Tomorrow's large dimensional AI: renewed intuitions and new mathematics?

Workshop MACS COMET-SCA on "Automatics and AI"

Romain COUILLET

CentraleSupélec, L2S, University of ParisSaclay, France
GSTATS IDEX DataScience Chair, GIPSA-lab, University Grenoble–Alpes, France.

June 2, 2021

2. Resurrecting Semi-Supervised Learning

Another, more striking, example: Semi-supervised Learning

Semi-supervised learning: a great idea that never worked!

Another, more striking, example: Semi-supervised Learning

Semi-supervised learning: a great idea that never worked!

- **Setting**: assume now
  - $x_1^{(a)}, \ldots, x_{n_{a,[l]}}^{(a)}$ already labelled (few),
  - $x_{n_{a,[l]}+1}^{(a)}, \ldots, x_{n_a}^{(a)}$ unlabelled (a lot).

Another, more striking, example: Semi-supervised Learning

Semi-supervised learning: a great idea that never worked!

- **Setting**: assume now
  - $x_1^{(a)}, \ldots, x_{n_{a,[l]}}^{(a)}$ already labelled (few),
  - $x_{n_{a,[l]}+1}^{(a)}, \ldots, x_{n_a}^{(a)}$ unlabelled (a lot).
- **Machine Learning original idea**: find "scores" F_{ia} for x_i to belong to class a

$$F = \operatorname{argmin}_{F \in \mathbb{R}^{n \times k}} \sum_{a=1}^{k} \sum_{i,j} K_{ij} \left(F_{ia} \qquad - F_{ja} \qquad \right)^2, \quad F_{ia}^{[l]} = \boldsymbol{\delta}_{\{x_i \in \mathcal{C}_a\}}.$$

Another, more striking, example: Semi-supervised Learning

Semi-supervised learning: a great idea that never worked!

- **Setting**: assume now
  - $x_1^{(a)}, \ldots, x_{n_{a,[l]}}^{(a)}$ already labelled (few),
  - $x_{n_{a,[l]}+1}^{(a)}, \ldots, x_{n_a}^{(a)}$ unlabelled (a lot).

- **Machine Learning original idea**: find "scores" F_{ia} for x_i to belong to class a

$$F = \operatorname{argmin}_{F \in \mathbb{R}^{n \times k}} \sum_{a=1}^{k} \sum_{i,j} K_{ij} \left(F_{ia} D_{ii}^{\alpha} - F_{ja} D_{jj}^{\alpha} \right)^2, \quad F_{ia}^{[l]} = \boldsymbol{\delta}_{\{x_i \in \mathcal{C}_a\}}.$$

Another, more striking, example: Semi-supervised Learning

Semi-supervised learning: a great idea that never worked!

- **Setting**: assume now
  - $x_1^{(a)}, \ldots, x_{n_{a,[l]}}^{(a)}$ already labelled (few),
  - $x_{n_{a,[l]}+1}^{(a)}, \ldots, x_{n_a}^{(a)}$ unlabelled (a lot).

- **Machine Learning original idea**: find "scores" F_{ia} for x_i to belong to class a

$$F = \operatorname{argmin}_{F \in \mathbb{R}^{n \times k}} \sum_{a=1}^{k} \sum_{i,j} K_{ij} \left(F_{ia} D_{ii}^{\alpha} - F_{ja} D_{jj}^{\alpha} \right)^2, \quad F_{ia}^{[l]} = \boldsymbol{\delta}_{\{x_i \in \mathcal{C}_a\}}.$$

- **Explicit solution**:

$$F^{[u]} = \left(I_{n_{[u]}} - D_{[u]}^{-1-\alpha} K_{[uu]} D^{\alpha}{}_{[u]} \right)^{-1} D_{[u]}^{-1-\alpha} K_{[ul]} D^{\alpha}{}_{[l]} F^{[l]}$$

where $D = \operatorname{diag}(K 1_n)$ (degree matrix) and $[ul]$, $[uu]$, …blocks of **l**abeled/**u**nlabeled data.

