Learning in Heterogeneous Networks

Stefan Vlaski
work with Roula Nassif, Mert Kayaalp, Virginia Bordignon, Cédric Richard and Ali H. Sayed

Imperial College London, United Kingdom

Optimization by Quantum and Machine Learning Workshop

Imperial College London

Multi-Agent Systems

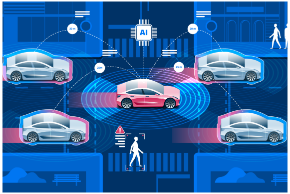


Figure: Autonomous vehicles [smartcitiesworld.net]

Figure: Social network [medium.com]

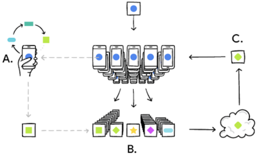
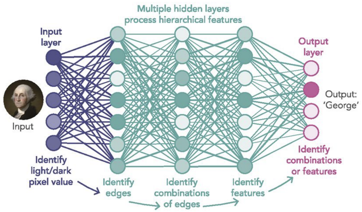


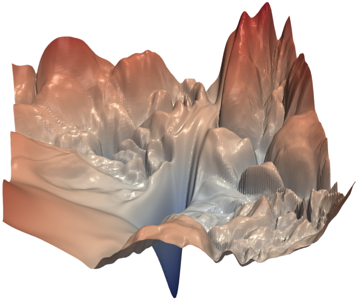
Figure: Phones [ai.googleblog.com]

Figure: Drone swarms [ft.com]

Optimisation for Learning

- Learning is building models from data.
- We want the model that fits best.
- Optimisation enables "good" learning.





Towards Provable and Efficient Learning over Networks

- Performance metrics
  - Error probability
  - Excess risk
  - Mean-square deviation
  - Convergence rate
  - First-order stationarity
  - Second-order stationarity
  - ...
- Limitations of distributed systems:
  - Network topology
  - Limited, streaming data
  - Unreliable participation
  - Noisy links
  - Quantisation
  - Privacy
  - Heterogeneity

Aim

Relate performance and limitations in a unified manner to inform design of distributed systems.

Learning Problems as Optimisation Problems

- Consider a linear regression (channel estimation) problem:

$$\boldsymbol{\gamma}_{k,i} = {w_k^o}^{\mathsf{T}} \boldsymbol{h}_{k,i} + \boldsymbol{v}_{k,i} \tag{1}$$

- We can pursue the solution via least mean-squares:

$$\arg\min_{w_k} \mathbb{E} \|\boldsymbol{\gamma}_{k,i} - {w_k}^{\mathsf{T}} \boldsymbol{h}_{k,i}\|^2 \tag{2}$$

- If the relation is non-linear, we may use a deep neural net:

$$\arg\min_{w_k} \mathbb{E} \|\boldsymbol{\gamma}_{k,i} - \sigma\left(W_{k,L} \cdot \sigma\left(W_{k,L-1} \cdot \sigma\left(W_{k,1} \boldsymbol{h}_{k,i}\right)\right)\right)\|^2 \tag{3}$$

- **Unified formulation:**

$$\arg\min_{w_k} \mathbb{E} Q(w_k, \boldsymbol{x}_k) \tag{4}$$

Learning Paradigms

- **Non-cooperative learning**:

$$w_k^o = \arg\min_{w_k} \mathbb{E} Q(w_k, \boldsymbol{x}_k) \tag{5}$$

- **Single-task learning** (i.e., consensus optimization):

$$w^o = \arg\min_{w} \sum_{k=1}^{K} p_k \mathbb{E} Q(w, \boldsymbol{x}_k) \tag{6}$$

- **Multi-task learning** over aggregate tasks $\mathcal{W} = \mathrm{col}\{w_k\}$:

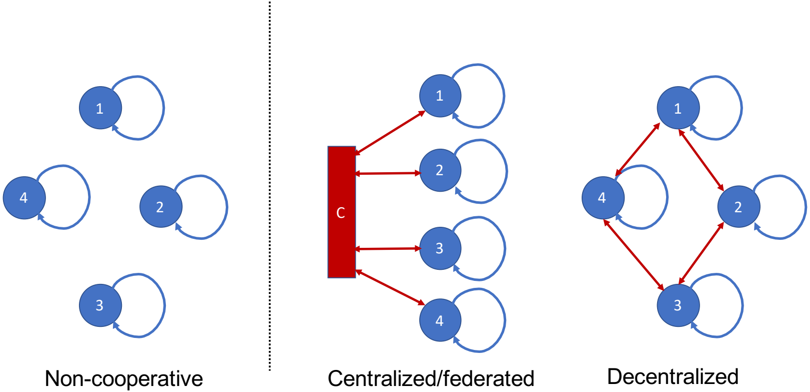
$$\mathcal{W}^o = \arg\min_{\mathcal{W}} \sum_{k=1}^{K} \mathbb{E} Q(w_k, \boldsymbol{x}_k) + \frac{\eta}{2}\mathcal{R}(\mathcal{W}) \tag{7}$$

$$\text{subject to } \mathcal{W} \in \Omega \tag{8}$$

  - $\mathcal{R}(\mathcal{W})$ and Ω encode priors on w_k.

Communication Paradigms

- Restrictions on the flow of information.



- Communication efficiency,
- robustness to failure,
- privacy.

Single- vs. Multi-Task Learning

Single-Task Objective:

$$\sum_{k=1}^{K} p_k \mathbb{E} Q(w, \boldsymbol{x}_k)$$

Multi-Task Objective:

$$\sum_{k=1}^{K} \mathbb{E} Q(w_k, \boldsymbol{x}_k) + \frac{\eta}{2}\mathcal{R}(\mathcal{W})$$

subject to $\mathcal{W} \in \Omega$

Example algorithm [Chen and Sayed '12]:

$$\boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} - \mu \widehat{\nabla J}_k(\boldsymbol{w}_{k,i-1})$$
$$\boldsymbol{w}_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \boldsymbol{\psi}_{\ell,i}$$

Generic framework:

$$\boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} - \mu \widehat{\nabla J}_k(\boldsymbol{w}_{k,i-1})$$
$$\boldsymbol{w}_{k,i} = \mathrm{Agg}\left(\{\boldsymbol{\psi}_{\ell,i}\}_{\ell \in \mathcal{N}_k}\right)$$

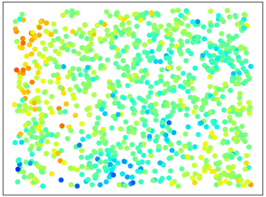
Example 1 – Regularized Multitask Learning

- Relationship prior through smoothness regularization:

$$\mathcal{W}_\eta^o = \arg\min_{\mathcal{W}} \sum_{k=1}^{K} \mathbb{E} Q(w_k; \boldsymbol{x}_k) + \frac{\eta}{2} \mathcal{W}^\top (L \otimes I) \mathcal{W} \tag{9}$$

- **Example**:

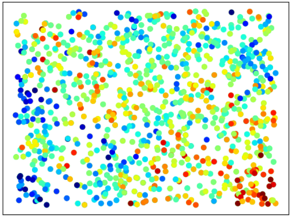
$$\mathcal{W}_\eta^o = \arg\min_{\mathcal{W}} \frac{1}{2\sigma_v^2} \sum_{k=1}^{K} \mathbb{E} \|\boldsymbol{\gamma}_k - \boldsymbol{h}_k^\mathsf{T} w_k\|^2 + \frac{\eta}{2} \mathcal{W}^\mathsf{T} (L \otimes I) \mathcal{W}$$



