be the ith agent's opinion[1] at time t and topic s.
- Let $\boldsymbol{x}(t; s) = [x_1(t; s); x_2(t; s); \ldots; x_n(t; s)]$, we have [DeG74]:

$$\text{DeGroot model} : \boldsymbol{x}(t; s) = \boldsymbol{W}(t; s)\boldsymbol{x}(t-1; s) \tag{1}$$

  where $\boldsymbol{W}(t; s)$ is stationary, $\mathbb{E}[\boldsymbol{W}(t; s)] = \overline{\boldsymbol{W}}$ and stochastic, $\boldsymbol{W}(t; s)\mathbf{1} = \mathbf{1}$.

- *Intuition* — agent weights the opinions of his/her neighbors and update to a weighted average. (similar to average consensus)
- Other models — e.g., H.-K. model [HK02], voter's model [HL75].

Our goal: to learn the trust matrix $\overline{\boldsymbol{W}} = \mathbb{E}[\boldsymbol{W}(t; s)]$.

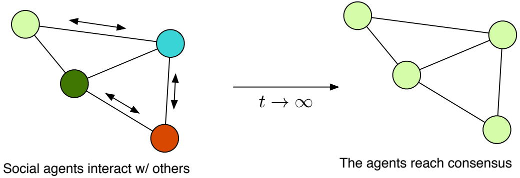
[1] Example: it can be a p.d.f. for attitude (buying Apple vs buying Samsung).

Previous work — "Passive" Method

- [Tim07, WLGY11, DBB+14] take the intuition from (1) and solves:

$$\min_{\boldsymbol{W}} \sum_{t=t_0+1}^{t_0+T} \|\boldsymbol{x}(t;s) - \boldsymbol{W}\boldsymbol{x}(t-1;s)\|_2^2 \ \text{s.t.} \ \boldsymbol{W}\mathbf{1} = \mathbf{1}, \ \boldsymbol{W} \geq \mathbf{0}. \qquad \text{(P0)}$$

- **Difficult to realize** as it takes $\boldsymbol{x}(t_0;s)$, $\boldsymbol{x}(t_0+1;s)$, $\boldsymbol{x}(t_0+2;s), \ldots$
  1. **Observability** – 'opinion updates' are random and happen in human brains. We cannot accurately track the 'discrete-time' update of opinions.
  2. **Stationarity** – At $t_0 \gg 0$, we may have $\boldsymbol{x}(t_0;s) \approx \boldsymbol{x}(t_0+1;s)$.



Proposed "Active" method

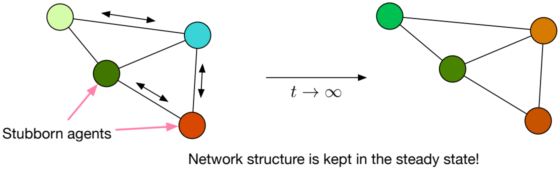
- **Stubborn agents** — agents whose opinions do not change.
- Let the first n_s agents be stubborn agents, we have:

$$\boldsymbol{W}(t;s) = \begin{pmatrix} \boldsymbol{I}_{n_s} & \boldsymbol{0} \\ \boldsymbol{B}(t;s) & \boldsymbol{D}(t;s) \end{pmatrix}, \tag{2}$$

- Let $\boldsymbol{z}(t;s)$, $\boldsymbol{y}(t;s)$ be the stubborn agents' and normal agents' opinions.

$$\mathbb{E}[\boldsymbol{y}(\infty;s)] = (\boldsymbol{I} - \overline{\boldsymbol{D}})^{-1}\overline{\boldsymbol{B}}\boldsymbol{z}(0;s) = (\boldsymbol{I} - \overline{\boldsymbol{D}})^{-1}\overline{\boldsymbol{B}}\boldsymbol{z}(\infty;s). \tag{3}$$

