DECENTRALIZED ESTIMATION AND OPTIMIZATION OF PAIRWISE FUNCTIONS

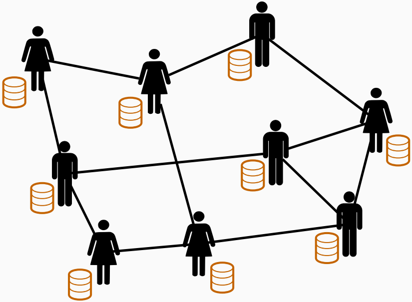
Aurélien Bellet (Inria MAGNET)

Joint work with I. Colin, J. Salmon and S. Clémençon (Télécom ParisTech)

Séminaire SIGMA, École Centrale de Lille, 30/01/2017

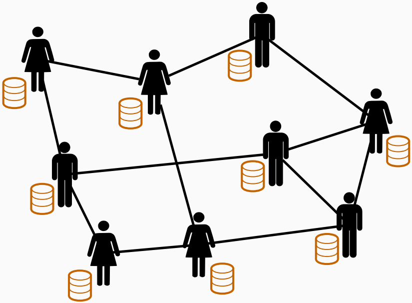
INTRODUCTION

DECENTRALIZED DATA NETWORKS



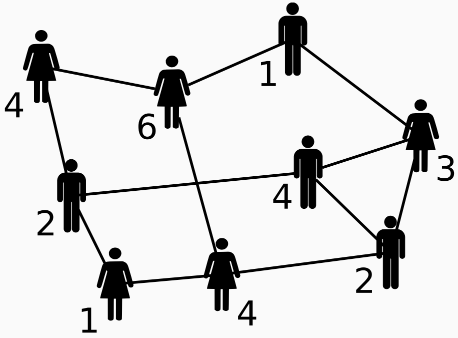
- A set of n agents with local data (agent i holds $x_i \in \mathcal{X}$)
- A communication network (connected graph)
- **Goal:** compute or optimize a global function of the data
- Some use-cases:
  - Estimation and optimization in sensor networks, IoT
  - Collaborative peer-to-peer machine learning (no third party)

KEY PRINCIPLE: RANDOMIZED GOSSIP ALGORITHM



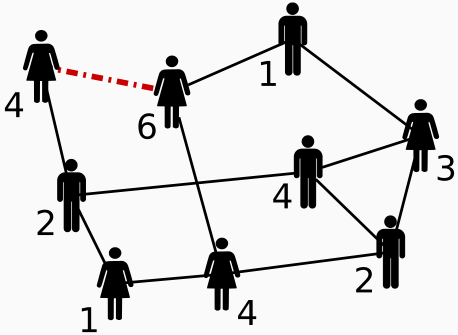
- Users wake up independently and asynchronously, select a random neighbor and exchange information
  - Equivalent view: at each step, activate a random network edge
- Simple and asynchronous $\rightarrow$ well suited to large networks

- **Goal:** compute the network average $\frac{1}{n}\sum_{i=1}^{n} f(x_i)$ [Boyd et al., 2006]



- Convergence rate of $O(e^{-C_G t})$ with C_G proportional to the spectral gap of the network

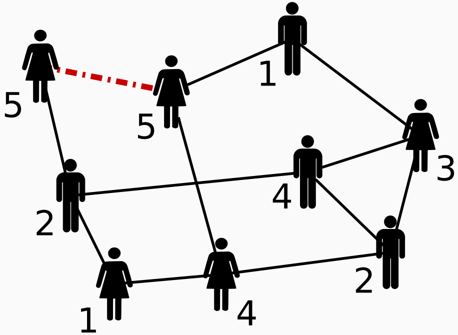
- **Goal:** compute the network average $\frac{1}{n}\sum_{i=1}^{n} f(x_i)$ [Boyd et al., 2006]



- Convergence rate of $O(e^{-C_G t})$ with C_G proportional to the spectral gap of the network

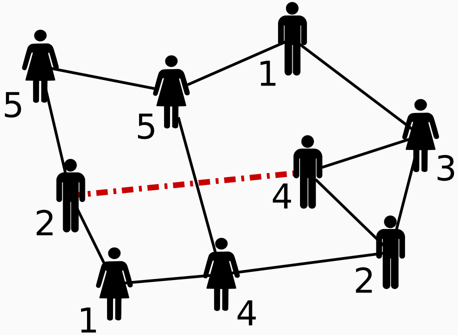
RANDOMIZED GOSSIP FOR AVERAGING

- **Goal:** compute the network average $\frac{1}{n}\sum_{i=1}^{n} f(x_i)$ [Boyd et al., 2006]



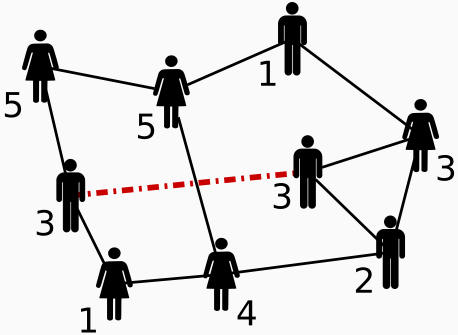
- Convergence rate of $O(e^{-C_G t})$ with C_G proportional to the spectral gap of the network

- **Goal:** compute the network average $\frac{1}{n}\sum_{i=1}^{n} f(x_i)$ [Boyd et al., 2006]



- Convergence rate of $O(e^{-C_G t})$ with C_G proportional to the spectral gap of the network

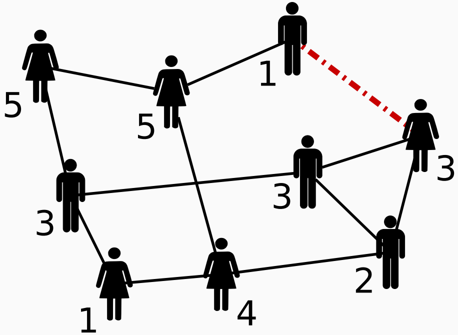
- **Goal:** compute the network average $\frac{1}{n}\sum_{i=1}^{n} f(x_i)$ [Boyd et al., 2006]



- Convergence rate of $O(e^{-C_G t})$ with C_G proportional to the spectral gap of the network