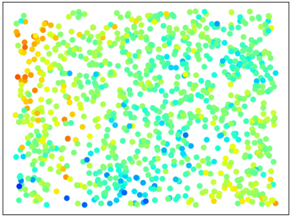
denotes the maximum aposteriori estimate for a linear model with GMRF prior:

$$\boldsymbol{\gamma}_k = \boldsymbol{h}_k^\mathsf{T} w_k^o + \boldsymbol{v}_k$$
$$f(\mathcal{W}) \triangleq (2\pi)^{-M(K-1)/2} (|\mathcal{L}|^*)^{1/2} e^{-\eta \frac{1}{2} \mathcal{W}^\mathsf{T} \mathcal{L} \mathcal{W}}$$

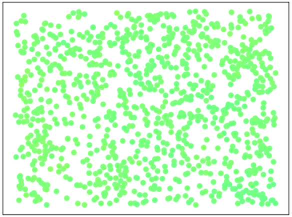
Visualizing GMRFs



$\eta = 0.001$



$\eta = 0.005$



$\eta = 1$

- Graph chosen according to Euclidean distance, η controls level of smoothness.

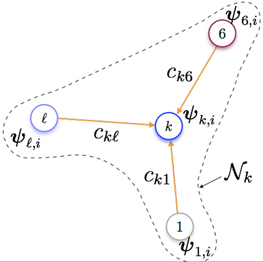
Multitask learning under smoothness

- Network global cost:

$$\boxed{\mathcal{W}^\star = \arg\min_{\mathcal{W}} \sum_{k=1}^{N} J_k(w_k) + \frac{\eta}{2}\, \mathcal{W}^\top (L \otimes I)\, \mathcal{W}}$$

- Multitask learning algorithm:

$$\begin{cases} \boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} - \mu \widehat{\nabla J_k}(\boldsymbol{w}_{k,i-1}) & \textit{(self-learning)} \\ \boldsymbol{w}_{k,i} = \left(1 - \mu\eta \sum_{\ell \in \mathcal{N}_k} c_{k\ell}\right)\boldsymbol{\psi}_{k,i} + \mu\eta \sum_{\ell \in \mathcal{N}_k} c_{k\ell}\boldsymbol{\psi}_{\ell,i} & \textit{(social learning)} \end{cases}$$

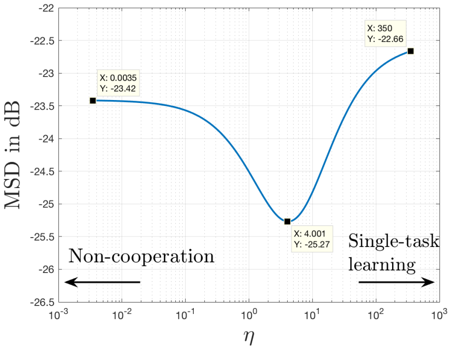


Analytical Performace Guarantee

$$\text{MSD} \approx \underbrace{\overline{\text{MSD}}}_{O(\mu),\eta} + \underbrace{\| \mathcal{W}^o_\eta - \mathcal{W}^o \|^2}_{\text{smoothness},\eta}$$

By increasing η:

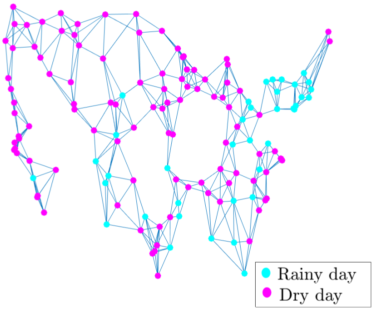
- First term is more likely to decrease
- Second term is more likely to increase and the size of this increase depends on the smoothness of $\mathcal{W}^o$



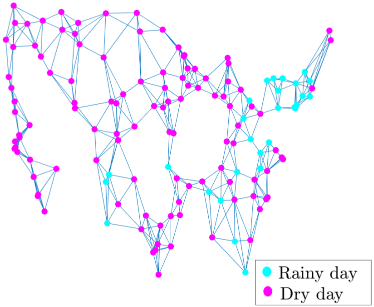
Application: Weather forecasting

Graph constructed based on geodesical distance (4-nearest neighbors).Training set $(2004 - 2012)$, test set $(2013 - 2017)$