The finite-dimensional case: What we expect

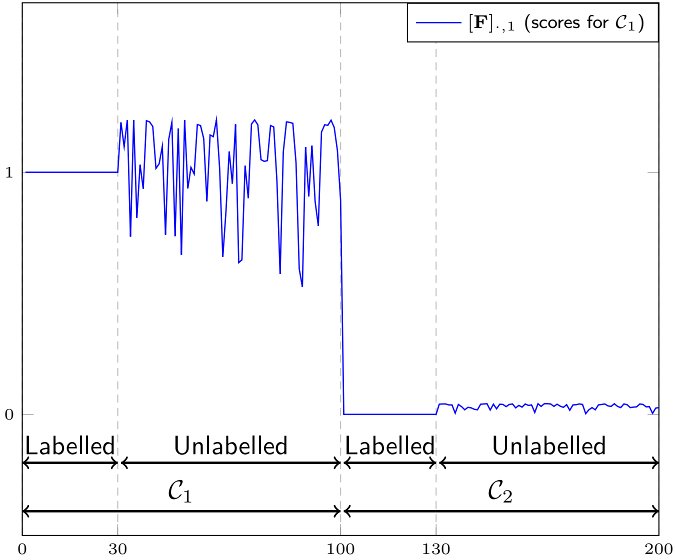


Figure: Outcome $\mathbf{F}$ of Laplacian algorithms ($\alpha = -1$) for $\mathcal{N}(\pm\mu, I_p)$ with $p = 1$.

The finite-dimensional case: What we expect

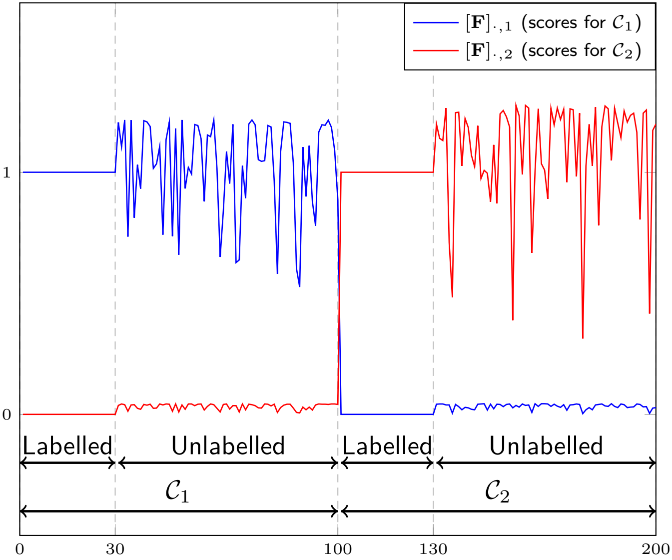


Figure: Outcome $\mathbf{F}$ of Laplacian algorithms ($\alpha = -1$) for $\mathcal{N}(\pm\mu, I_p)$ with $p = 1$.

The reality: What we see!

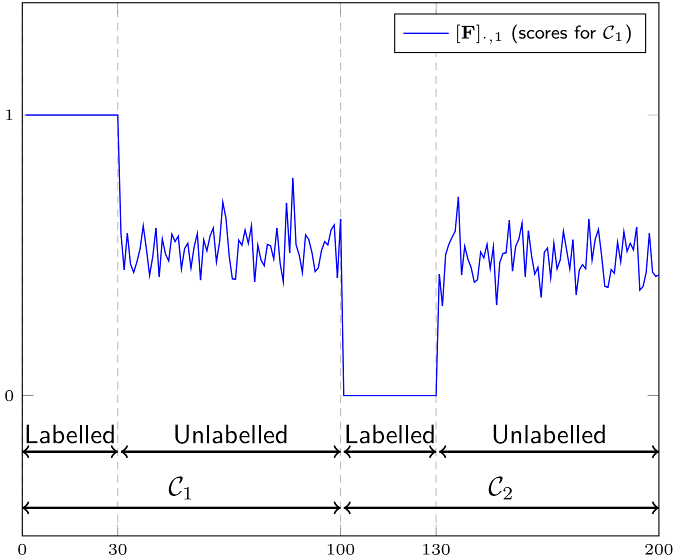


Figure: Outcome $\mathbf{F}$ of Laplacian algorithms ($\alpha = -1$) for $\mathcal{N}(\pm\mu, I_p)$ with $p = 80$.