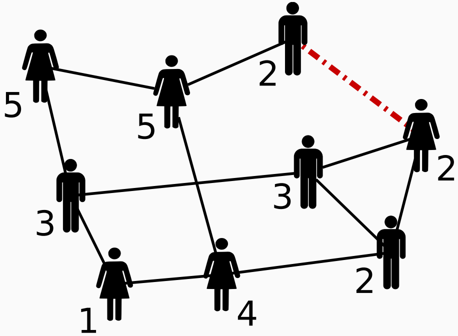
RANDOMIZED GOSSIP FOR AVERAGING

- **Goal:** compute the network average $\frac{1}{n}\sum_{i=1}^{n} f(x_i)$ [Boyd et al., 2006]



- Convergence rate of $O(e^{-C_G t})$ with C_G proportional to the spectral gap of the network

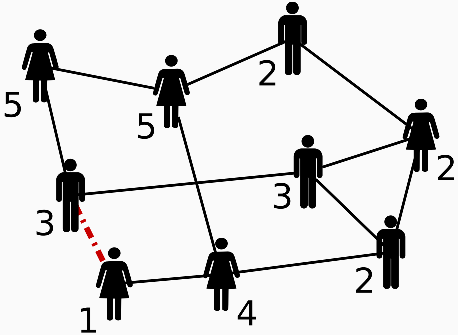
- **Goal:** compute the network average $\frac{1}{n}\sum_{i=1}^{n} f(x_i)$ [Boyd et al., 2006]



- Convergence rate of $O(e^{-C_G t})$ with C_G proportional to the spectral gap of the network

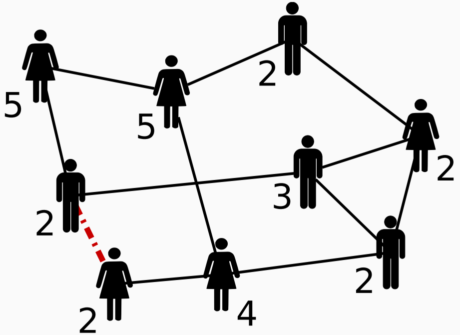
RANDOMIZED GOSSIP FOR AVERAGING

- **Goal:** compute the network average $\frac{1}{n}\sum_{i=1}^{n} f(x_i)$ [Boyd et al., 2006]



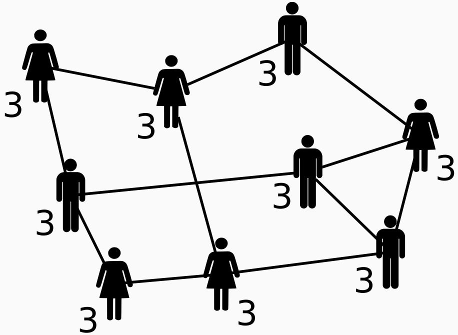
- Convergence rate of $O(e^{-C_G t})$ with C_G proportional to the spectral gap of the network

- **Goal:** compute the network average $\frac{1}{n}\sum_{i=1}^{n} f(x_i)$ [Boyd et al., 2006]



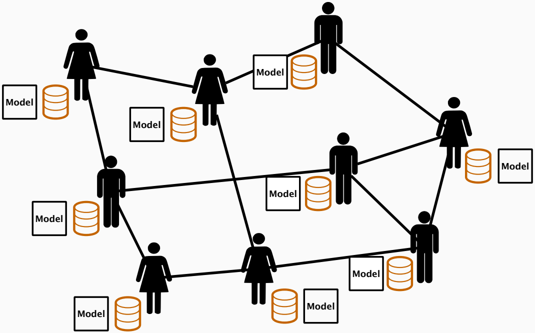
- Convergence rate of $O(e^{-C_G t})$ with C_G proportional to the spectral gap of the network

- **Goal:** compute the network average $\frac{1}{n}\sum_{i=1}^{n} f(x_i)$ [Boyd et al., 2006]



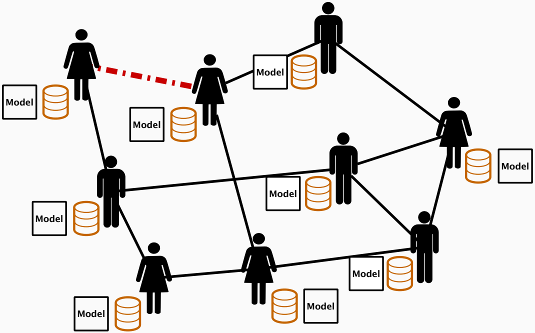
- Convergence rate of $O(e^{-C_G t})$ with C_G proportional to the spectral gap of the network

- **Goal:** solve $\min_{\theta\in\mathbb{R}^d} \frac{1}{n}\sum_{i=1}^{n} f(\theta; x_i)$ with f convex [Nedic and Ozdaglar, 2009, Duchi et al., 2012, Wei and Ozdaglar, 2012]



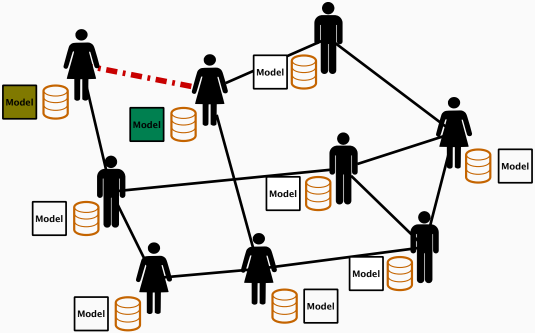
- Sublinear convergence rates: $O(1/t)$ or $O(1/\sqrt{t})$

- **Goal:** solve $\min_{\theta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} f(\theta; x_i)$ with f convex
[Nedic and Ozdaglar, 2009, Duchi et al., 2012, Wei and Ozdaglar, 2012]



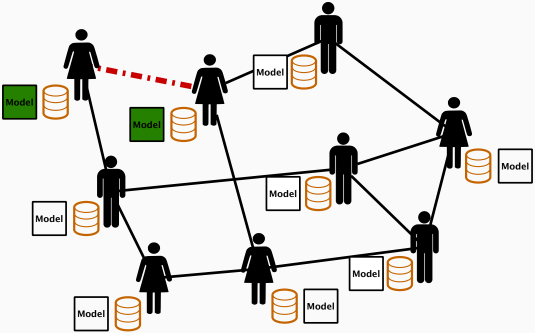
- Sublinear convergence rates: $O(1/t)$ or $O(1/\sqrt{t})$