η	0	10	45	100	1000	μ^{-1}
prediction error	0.309	0.232	**0.225**	0.226	0.228	0.232



Occurrence of rain on July 30, 2015



Prediction of rain occurrence ($\eta = 45$)

Example 2 – Subspace Prior

- An alternative model-based setting may be one where tasks are not necessarily smooth over the graph, but instead linearly related, i.e., $\mathcal{W} \in \mathrm{Range}(\mathcal{U})$ for some $\mathcal{U}$.

$$\mathcal{W}^o_{\mathcal{U}} = \arg\min_{\mathcal{W}} \quad J(\mathcal{W}) \triangleq \sum_{k=1}^{K} J_k(w_k), \quad \text{subject to} \quad \mathcal{W} \in \mathrm{Range}(\mathcal{U}), \tag{10}$$

  where $\mathrm{Range}(\cdot)$ denotes the range space operator, and $\mathcal{U}$ is an $KM \times P$ full-column rank matrix with $P \ll KM$.
- Appears naturally in many machine learning and signal processing applications.

Problem Formulation

- Network global cost ($\mathcal{U}$ full column-rank) [Nassif et al., 2020, Di Lorenzo et al., 2020]:

$$\boxed{\begin{aligned} \mathcal{W}^\star = \quad & \arg\min_{\mathcal{W}} \sum_{k=1}^{N} J_k(w_k) \\ & \text{subject to } \mathcal{W} \in \text{Range}(\mathcal{U}) \end{aligned}}$$

- Centralized stochastic gradient projection approach ($\mathcal{P}_u$: orthogonal projector onto Range($\mathcal{U}$))

$$\boldsymbol{\mathcal{W}}_i = \mathcal{P}_u \left(\boldsymbol{\mathcal{W}}_{i-1} - \mu \text{col} \left\{ \widehat{\nabla_{w_k} J_k}(\boldsymbol{w}_{k,i-1}) \right\}_{k=1}^{N} \right)$$

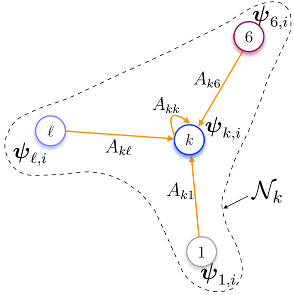
Algorithm

- To get a decentralized algorithm, replace $\mathcal{P}_u$ by an $\mathcal{A}$ such that:

$$\lim_{i\to\infty} \mathcal{A}^i = \mathcal{P}_u, \qquad A_{k\ell} = [\mathcal{A}]_{k\ell} = 0 \text{ if } \ell \notin \mathcal{N}_k$$

- Multitask learning algorithm [Nassif et al., 2020]:

$$\begin{cases} \boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} - \mu \widehat{\nabla J}_k(\boldsymbol{w}_{k,i-1}) & \textit{(self-learning)} \\ \boldsymbol{w}_{k,i} = \sum\limits_{\ell \in \mathcal{N}_k} A_{k\ell} \boldsymbol{\psi}_{\ell,i} & \textit{(social learning)} \end{cases}$$



Performance

- Network global cost:

$$\mathcal{W}^\star = \arg\min_{\mathcal{W}} \sum_{k=1}^{N} J_k(w_k) \quad \text{subject to } \mathcal{W} \in \text{Range}(\mathcal{U})$$

- Data characteristics: $H_k = \nabla^2 J_k(w_k^\star),\ R_k = \mathbb{E}[\boldsymbol{s}_{k,i}(w_k^\star)\boldsymbol{s}_{k,i}^\top(w_k^\star)]$

$$\mathcal{H} = \text{diag}\{H_1, \ldots, H_N\}, \qquad \mathcal{S} = \text{diag}\{R_1, \ldots, R_N\}$$