- Our idea is to reveal the network structure using (3) —



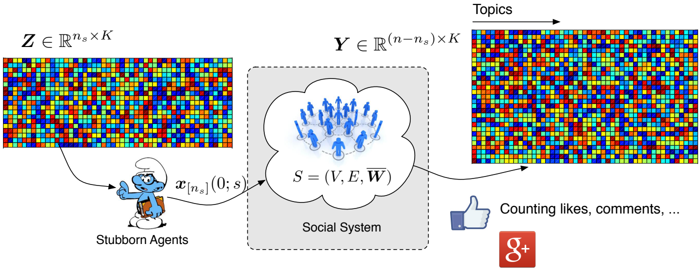
Active sensing — *Suppose* that we can estimate the opinion accurately.

- Excitation $\rightarrow$ Stubborn agents' initial opinion –

$$\boldsymbol{Z} = \begin{bmatrix} \boldsymbol{z}(0;1) & \boldsymbol{z}(0;2) & \ldots & \boldsymbol{z}(0;K) \end{bmatrix}$$

- Output $\rightarrow$ Non-stubborn agents' steady state opinion –

$$\boldsymbol{Y} = \mathbb{E}\big[\begin{bmatrix} \boldsymbol{y}(\infty;1) & \boldsymbol{y}(\infty;2) & \ldots & \boldsymbol{y}(\infty;K) \end{bmatrix}\big].$$



$$\boldsymbol{Y} = (\boldsymbol{I} - \overline{\boldsymbol{D}})^{-1}\overline{\boldsymbol{B}}\boldsymbol{Z}$$

Estimating the opinions online

Issue: We do not have access to $\mathbb{E}[\boldsymbol{y}(\infty; s)], \boldsymbol{z}(0; s)$ in general.

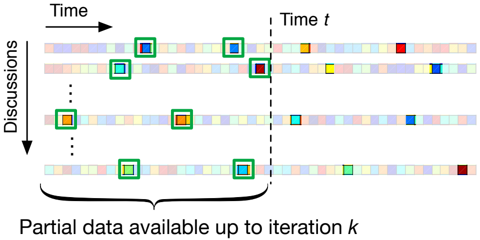
- At time t and topic s, we observe:

$$\hat{\boldsymbol{x}}(t; s) = \boldsymbol{x}(t; s) + \boldsymbol{n}(t; s), \tag{4}$$

  where $\boldsymbol{n}(t; s)$ is i.i.d. noise and $\hat{\boldsymbol{y}}(t; s), \hat{\boldsymbol{z}}(t; s)$ are defined similarly.
- For topic s, we have the opinion data $\{\hat{\boldsymbol{x}}(t; s)\}_{t \in \mathcal{T}_{k,s}}$, where $\mathcal{T}_{k,s} \subset \mathbb{N}$ is the sampling set up to the kth iteration.
- Estimate $\mathbb{E}[\boldsymbol{y}(\infty; s)], \boldsymbol{z}(0; s)$ by *temporal averaging*:

$$\boldsymbol{z}_s^k \triangleq \frac{1}{|\mathcal{T}_{k,s}|} \sum_{t \in \mathcal{T}_{k,s}} \hat{\boldsymbol{z}}(t; s) \approx \boldsymbol{z}(0; s), \quad \boldsymbol{y}_s^k \triangleq \frac{1}{|\mathcal{T}_{k,s}|} \sum_{t \in \mathcal{T}_{k,s}} \hat{\boldsymbol{y}}(t; s) \approx \mathbb{E}[\boldsymbol{z}(\infty; s)].$$

- Can be done in an *online* fashion.



$$\boldsymbol{z}_s^k \triangleq \frac{1}{|\mathcal{T}_{k,s}|} \sum_{t \in \mathcal{T}_{k,s}} \hat{\boldsymbol{z}}(t;s) \approx \boldsymbol{z}(0;s), \quad \boldsymbol{y}_s^k \triangleq \frac{1}{|\mathcal{T}_{k,s}|} \sum_{t \in \mathcal{T}_{k,s}} \hat{\boldsymbol{y}}(t;s) \approx \mathbb{E}[\boldsymbol{z}(\infty;s)].$$

Proposition 1 [WSL16]

Let $t_o = \min_{t \in \mathcal{T}_{k,s}} t$, $t_o \to \infty$, and $|\mathcal{T}_{k,s}| \to \infty$ as $k \to \infty$, then $(\boldsymbol{y}_s^k, \boldsymbol{z}_s^k) \to (\mathbb{E}[\boldsymbol{y}(\infty;s)], \boldsymbol{z}(0;s))$ in the mean square sense.