- **Goal:** solve $\min_{\theta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} f(\theta; x_i)$ with f convex
  [Nedic and Ozdaglar, 2009, Duchi et al., 2012, Wei and Ozdaglar, 2012]



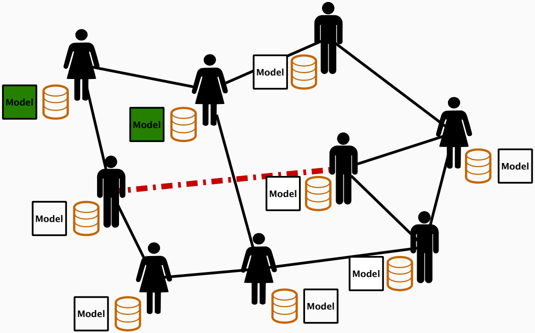
- Sublinear convergence rates: $O(1/t)$ or $O(1/\sqrt{t})$

- **Goal:** solve $\min_{\theta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} f(\theta; x_i)$ with f convex
  [Nedic and Ozdaglar, 2009, Duchi et al., 2012, Wei and Ozdaglar, 2012]



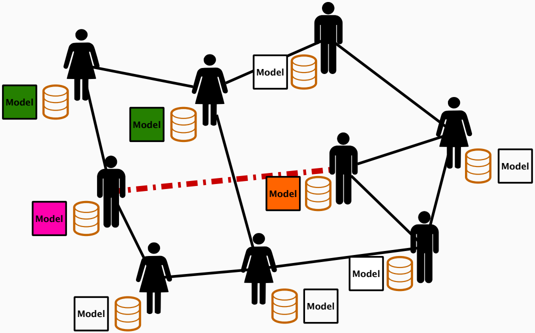
- Sublinear convergence rates: $O(1/t)$ or $O(1/\sqrt{t})$

- **Goal:** solve $\min_{\theta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} f(\theta; x_i)$ with f convex
  [Nedic and Ozdaglar, 2009, Duchi et al., 2012, Wei and Ozdaglar, 2012]



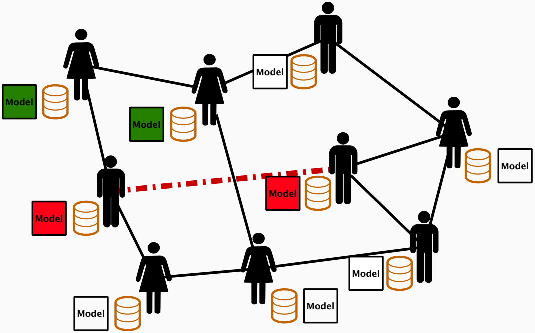
- Sublinear convergence rates: $O(1/t)$ or $O(1/\sqrt{t})$

- **Goal:** solve $\min_{\theta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} f(\theta; x_i)$ with f convex
  [Nedic and Ozdaglar, 2009, Duchi et al., 2012, Wei and Ozdaglar, 2012]



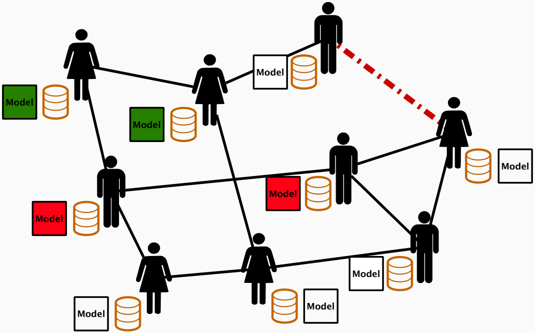
- Sublinear convergence rates: $O(1/t)$ or $O(1/\sqrt{t})$

- **Goal:** solve $\min_{\theta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} f(\theta; x_i)$ with f convex
  [Nedic and Ozdaglar, 2009, Duchi et al., 2012, Wei and Ozdaglar, 2012]



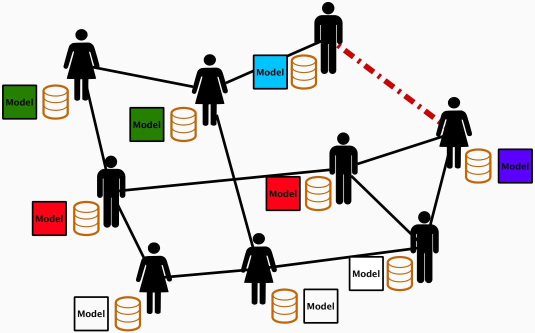
- Sublinear convergence rates: $O(1/t)$ or $O(1/\sqrt{t})$

- **Goal:** solve $\min_{\theta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} f(\theta; x_i)$ with f convex
  [Nedic and Ozdaglar, 2009, Duchi et al., 2012, Wei and Ozdaglar, 2012]



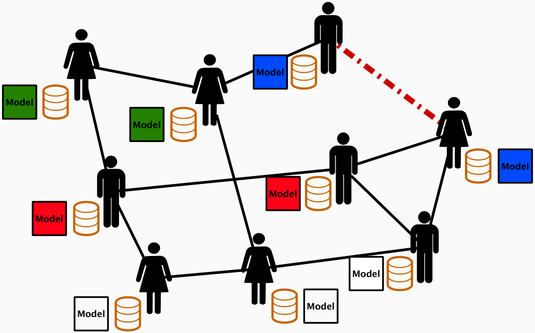
- Sublinear convergence rates: $O(1/t)$ or $O(1/\sqrt{t})$

- **Goal:** solve $\min_{\theta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} f(\theta; x_i)$ with f convex
  [Nedic and Ozdaglar, 2009, Duchi et al., 2012, Wei and Ozdaglar, 2012]