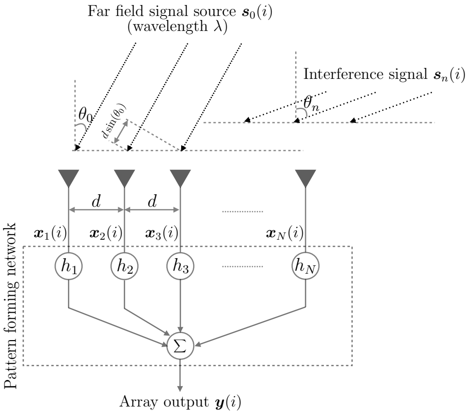
Mean-square-error w.r.t. $\mathcal{W}^\star$ (in the small adaptation regime) [Nassif et al., 2020]

$$\text{MSD} = \lim_{i\to\infty} \frac{1}{N}\mathbb{E}\|\,\mathcal{W}^\star - \boldsymbol{\mathcal{W}}_i\,\|^2 \approx \frac{\mu}{2N}\text{Tr}\left(\left(\mathcal{U}^\top\mathcal{H}\mathcal{U}\right)^{-1}\left(\mathcal{U}^\top\mathcal{S}\mathcal{U}\right)\right)$$

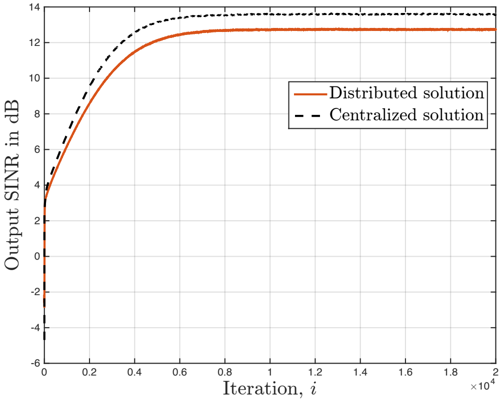
- For sufficiently small step-sizes, the decentralized strategy attains the same MSD performance as the centralized one

Application in Beamforming

- The framework is applied to an approximate linearly-constrained minimum-variance (LCMV) beamforming problem [Nassif et al., 2020, 2022]



Uniform linear array of N antennas



Comparison of output SINR

Example 3 – MAML for Multi-Agent Systems

- Instead of directly modeling the relationship between tasks w_k^o and w_ℓ^o, in model-agnostic meta-learning one assumes that both one or several (stochastic) gradient step away from a common launch-model:

$$w_k^o \approx w^o - \mu \nabla Q(w^o; \boldsymbol{x}_k) \tag{11}$$

- One then optimizes:

$$w^o \triangleq \arg\min_w \frac{1}{K} \sum_{k=1}^{K} \mathbb{E} Q(w - \mu \nabla Q(w; \boldsymbol{x}_k^1); \boldsymbol{x}_k^2) \tag{12}$$

  to determine a common launch model w^o, which adapts quickly to other tasks w_k^o via one or several (stochastic) gradient steps.

- See [Smith et al. 2017] for centralized MAML and [Fallah, Mokhtari, and Ozdaglar 2020] for federated implementation and analysis.

Diffusion-MAML: Decentralization

- If we denote:

$$\overline{Q}(w; \boldsymbol{x}_k^1, \boldsymbol{x}_k^2) \triangleq Q(w - \mu \nabla Q(w; \boldsymbol{x}_k^1); \boldsymbol{x}_k^2) \tag{13}$$

then the optimization problem

$$w^o \triangleq \arg\min_{w} \frac{1}{K} \sum_{k=1}^{K} \mathbb{E}\overline{Q}(w; \boldsymbol{x}_k^1, \boldsymbol{x}_k^2) \tag{14}$$

is a single-task problem over the common launch model w, and can be pursued via diffusion in a decentralized manner:

$$\begin{aligned}
\boldsymbol{\phi}_{k,i} &= \boldsymbol{w}_{k,i-1} - \mu \nabla \overline{Q}(\boldsymbol{w}_{k,i-1}; \boldsymbol{x}_{k,i}^1, \boldsymbol{x}_{k,i}^2) && (15) \\
&= \boldsymbol{w}_{k,i-1} - \mu \nabla Q(\boldsymbol{w}_{k,i-1} - \mu \nabla Q(\boldsymbol{w}_{k,i-1}; \boldsymbol{x}_{k,i}^1); \boldsymbol{x}_{k,i}^2) && (16) \\
\boldsymbol{w}_{k,i} &= \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \boldsymbol{\phi}_{\ell,i} && (17)
\end{aligned}$$

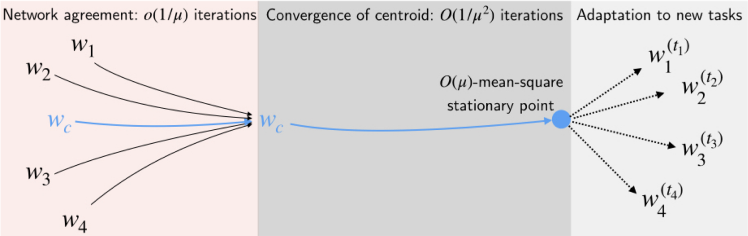
Diffusion-MAML: Performance

Convergence Guarantee [Kayaalp, Vlaski, and Sayed 2022]

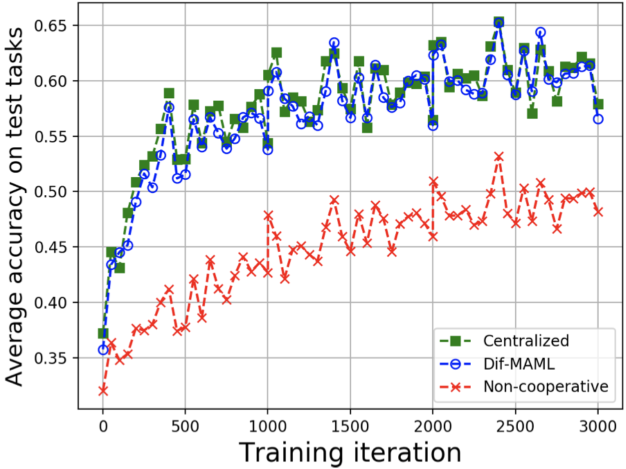
All agents agree on a common launch model in $o(1/\mu)$ iterations, and together find an approximately first-order stationary point of the aggregate objective:

$$\min_{w} \frac{1}{K} \sum_{k=1}^{K} \mathbb{E} Q(w - \mu \nabla Q(w; \boldsymbol{x}_k^1); \boldsymbol{x}_k^2) \tag{18}$$

in $O(1/\mu^2)$ iterations.

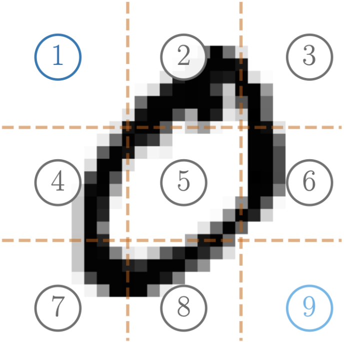


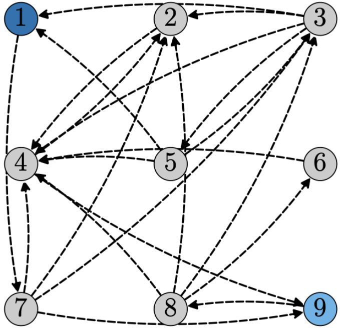
Diffusion-MAML: Few-Shot Image Recognition



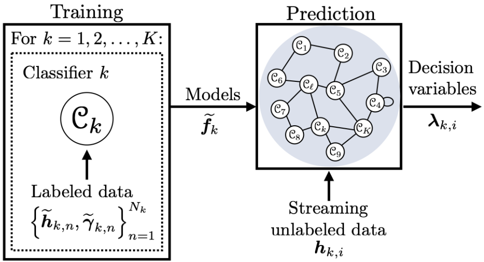
Example 4: Social Machine Learning

- So far we either explicitly induced a prior (multitask learning) or assumed an unknown relationship (model-agnostic meta-learning).
- What if there is no relationship between models, but rather their decisions?