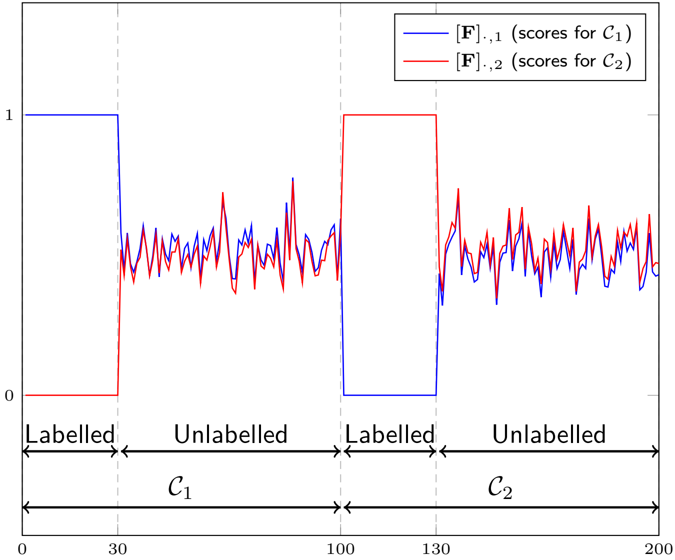
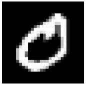
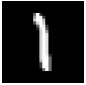


Figure: Outcome $\mathbf{F}$ of Laplacian algorithms ($\alpha = -1$) for $\mathcal{N}(\pm\mu, I_p)$ with $p = 80$.

The reality: What we see! (on MNIST)





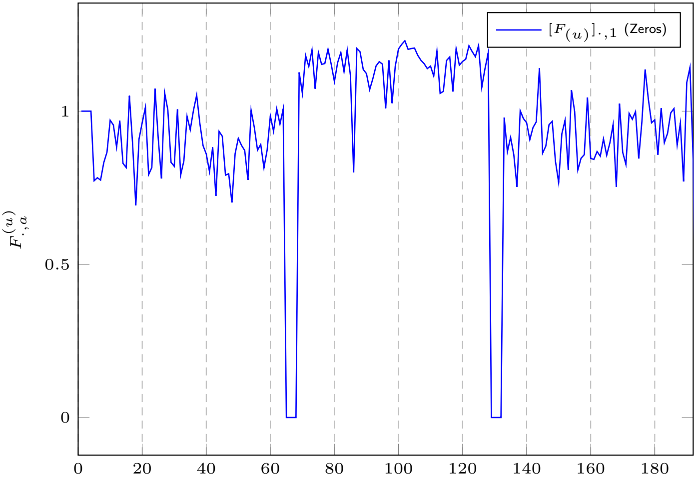


Figure: Vectors $[F^{(u)}]_{\cdot,a}$, $a = 1, 2, 3$, for 3-class MNIST data (zeros, ones, twos), $n = 192$, $p = 784$, $n_l/n = 1/16$, Gaussian kernel.

The reality: What we see! (on MNIST)

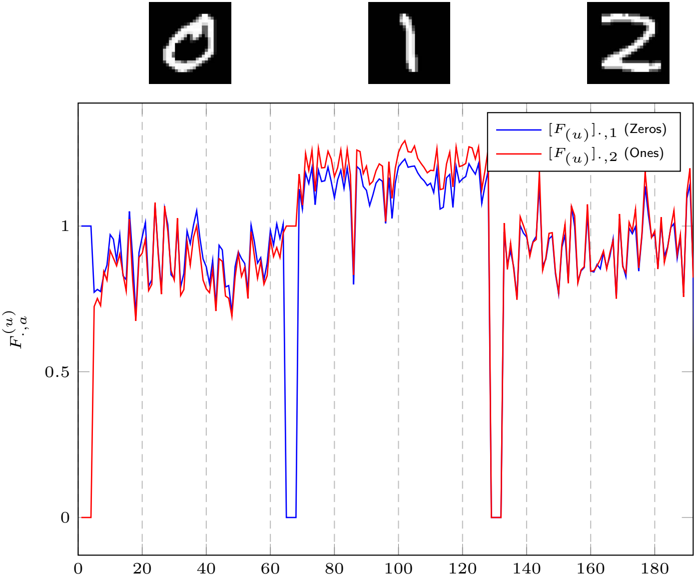


Figure: Vectors $[F^{(u)}]_{\cdot,a}$, $a = 1, 2, 3$, for 3-class MNIST data (zeros, ones, twos), $n = 192$, $p = 784$, $n_l/n = 1/16$, Gaussian kernel.