- Sublinear convergence rates: $O(1/t)$ or $O(1/\sqrt{t})$

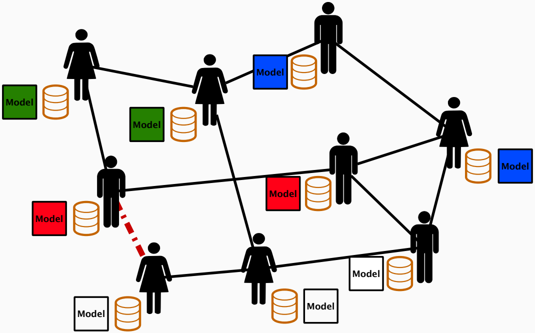
- **Goal:** solve $\min_{\theta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} f(\theta; x_i)$ with f convex
  [Nedic and Ozdaglar, 2009, Duchi et al., 2012, Wei and Ozdaglar, 2012]



- Sublinear convergence rates: $O(1/t)$ or $O(1/\sqrt{t})$

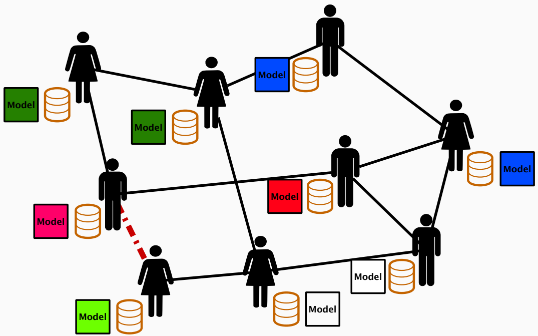
RANDOMIZED GOSSIP FOR OPTIMIZATION

- **Goal:** solve $\min_{\theta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} f(\theta; x_i)$ with f convex
  [Nedic and Ozdaglar, 2009, Duchi et al., 2012, Wei and Ozdaglar, 2012]



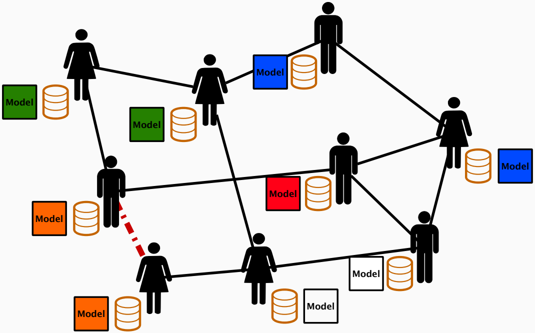
- Sublinear convergence rates: $O(1/t)$ or $O(1/\sqrt{t})$

- **Goal:** solve $\min_{\theta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} f(\theta; x_i)$ with f convex
  [Nedic and Ozdaglar, 2009, Duchi et al., 2012, Wei and Ozdaglar, 2012]



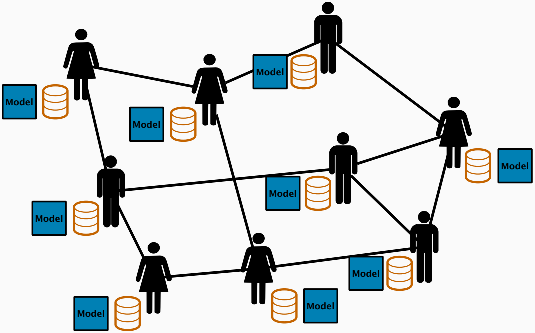
- Sublinear convergence rates: $O(1/t)$ or $O(1/\sqrt{t})$

- **Goal:** solve $\min_{\theta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} f(\theta; x_i)$ with f convex
  [Nedic and Ozdaglar, 2009, Duchi et al., 2012, Wei and Ozdaglar, 2012]



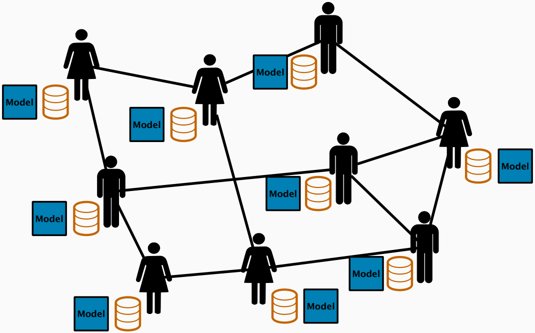
- Sublinear convergence rates: $O(1/t)$ or $O(1/\sqrt{t})$

- **Goal:** solve $\min_{\theta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} f(\theta; x_i)$ with f convex
  [Nedic and Ozdaglar, 2009, Duchi et al., 2012, Wei and Ozdaglar, 2012]



- Sublinear convergence rates: $O(1/t)$ or $O(1/\sqrt{t})$

- **Goal:** solve $\min_{\theta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} f(\theta; x_i)$ with f convex
  [Nedic and Ozdaglar, 2009, Duchi et al., 2012, Wei and Ozdaglar, 2012]



- Sublinear convergence rates: $O(1/t)$ or $O(1/\sqrt{t})$

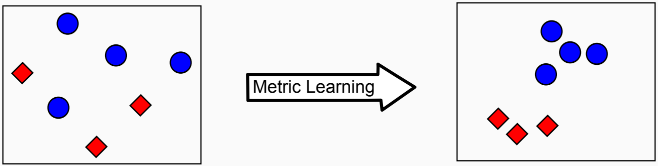
HOW ABOUT PAIRWISE FUNCTIONS?

- These algorithms do **not** generalize to pairwise functions
- Cannot compute sample statistics of the form $\frac{1}{n^2}\sum_{i,j=1}^n f(x_i, x_j)$
  - Sample variance: $f(x, x') = (x - x')^2/2$
  - Average distance: $f(x, x') = \|x - x'\|$
  - Other U-statistics: Kendall's τ, Wilcoxon-Mann-Whitney test...
- Machine learning: cannot solve Empirical Risk Minimization (ERM) problems of the form

$$\min_{\theta\in\mathbb{R}^d} \frac{1}{n^2}\sum_{i,j=1}^n f(\theta; x_i, x_j)$$

  - Metric learning, bipartite ranking, clustering, graph inference...