- ✓ — the estimates $\boldsymbol{z}_s^k$, $\boldsymbol{y}_s^k$ converge.

Regression problem

- Let $\mathcal{I}_{\mathcal{F}}(\boldsymbol{B},\boldsymbol{D})$ be an indicator function for the feasible $(\boldsymbol{B},\boldsymbol{D})$, $\gamma,\lambda > 0$

$$f_k(\boldsymbol{B},\boldsymbol{D}) = \underbrace{\textstyle\sum_{s=1}^{K} \|\boldsymbol{y}_s^k - \boldsymbol{D}\boldsymbol{y}_s^k - \boldsymbol{B}\boldsymbol{z}_s^k\|_2^2}_{\boldsymbol{Y}=(\mathbf{I}-\overline{\boldsymbol{D}})^{-1}\overline{\boldsymbol{B}}\boldsymbol{Z}} + \gamma \underbrace{\|\boldsymbol{D}\mathbf{1} + \boldsymbol{B}\mathbf{1} - \mathbf{1}\|_2^2}_{\overline{\boldsymbol{B}}\mathbf{1}+\overline{\boldsymbol{D}}\mathbf{1}=\mathbf{1}},$$

$$h(\boldsymbol{B},\boldsymbol{D}) = \lambda \underbrace{\|\mathrm{vec}(\boldsymbol{D})\|_1}_{\overline{\boldsymbol{D}} \text{ is sparse}} + \mathcal{I}_{\mathcal{F}}(\boldsymbol{B},\boldsymbol{D}).$$

Active sensing of Social Networks

To identify $(\overline{\boldsymbol{B}},\overline{\boldsymbol{D}})$, we solve

$$\min_{\boldsymbol{B},\boldsymbol{D}} \; f_k(\boldsymbol{B},\boldsymbol{D}) + h(\boldsymbol{B},\boldsymbol{D}) \qquad (\mathrm{P1}_k)$$

- $\Longrightarrow$ $(\mathrm{P1}_\infty)$ can identify the true $\overline{\boldsymbol{B}},\overline{\boldsymbol{D}}$; see [WSL16].
- Solving $(\mathrm{P1}_k)$ at $k \to \infty$? ✗ need a huge amount of data. For each k? ✗ high complexity.

Online algorithm for (P1$_k$)

- *Stochastic proximal gradient (SPG)* — at iteration k,

$$(\boldsymbol{B}^{k+1}, \boldsymbol{D}^{k+1}) = \mathrm{prox}_{\alpha h}\big((\boldsymbol{B}^{k+1} - \alpha \boldsymbol{gB}^k, \boldsymbol{D}^{k+1} - \alpha \boldsymbol{gD}^k)\big), \qquad (5)$$

  where $\alpha > 0$ is a step size and

$$\boldsymbol{gB}^k = \nabla_{\boldsymbol{B}} f_k(\boldsymbol{B}^k, \boldsymbol{D}^k), \quad \boldsymbol{gD}^k = \nabla_{\boldsymbol{D}} f_k(\boldsymbol{B}^k, \boldsymbol{D}^k)$$

  Notice that the proximal operator can be carried out by a soft-thresholding operation $\Longrightarrow$ ✓ low computational complexity.

- ✓ as $f_k(\cdot,\cdot)$ is known at iteration k, (5) can be implemented *online*!
- The gradients are *inexact* $\Longrightarrow$ does it converge?