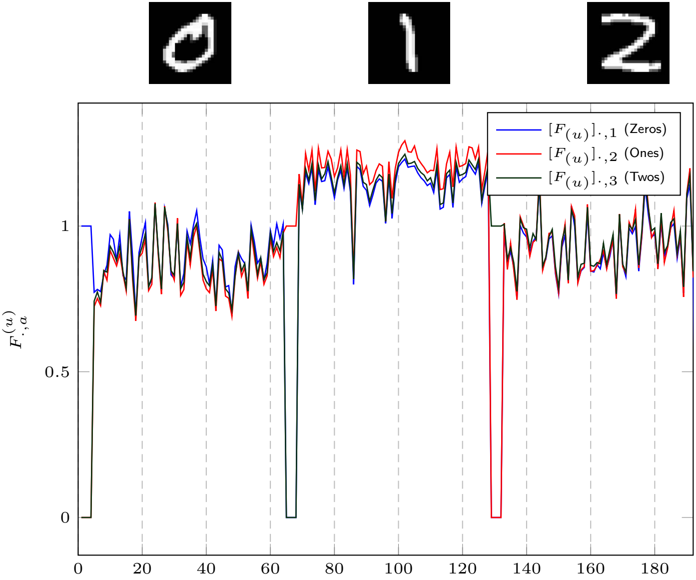


Figure: Vectors $[F^{(u)}]_{\cdot,a}$, $a = 1, 2, 3$, for 3-class MNIST data (zeros, ones, twos), $n = 192$, $p = 784$, $n_l/n = 1/16$, Gaussian kernel.

Consequences of the finite-dimensional "mismatch"

- A priori, **the algorithm does not work**

Exploiting RMT to resurrect SSL

Consequences of the finite-dimensional "mismatch"

- A priori, **the algorithm does not work**
- But, after some (not clearly motivated) normalization:

$$\alpha = -1, \quad F_{i\cdot} \leftarrow F_{i\cdot}/n_{[l],i}$$

  it works again...

Exploiting RMT to resurrect SSL

Consequences of the finite-dimensional "mismatch"

- A priori, **the algorithm does not work**
- But, after some (not clearly motivated) normalization:

$$\alpha = -1, \quad F_{i\cdot} \leftarrow F_{i\cdot}/n_{[l],i}$$

  it works again...

- **BUT** it does not use efficiently unlabelled data!

Consequences of the finite-dimensional "mismatch"

- A priori, **the algorithm does not work**
- But, after some (not clearly motivated) normalization:

$$\alpha = -1, \quad F_{i\cdot} \leftarrow F_{i\cdot}/n_{[l],i}$$

  it works again...

- **BUT** it does not use efficiently unlabelled data!

Chapelle, Schölkopf, Zien, "**Semi-Supervised Learning**", Chapter 4, 2009.

Our concern is this: it is frequently the case that we would be better off just discarding the unlabeled data and employing a supervised method, rather than taking a semi-supervised route. Thus ***we worry about the embarrassing situation where the addition of unlabeled data degrades the performance*** *of a classifier.*

Theorem (**[Mai,C'18]** Asymptotic Performance of SSL)

Letting $\alpha = -1$ and normalizing scores, for $x_i \in \mathcal{C}_b$ unlabelled, score vector $F_{i,\cdot} \in \mathbb{R}^k$ satisfies:

$$F_{i,\cdot} - G_b \to 0,\ G_b \sim \mathcal{N}(m_b, \Sigma_b)$$

with $m_b \in \mathbb{R}^k$, $\Sigma_b \in \mathbb{R}^{k\times k}$ function of

- $f(\tau), f'(\tau), f''(\tau)$, $\mu_1, \ldots, \mu_k$, $C_1, \ldots, C_k$
- *only n_l.*

Theorem ([Mai,C'18] Asymptotic Performance of SSL)

Letting $\alpha = -1$ and normalizing scores, for $x_i \in \mathcal{C}_b$ unlabelled, score vector $F_{i,\cdot} \in \mathbb{R}^k$ satisfies:

$$F_{i,\cdot} - G_b \to 0,\ G_b \sim \mathcal{N}(m_b, \Sigma_b)$$

with $m_b \in \mathbb{R}^k$, $\Sigma_b \in \mathbb{R}^{k \times k}$ function of

- $f(\tau), f'(\tau), f''(\tau)$, $\mu_1, \ldots, \mu_k$, $C_1, \ldots, C_k$
- *only n_l.*