EXAMPLE 1: METRIC LEARNING



- Labeled data: $(x_i, \ell_i) \in \mathcal{X} \times \{1, \ldots, C\}$
- Learn distance measure adapted to the task [Bellet et al., 2015]
- Distance function $D : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_+$
- Empirical risk measure associated with D:

$$\frac{1}{n^2} \sum_{i,j=1}^{n} \Phi \left((1 - D(x_i, x_j))(2\mathbb{I}\{\ell_i = \ell_j\} - 1) \right)$$

(Φ convex surrogate of the 0-1 loss)

EXAMPLE 2: LEARNING TO RANK

- Labeled data: $(x_i, \ell_i) \in \mathcal{X} \times \{-1, 1\}$
- Learn to rank items (e.g., relevant vs irrelevant)
- Scoring function $s : \mathcal{X} \to \mathbb{R}$
- Empirical risk measures associated with s [Zhao et al., 2011]:

$$\frac{1}{n^2} \sum_{i,j=1}^{n} \mathbb{I}\{\ell_i > \ell_j\} \Phi \left((s(x_j) - s(x_i) \right)$$

- Known as the Area Under the ROC Curve (AUC)

DECENTRALIZED ESTIMATION

- Data points $x_1, \dots, x_n \in \mathcal{X}$
- Network represented as a connected graph $G = (V, E)$
  - Nodes $V = [n] = \{1, \dots, n\}$
  - Node i holds point x_i
  - $(i, j) \in E$: i and j can exchange information directly
- **Goal**: estimate pairwise statistic

$$U(f) = \frac{1}{n^2} \sum_{i,j=1}^{n} f(x_i, x_j)$$

SYNCHRONOUS VS. ASYNCHRONOUS

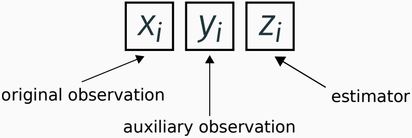
- Synchronous algorithm
  - Global clock ticking at the times of a rate 1 Poisson process
  - Each time the clock ticks, all nodes activate
- Asynchronous algorithm
  - Each node has a local clock
  - Each time a node's clock ticks, it activates
  - For modeling purposes: equivalent to a single Poisson clock ticking at rate n with random selection of node to activate

- Observe that we can write

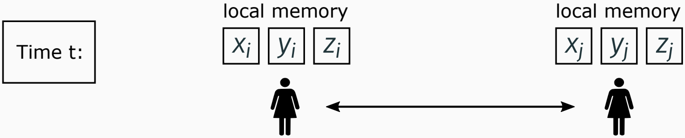
$$U(f) = \frac{1}{n}\sum_{i=1}^{n}\overline{f}_i, \quad \text{with } \overline{f}_i = \frac{1}{n}\sum_{j=1}^{n} f(x_i, x_j)$$

- Key difference with standard averaging: each “local value” $\overline{f}_i$ depends on the entire dataset
- Our algorithms will combine two steps at each iteration:
  - Data propagation step so that each node i can estimate $\overline{f}_i$
  - Averaging step to ensure global convergence to $U(f)$

- Each node stores an auxiliary observation and an estimate



- At time t, an edge $(i, j) \in E$ is activated



mix estimates: $z_i \leftarrow \frac{z_i+z_j}{2}$ $\qquad$ $z_j \leftarrow \frac{z_i+z_j}{2}$

update: $z_i \leftarrow (1-\alpha_t)z_i + \alpha_t f(x_i, y_i)$ $\qquad$ $z_j \leftarrow (1-\alpha_t)z_j + \alpha_t f(x_j, y_j)$

swap auxiliary data: $y_i \leftarrow y_j$ $\qquad$ $y_j \leftarrow y_i$

- Need a global clock

Algorithm 1 GoSta-sync

Require: Each node k holds x_k

Each node k initializes $y_k = x_k$ and $z_k = 0$
for $t = 1, 2, \dots$ **do**
  for $k = 1, \dots, n$ **do**
    Set $z_k \leftarrow \frac{t-1}{t} z_k + \frac{1}{t} f(x_k, y_k)$
  end for
  Draw (i, j) uniformly at random from E
  Set $z_i, z_j \leftarrow \frac{1}{2}(z_i + z_j)$
  Swap auxiliary observations: $y_i \leftrightarrow y_j$
end for

convergence analysis (synchronous)

Theorem (Synchronous setting, [Colin et al., 2015])

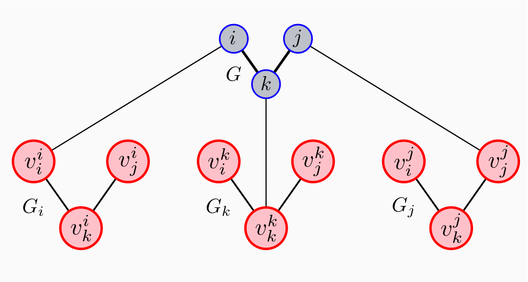
If $G = (V, E)$ is connected and non-bipartite, then for any $t > 0$:

$$\|\mathbb{E}[\mathbf{z}(t)] - U(f)\mathbf{1}_n\| \leq \frac{1}{C_G t}\left\|\bar{\mathbf{f}} - U(f)\mathbf{1}_n\right\| + \left(\frac{2}{C_G t} + e^{-C_G t}\right)\left\|\mathbf{F} - \bar{\mathbf{f}}\mathbf{1}_n^\top\right\|,$$

where $C_G = \beta_{n-1}/|E|$, β_{n-1} is the spectral gap of G, $\bar{\mathbf{f}} = (\bar{f}_i)_{1\leq i\leq n}$ and $\mathbf{F} \in \mathbb{R}^{n\times n}$ s.t. $\mathbf{F}_{ij} = f(x_i, x_j)$.

- Data-dependent terms: quantify difficulty of estimation problem
  - Dispersion measure between the values to be averaged
- Network-dependent terms: quantify how well things propagate
  - Graphs with larger spectral gap $\rightarrow$ better connectivity [Chung, 1997]

PROOF IDEA: PHANTOM NETWORKS



- Equivalent reformulation of the problem to model data propagation and estimate update/averaging **separately**
- "Phantom" networks $G_1, \ldots, G_n$
  - For $k, i \in [n]$, node v_i^k initially holds $H(x_k, x_i)$
  - Data propagation: for all $k \in [n]$, nodes v_i^k and v_j^k swap values
- Original network G
  - Nodes $1, \ldots, n$ hold estimates $z_1(t), \ldots, z_n(t)$
  - To update the estimates: each node k uses the value of node v_k^k