- Re-expressing the gradient estimates:

$$\boldsymbol{gB}^k = \nabla_{\boldsymbol{B}} f_\infty(\boldsymbol{B}^k, \boldsymbol{D}^k) + \boldsymbol{\eta}_B^k, \quad \boldsymbol{gD}^k = \nabla_{\boldsymbol{D}} f_\infty(\boldsymbol{B}^k, \boldsymbol{D}^k) + \boldsymbol{\eta}_B^k$$

Theorem 1 [AFM14]

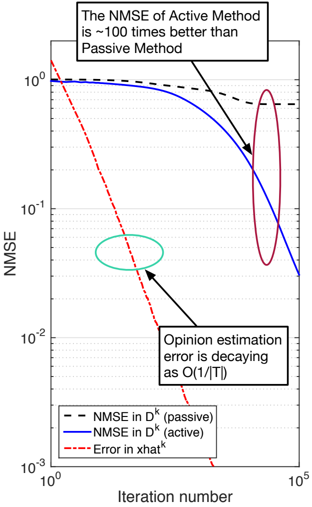
Let $\boldsymbol{\eta}^k = (\boldsymbol{\eta}_B^k, \boldsymbol{\eta}_D^k)$. If $\limsup_{k\to\infty} \|\boldsymbol{\eta}^k\| < \infty$ and $\lim_{k\to\infty} \boldsymbol{\eta}^k = 0$ almost surely (a.s.), then $\lim_{k\to\infty}(\boldsymbol{B}^k, \boldsymbol{D}^k) = (\boldsymbol{B}^\star, \boldsymbol{D}^\star)$ a.s., where $(\boldsymbol{B}^\star, \boldsymbol{D}^\star)$ is an optimal solution to $(\mathrm{P1}_\infty)$.

Proposition 2 [this paper]

The online estimators $\boldsymbol{z}_s^k, \boldsymbol{y}_s^k$ converge to their respective expected values $\boldsymbol{z}(0; s)$, $\mathbb{E}[\boldsymbol{y}(\infty; s)]$ almost surely if $k \to \infty$.

The SPG method for $(\mathrm{P1}_k)$ *converges almost surely.*

Evolution of NMSE in estimating $\overline{\boldsymbol{D}}$



- **Dimension** — $n - n_s = 100$ normal agents, $n_s = 36$ stubborn agents, $K = 72$ topics.
- **Benchmark** — "passive" method: solve (P0) using the same set of data available.
- *Graph G* — ER with $p = 0.05$ and uniform dist. weights on $\overline{\boldsymbol{W}}$.
- *SPG parameter* — $\gamma = 0.1, \lambda = 10^{-10}, \alpha = 0.01$.
- *Batch processing* — an SPG iteration is run when $5K$ new opinion samples are collected.
- *Sampling set* — $t \in \mathcal{T}_{k,s}$, $t \sim \{10^3, ..., 10^7\}$, $|\mathcal{T}_{k,s}| = 5 \times 10^5$.
- We defined $NMSE = \|\boldsymbol{D}^k - \overline{\boldsymbol{D}}\|_F^2 / \|\overline{\boldsymbol{D}}\|_F^2$.

Evolution of the error distribution in $\overline{\boldsymbol{D}}$

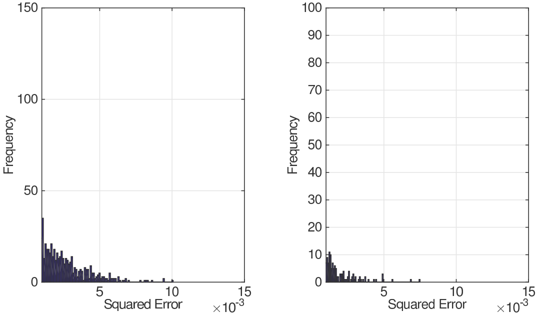
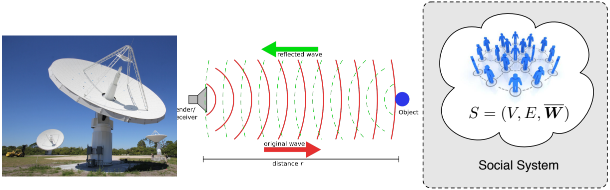


Fig. Histogram of the squared error in $\boldsymbol{D}$, i.e., $(D_{ij}^k - \bar{D}_{ij})^2$. (Left) after 5×10^3 SPG iterations. (Right) after 50×10^3 iterations.