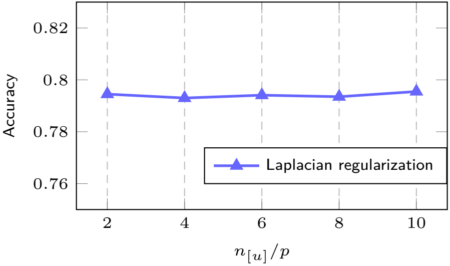


Figure: Accuracy as a function of $n_{[u]}/p$ with $n_{[l]}/p = 2$, $c_1 = c_2$, $p = 100$, $-\boldsymbol{\mu}_1 = \boldsymbol{\mu}_2 = [1; \mathbf{0}_{p-1}]$, $\{\mathbf{C}\}_{i,j} = .1^{|i-j|}$. Graph constructed with $K_{ij} = e^{-\|x_i - x_j\|^2/p}$.

Theorem ([Mai,C'18] Asymptotic Performance of SSL)

Letting $\alpha = -1$ and normalizing scores, for $x_i \in \mathcal{C}_b$ unlabelled, score vector $F_{i,\cdot} \in \mathbb{R}^k$ satisfies:

$$F_{i,\cdot} - G_b \to 0, \; G_b \sim \mathcal{N}(m_b, \Sigma_b)$$

with $m_b \in \mathbb{R}^k$, $\Sigma_b \in \mathbb{R}^{k\times k}$ function of

- $f(\tau), f'(\tau), f''(\tau)$, $\mu_1, \ldots, \mu_k$, $C_1, \ldots, C_k$
- *only n_l.*

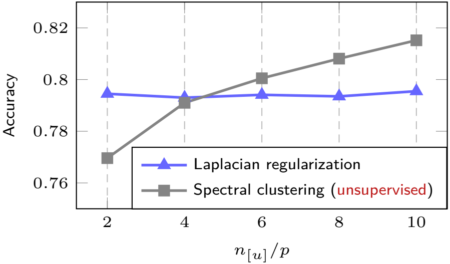


Figure: Accuracy as a function of $n_{[u]}/p$ with $n_{[l]}/p = 2$, $c_1 = c_2$, $p = 100$, $-\boldsymbol{\mu}_1 = \boldsymbol{\mu}_2 = [1; \mathbf{0}_{p-1}]$, $\{\mathbf{C}\}_{i,j} = .1^{|i-j|}$. Graph constructed with $K_{ij} = e^{-\|x_i - x_j\|^2/p}$.

Improved SSL

Solution: From RMT calculus (but not from ML intuition!), solution is to replace K by

$$\tilde{K} \equiv PKP, \quad P = I_n - \frac{1}{n} 1_n 1_n^{\mathsf{T}}.$$

Solution: From RMT calculus (but not from ML intuition!), solution is to replace K by

$$\tilde{K} \equiv PKP, \quad P = I_n - \frac{1}{n} 1_n 1_n^\mathsf{T}.$$

Theorem ([Mai,C'19] Asymptotic Performance of **Improved** SSL)

For $x_i \in \mathcal{C}_b$ unlabelled, score vector $\tilde{F}_{i,\cdot} \in \mathbb{R}^k$ satisfies:

$$\tilde{F}_{i,\cdot} - \tilde{G}_b \to 0, \ \tilde{G}_b \sim \mathcal{N}(\tilde{m}_b, \tilde{\Sigma}_b)$$

with $\tilde{m}_b \in \mathbb{R}^k$, $\tilde{\Sigma}_b \in \mathbb{R}^{k \times k}$ function of

- $f(\tau), f'(\tau), f''(\tau)$, $\mu_1, \ldots, \mu_k$, $C_1, \ldots, C_k$
- n_l **and** n_u.