- We can now represent the system at time t as $\mathbf{S}(t) = \begin{pmatrix} \mathbf{S}_1(t) \\ \mathbf{S}_2(t) \end{pmatrix}$
    - $\mathbf{S}_1(t) \in \mathbb{R}^n$ correspond to estimate vector $\boldsymbol{z}(t) = [z_1(t), \dots, z_n(t)]$
    - $\mathbf{S}_2(t) \in \mathbb{R}^{n^2}$ represent the data propagation in the network
- Characterize the transition matrix

$$M(t) = \begin{pmatrix} \underbrace{M_1(t)}_{\text{averaging}} & \underbrace{M_2(t)}_{\text{estimate update}} \\ 0 & \underbrace{M_3(t)}_{\text{data propagation}} \end{pmatrix} \in \mathbb{R}^{(n+n^2)\times(n+n^2)}$$

  such that $\mathbb{E}[\mathbf{S}(t+1)] = M(t)\mathbb{E}[\mathbf{S}(t)]$

- Exploit spectral structure of $M(t)$ to prove convergence of $\mathbf{S}_1(t)$

- No global clock: only selected nodes are active
- Each node i stores an unbiased estimate m_i of current iteration
  - Probability $p_i = 2d_i/|E|$ that i awakes at a given iteration
  - When i awakes, it updates $m_i \leftarrow m_i + 1/p_i$

Algorithm 2 GoSta-async

Require: Each node k holds x_k and p_k

Each node k initializes $y_k = x_k$, $z_k = 0$ and $m_k = 0$

for $t = 1, 2, \dots$ **do**

Draw (i, j) uniformly at random from E

Set $m_i \leftarrow m_i + 1/p_i$ and $m_j \leftarrow m_j + 1/p_j$

Set $z_i, z_j \leftarrow \frac{1}{2}(z_i + z_j)$

Set $z_i \leftarrow (1 - \frac{1}{p_i m_i})z_i + \frac{1}{p_i m_i} f(x_i, y_i)$

Set $z_j \leftarrow (1 - \frac{1}{p_j m_j})z_j + \frac{1}{p_j m_j} f(x_j, y_j)$

Swap auxiliary observations: $y_i \leftrightarrow y_j$

end for

Theorem (Asynchronous setting, [Colin et al., 2015])

If $G = (V, E)$ is connected and non-bipartite, then for any $t > 1$:

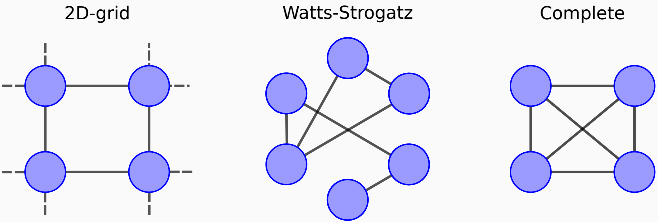
$$\|\mathbb{E}[\mathbf{z}(t)] - U(f)\mathbf{1}_n\| \leq C'_G \frac{\log t}{t} \|H\|,$$

for some constant $C'_G > 0$.

- Similar proof technique as in synchronous setting
- But time dependency of transition matrix more complex

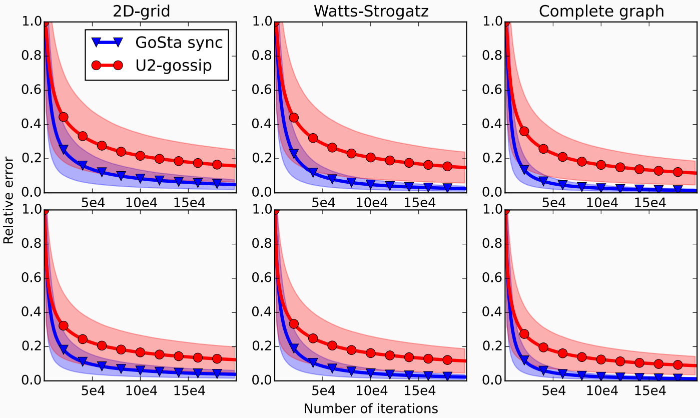
- Two estimation problems
  - Within-cluster point scatter on Wine quality dataset ($n = 1,599$)
  - Area Under the ROC Curve on SMVguide3 dataset ($n = 1,260$)

- Three types of networks



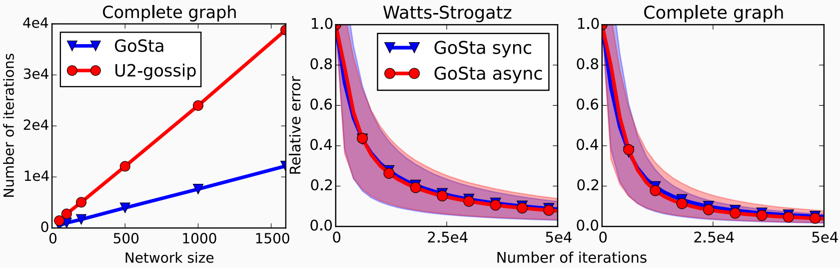
Comparison to U2-Gossip [Pelckmans and Suykens, 2009]

- U2-Gossip: propagates two observations, no averaging
- Only synchronous, worst theoretical guarantees



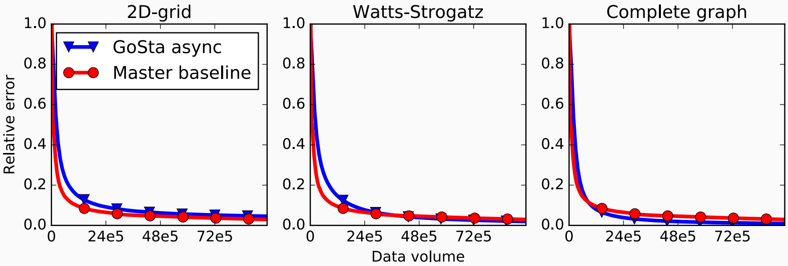
Comparison to U2-Gossip [Pelckmans and Suykens, 2009]

- GoSta scales better with n
- GoSta-sync and GoSta-async have similar performance



Comparison to “Master Node” baseline

- Baseline has access to master node connected to all nodes
- Our algorithm compensates well for lack of central node