- The error distribution becomes concentrated at zero as more opinion data are being collected.

Conclusions



- Derived an active online learning algorithm for trusts in social networks.
- Demonstrated both empirically and analytically that the online algorithm converges.

- **Future works:** combining the online algorithm with public data collected from online social networks, faster convergence rate using Nesterov accelerated gradient methods.

Questions?

Check out `http://arxiv.org/abs/1601.05834` for [WSL16].

[AFM14] Y. F. Atchade, G. Fort, and E. Moulines.
On stochastic proximal gradient algorithms.
ArXiv e-prints, February 2014.

[DBB+14] Abir De, Sourangshu Bhattacharya, Parantapa Bhattacharya, Niloy Ganguly, and Soumen Chakrabarti.
Learning a Linear Influence Model from Transient Opinion Dynamics.
CIKM '14, pages 401–410, 2014.

[DeG74] M.H. DeGroot.
Reaching a consensus.
In *Journal of American Statistcal Association*, volume 69, pages 118–121, 1974.

[HK02] R. Hegselmann and U. Krause.
Opinion dynamics and bounded confidence models, analysis and simulations.
In *Journal of Artificial Societies and Social Simulation*, volume 5, 2002.

[HL75] Richard A Holley and Thomas M Liggett.
Ergodic theorems for weakly interacting infinite systems and the voter model.
The annals of probability, pages 643–663, 1975.

[Tim07] Marc Timme.
Revealing network connectivity from response dynamics.
Physical Review Letters, 98(22):1–4, 2007.

[WLGY11] Wen-Xu Wang, Ying-Cheng Lai, Celso Grebogi, and Jieping Ye.
Network Reconstruction Based on Evolutionary-Game Data via Compressive Sensing.
Physical Review X, 1(2):1–7, 2011.

[WSL16] Hoi-To Wai, Anna Scaglione, and Amir Leshem.
Active Sensing of Social Networks.
IEEE Trans. Sig. and Inf. Proc. over Networks, March 2016.

Proposed algorithm for active online learning of social network

1: **Initialize:** $(\boldsymbol{B}^0, \boldsymbol{D}^0) \in \mathcal{F}$, $k = 1$;
2: **while** *convergence is not reached* **do**
3: Observe new opinion samples $\{\hat{\boldsymbol{y}}(t;s), \hat{\boldsymbol{z}}(t;s)\}_{t \in \mathcal{T}_{k,s} \setminus \mathcal{T}_{k-1,s}}$ and update the estimators $\hat{\boldsymbol{y}}_s^k, \hat{\boldsymbol{z}}_s^k$ accordingly.
4: Compute the gradient $\boldsymbol{gD}^k, \boldsymbol{gB}^k$ using $\hat{\boldsymbol{y}}_s^k, \hat{\boldsymbol{z}}_s^k$.
5: Perform the proximal gradient updates:

$$(\boldsymbol{B}^{k+1}, \boldsymbol{D}^{k+1}) \leftarrow \mathrm{prox}_{\alpha h}\big((\boldsymbol{B}^{k+1} - \alpha \boldsymbol{gB}^k, \boldsymbol{D}^{k+1} - \alpha \boldsymbol{gD}^k)\big)$$

6: $k \leftarrow k + 1$.
7: **end while**
8: **Return:** $(\boldsymbol{B}^k, \boldsymbol{D}^k)$.