Solution: From RMT calculus (but not from ML intuition!), solution is to replace K by

$$\tilde{K} \equiv PKP, \quad P = I_n - \frac{1}{n} 1_n 1_n^{\mathsf{T}}.$$

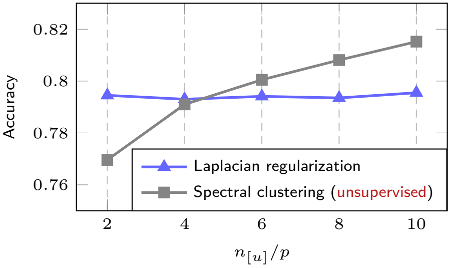
Theorem ([Mai,C'19] Asymptotic Performance of **Improved** SSL)

For $x_i \in \mathcal{C}_b$ unlabelled, score vector $\tilde{F}_{i,\cdot} \in \mathbb{R}^k$ satisfies:

$$\tilde{F}_{i,\cdot} - \tilde{G}_b \to 0, \ \tilde{G}_b \sim \mathcal{N}(\tilde{m}_b, \tilde{\Sigma}_b)$$

with $\tilde{m}_b \in \mathbb{R}^k$, $\tilde{\Sigma}_b \in \mathbb{R}^{k \times k}$ function of

- $f(\tau), f'(\tau), f''(\tau), \mu_1, \ldots, \mu_k, C_1, \ldots, C_k$
- n_l **and** n_u.



Solution: From RMT calculus (but not from ML intuition!), solution is to replace K by

$$\tilde{K} \equiv PKP, \quad P = I_n - \frac{1}{n} 1_n 1_n^\mathsf{T}.$$

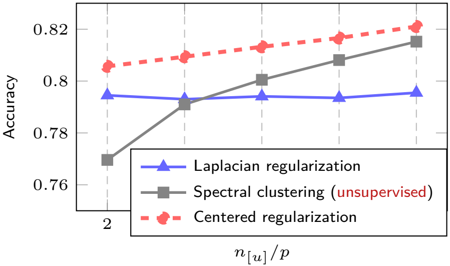
Theorem ([Mai,C'19] Asymptotic Performance of **Improved** SSL)

For $x_i \in \mathcal{C}_b$ unlabelled, score vector $\tilde{F}_{i,\cdot} \in \mathbb{R}^k$ satisfies:

$$\tilde{F}_{i,\cdot} - \tilde{G}_b \to 0, \; \tilde{G}_b \sim \mathcal{N}(\tilde{m}_b, \tilde{\Sigma}_b)$$

with $\tilde{m}_b \in \mathbb{R}^k$, $\tilde{\Sigma}_b \in \mathbb{R}^{k \times k}$ function of

- $f(\tau), f'(\tau), f''(\tau)$, $\mu_1, \ldots, \mu_k$, $C_1, \ldots, C_k$
- n_l **and** n_u.



What about real data?

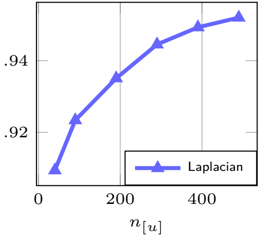


Figure: **Top**: distribution of normalized pairwise distances for noisy MNIST data (8,9). **Bottom**: average accuracy as a function of $n_{[u]}$ with $n_{[l]} = 10$, computed over 1000 random realizations.

What about real data?

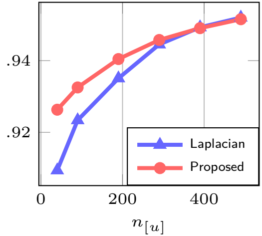


Figure: **Top**: distribution of normalized pairwise distances for noisy MNIST data (8,9). **Bottom**: average accuracy as a function of $n_{[u]}$ with $n_{[l]} = 10$, computed over 1000 random realizations.

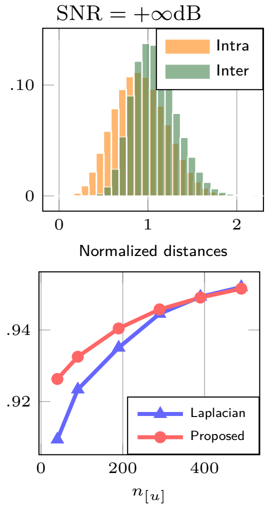


Figure: **Top**: distribution of normalized pairwise distances for noisy MNIST data (8,9). **Bottom**: average accuracy as a function of $n_{[u]}$ with $n_{[l]} = 10$, computed over 1000 random realizations.