DECENTRALIZED OPTIMIZATION

PROBLEM SETUP

- Data points $x_1, \dots, x_n \in \mathcal{X}$
- Network represented as a connected graph $G = (V, E)$
  - Nodes $V = [n] = \{1, \dots, n\}$
  - Node i holds point x_i
  - $(i, j) \in E$: i and j can exchange information directly
- **Goal**: solve regularized problem

$$\min_{\theta \in \mathbb{R}^d} \underbrace{\frac{1}{n^2} \sum_{i,j=1}^{n} f(\theta; x_i, x_j)}_{\bar{f}(\theta)} + \psi(\theta)$$

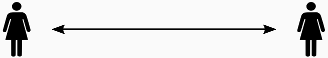
  - f convex and differentiable w.r.t. θ, L_f-Lipschitz
  - ψ convex, nonnegative, possibly nonsmooth

CENTRALIZED DUAL AVERAGING

- Two variables: primal $\theta(t)$ and "dual" $z(t)$
- Positive, non-increasing step size sequence $(\gamma(t))_{t\geq 1}$
- Dual Averaging (DA) update rule [Nesterov, 2009]
    - $z(t+1) = z(t) + g(t)$, with $g(t)$ unbiased estimate of $\nabla \bar{f}(\theta(t))$
    - $\theta(t+1) = \underbrace{\arg\min_{\theta \in \mathbb{R}^d} \left\{ -z^\top \theta + \frac{\|\theta\|^2}{2\gamma(t)} + t\psi(\theta) \right\}}_{\pi(z;\gamma(t))}$
- Convergence of average iterate in $O(1/\sqrt{t})$ with $\gamma(t) \propto 1/\sqrt{t}$
- π is a scaled version of the proximal operator of ψ: can deal with popular nonsmooth regularizers such as L_1-norm
- DA updates well suited to decentralized setting [Duchi et al., 2012]

DECENTRALIZED DUAL AVERAGING

	local memory	local memory
Time t:	$\boxed{x_i}\ \boxed{y_i}\ \boxed{z_i}\ \boxed{\theta_i}$	$\boxed{x_j}\ \boxed{y_j}\ \boxed{z_j}\ \boxed{\theta_j}$



Swap auxiliary data:	$y_i \leftarrow y_j$	$y_j \leftarrow y_i$
Average dual variables:	$z_i \leftarrow \frac{z_i + z_j}{2}$	$z_j \leftarrow \frac{z_i + z_j}{2}$
Update dual variables:	$z_i \leftarrow z_i + \nabla f(\theta_i; x_i, y_i)$	$z_j \leftarrow z_j + \nabla f(\theta_j; x_j, y_j)$
Update iterates:	$\theta_i \leftarrow \pi(z_i; \gamma(t))$	$\theta_j \leftarrow \pi(z_j; \gamma(t))$

- Let us denote $\overline{f}_i(\theta) = \frac{1}{n}\sum_{k=1}^{n} f(\theta; x_i, x_k)$
- $g_i(t) = \nabla f(\theta_i(t); x_i, y_i)$ is a biased estimate of $\nabla \overline{f}_i(\theta_i(t))$:

$$\mathbb{E}[g_i(t)] = \overline{f}_i(\theta_i(t)) + \epsilon_i(t)$$

Algorithm 3 Gossip pairwise dual averaging (synchronous)

Require: Each node k holds x_k, $(\gamma(t))_{t\geq 1} > 0$

Each node k initializes $y_k = x_k$, $z_k = \theta_k = \bar{\theta}_k = 0$

for $t = 1, 2, \ldots$ **do**

Draw (i, j) uniformly at random from E

Set $z_i, z_j \leftarrow \frac{1}{2}(z_i + z_j)$

Swap auxiliary observations: $y_i \leftrightarrow y_j$

for $k = 1, \ldots, n$ **do**

Set $z_k \leftarrow z_k + \nabla_\theta f(\theta_k; x_k, y_k)$

Set $\theta_k \leftarrow \pi(z_k; \gamma(t))$

Set $\bar{\theta}_k \leftarrow \left(1 - \frac{1}{t}\right) \bar{\theta}_k + \frac{1}{t}\theta_k$

end for

end for

convergence analysis (synchronous)

Theorem (Synchronous setting, [Colin et al., 2016])

Let G be connected and non-bipartite. Let $R(\theta) = \bar{f}(\theta) + \psi(\theta)$, $\theta^* \in \arg\min_{\theta \in \Theta} R(\theta)$ and let $(\gamma(t))_{t \geq 1}$ be such that $\gamma(t) \propto 1/\sqrt{t}$. For any $i \in [n]$ and any $t > 1$, we have:

$$\mathbb{E}[R(\bar{\theta}_i(t)) - R(\theta^*)] \leq \frac{\|\theta^*\|^2 + 2L_f^2}{2\sqrt{t}} + \frac{6L_f^2}{(1-\sqrt{\lambda})\sqrt{t}} + O\left(\frac{1}{t}\sum_{t'=1}^{t} \bar{\epsilon}(t')\right),$$

where $\lambda < 1$, $1 - \lambda = \beta_{n-1}/|E|$, β_{n-1} is the spectral gap of G and $\bar{\epsilon}(t') = \frac{1}{n}\sum_{k=1}^{n} \epsilon_k(t')$.

- First term: data dependent (same as centralized dual averaging)
- Second term: network dependent
- Third term: bias of the gradient estimates

Algorithm 4 Gossip pairwise dual averaging (asynchronous)

Require: Each node k holds x_k, $(\gamma(t))_{t\geq 1} > 0$, probabilities $(p_k)_{k\in[n]}$

Each node k initializes $y_k = x_k$, $z_k = \theta_k = \bar{\theta}_k = 0$, $m_i = 0$

for $t = 1, 2, \ldots$ **do**

Draw (i, j) uniformly at random from E

Swap auxiliary observations: $y_i \leftrightarrow y_j$

for $k \in \{i, j\}$ **do**

Set $z_k \leftarrow \frac{z_i + z_j}{2}$

Set $z_k \leftarrow \frac{1}{p_k} \nabla_\theta f(\theta_k; x_k, y_k)$

Set $m_k \leftarrow m_k + \frac{1}{p_k}$

Set $\theta_k \leftarrow \pi(z_k; \gamma(m_k))$

Set $\bar{\theta}_k \leftarrow \left(1 - \frac{1}{m_k p_k}\right) \bar{\theta}_k$

end for

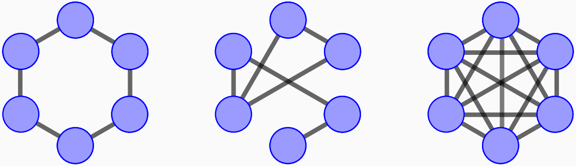
end for

- Slower convergence result of $O(t^{-1/3})$

- Task: AUC maximization with linear scoring function

$$R(\theta) = \frac{1}{n^2} \sum_{i,j=1}^{n} \mathbb{I}\{\ell_i > \ell_j\} \log\left(1 + \exp((x_j - x_i)^\top \theta)\right)$$