Algorithm: Independent Training followed by Cooperative Inference



Social inference using consensus protocol on the decision [Bordignon et al, 2023]:

$$\boldsymbol{\delta}_{k,i} = \boldsymbol{\lambda}_{k,i-1} + \frac{\widehat{L}_k^{\widetilde{\boldsymbol{f}}_k}(\boldsymbol{h}_{k,i}|\gamma = +1)}{\widehat{L}_k^{\widetilde{\boldsymbol{f}}_k}(\boldsymbol{h}_{k,i}|\gamma = -1)}$$

$$\boldsymbol{\lambda}_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \boldsymbol{\delta}_{\ell,i}$$

Performance

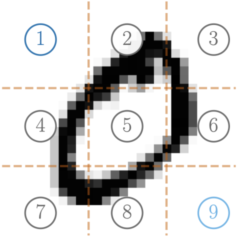
Consistency of Social Machine Learning [Bordignon et al, 2023]]

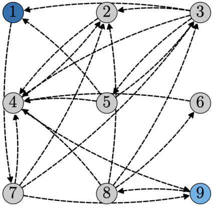
The social machine learning strategy enables consisten learning with probability:

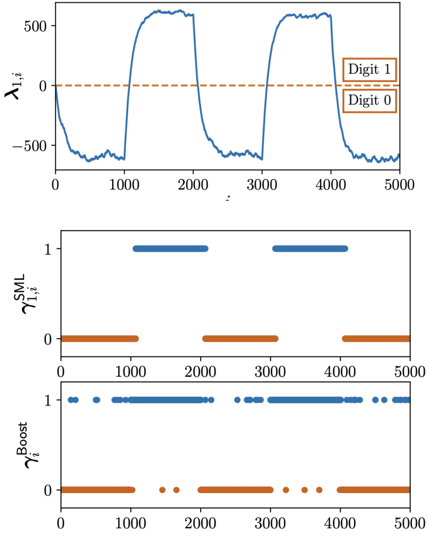
$$P_c \geq 1 - 2\exp\left\{-\frac{8N_{\max}}{\alpha^2\beta^2}(\epsilon-\rho)^2\right\} \tag{19}$$

where $\rho \geq \epsilon$, ϵ quantifies the complexity of the of the classification problem and ρ quantitfies the Rademacher complexity of the classifiers.

Application: Partial MNIST







Take-Aways

- Ordinary averaging for distributed learning yields improvement only in sufficiently homogeneous environments.
- **Multitask learning:** Induce priors through regularization and constraints.
- **Model-agnostic meta-learning:** Absence of task priors.
- **Social machine learning:** Fully heterogeneous classifiers observing a common state.

References

- R. Nassif, S. Vlaski, C. Richard and A. H. Sayed, "Learning Over Multitask Graphs—Part I: Stability Analysis," in IEEE Open Journal of Signal Processing, vol. 1, pp. 28-45, 2020.
- R. Nassif, S. Vlaski and A. H. Sayed, "Adaptation and Learning Over Networks Under Subspace Constraints—Part I: Stability Analysis," in IEEE Transactions on Signal Processing, vol. 68, pp. 1346-1360, 2020.
- M. Kayaalp, S. Vlaski and A. H. Sayed, "Dif-MAML: Decentralized Multi-Agent Meta-Learning," in IEEE Open Journal of Signal Processing, vol. 3, pp. 71-93, 2022.
- V. Bordignon, S. Vlaski, V. Matta and A. H. Sayed, "Learning from Heterogeneous Data Based on Social Interactions over Graphs," in IEEE Transactions on Information Theory, to appear.