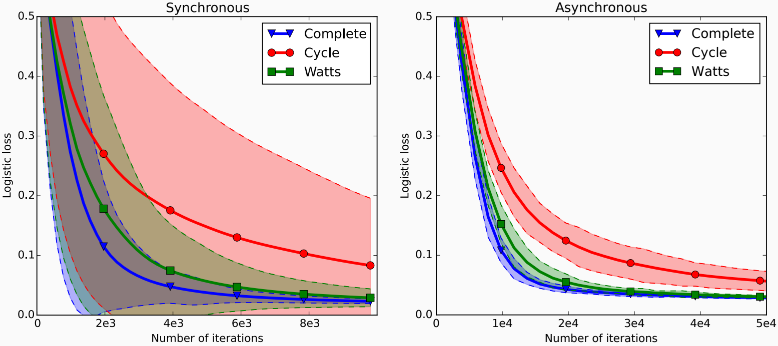
- UCI Breast Cancer dataset: $n = 699$ points in $d = 11$ dimensions
- Three types of networks



Cycle (worst case) Watts-Strogatz Complete (best case)

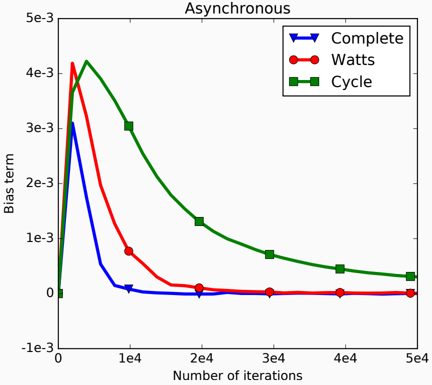
Synchronous vs. asynchronous

- Speed of convergence depends on network topology
- More variance in synchronous case: node k performs roughly $1/p_k$ gradient steps before swapping its auxiliary observation



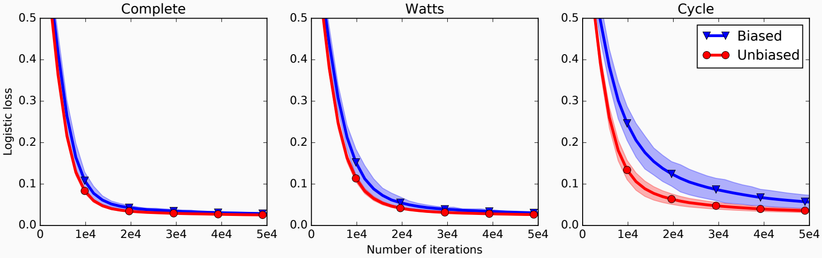
Evolution of bias term

- Vanishes quickly (also depends on spectral gap)
- Negligible: 3 orders of magnitude smaller than loss function



Comparison to oracle baseline

- Baseline has access to unbiased estimates of the gradients
- Performance is similar on reasonably-connected networks



conclusion & perspectives

Wrapping up

- Pairwise functions involved in many interesting problems
- Gossip algorithms for decentralized estimation and optimization

Looking ahead

- Personalized models [Vanhaesebrouck et al., 2017]
- Privacy, robustness to malicious users (under progress)
- Adaptive communication schemes: learn who to talk to

Thank you for your attention!

Questions?

[Bellet et al., 2015] Bellet, A., Habrard, A., and Sebban, M. (2015).
Metric Learning.
Morgan & Claypool Publishers.

[Boyd et al., 2006] Boyd, S. P., Ghosh, A., Prabhakar, B., and Shah, D. (2006).
Randomized gossip algorithms.
IEEE Transactions on Information Theory, 52(6):2508–2530.

[Chung, 1997] Chung, F. R. K. (1997).
Spectral Graph Theory, volume 92.
American Mathematical Society.

[Colin et al., 2015] Colin, I., Bellet, A., Salmon, J., and Clémençon, S. (2015).
Extending Gossip Algorithms to Distributed Estimation of U-statistics.
In NIPS.

[Colin et al., 2016] Colin, I., Bellet, A., Salmon, J., and Clémençon, S. (2016).
Gossip Dual Averaging for Decentralized Optimization of Pairwise Functions.
In ICML.

[Duchi et al., 2012] Duchi, J. C., Agarwal, A., and Wainwright, M. J. (2012).
Dual Averaging for Distributed Optimization: Convergence Analysis and Network Scaling.
IEEE Transactions on Automatic Control, 57(3):592–606.

[Nedic and Ozdaglar, 2009] Nedic, A. and Ozdaglar, A. E. (2009).
Distributed Subgradient Methods for Multi-Agent Optimization.
IEEE Transactions on Automatic Control, 54(1):48–61.

[Nesterov, 2009] Nesterov, Y. (2009).
Primal-dual subgradient methods for convex problems.
120(1):261–283.

[Pelckmans and Suykens, 2009] Pelckmans, K. and Suykens, J. (2009).
Gossip algorithms for computing u-statistics.
In IFAC Workshop on Estimation and Control of Networked Systems, pages 48–53.

[Vanhaesebrouck et al., 2017] Vanhaesebrouck, P., Bellet, A., and Tommasi, M. (2017).
Decentralized Collaborative Learning of Personalized Models over Networks.
In AISTATS.

[Wei and Ozdaglar, 2012] Wei, E. and Ozdaglar, A. E. (2012).
Distributed Alternating Direction Method of Multipliers.
In CDC, pages 5445–5450.

[Zhao et al., 2011] Zhao, P., Hoi, S. C. H., Jin, R., and Yang, T. (2011).
Online AUC Maximization.
In ICML, pages 233–240.