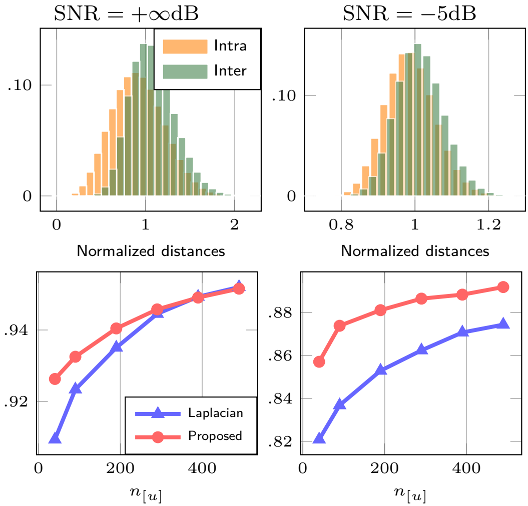


Figure: **Top**: distribution of normalized pairwise distances for noisy MNIST data (8,9). **Bottom**: average accuracy as a function of $n_{[u]}$ with $n_{[l]} = 10$, computed over 1000 random realizations.

What about real data?

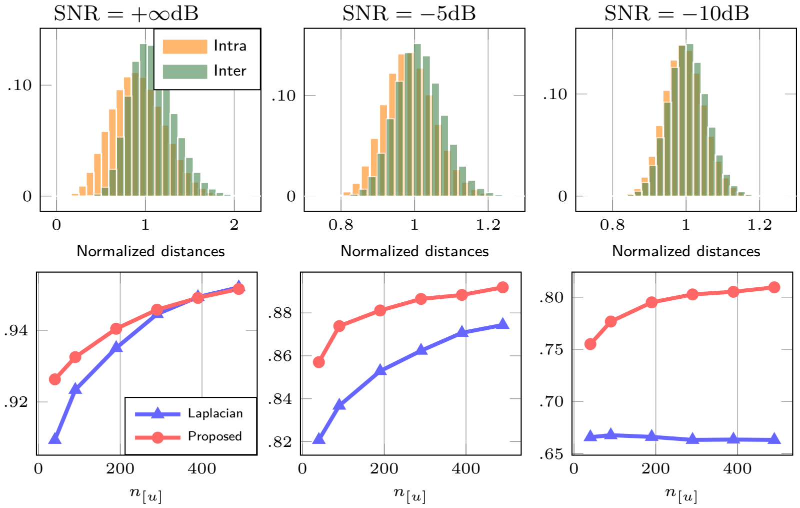
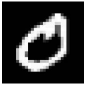
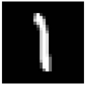


Figure: **Top**: distribution of normalized pairwise distances for noisy MNIST data (8,9). **Bottom**: average accuracy as a function of $n_{[u]}$ with $n_{[l]} = 10$, computed over 1000 random realizations.

Experimental evidence: MNIST

Digits	(0,8)	(2,7)	(6,9)
$n_u = 100$			
Centered kernel **(RMT)**	**89.5±3.6**	**89.5±3.4**	**85.3±5.9**
Iterated centered kernel **(RMT)**	**89.5±3.6**	**89.5±3.4**	**85.3±5.9**
Laplacian	75.5±5.6	74.2±5.8	70.0±5.5
Iterated Laplacian	87.2±4.7	86.0±5.2	81.4±6.8
Manifold	88.0±4.7	88.4±3.9	82.8±6.5
$n_u = 1000$			
Centered kernel **(RMT)**	92.2±0.9	92.5±0.8	92.6±1.6
Iterated centered kernel **(RMT)**	**92.3±0.9**	**92.5± 0.8**	**92.9±1.4**
Laplacian	65.6±4.1	74.4±4.0	69.5±3.7
Iterated Laplacian	**92.2±0.9**	92.4±0.9	92.0±1.6
Manifold	91.1±1.7	91.4±1.9	91.4±2.0

Table: Comparison of classification accuracy (%) on MNIST datasets with $n_l = 10$. Computed over 1000 random iterations for $n_u = 100$ and 100 for $n_u = 1000$.

Experimental evidence: Traffic signs (HOG features)

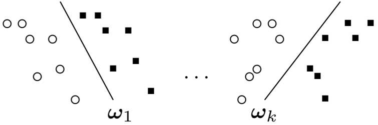
Class ID	(2,7)	(9,10)	(11,18)
$n_u = 100$			
Centered kernel **(RMT)**	79.0±10.4	77.5±9.2	78.5±7.1
Iterated centered kernel **(RMT)**	**85.3±5.9**	**89.2±5.6**	**90.1±6.7**
Laplacian	73.8±9.8	77.3±9.5	78.6±7.2
Iterated Laplacian	83.7±7.2	88.0±6.8	87.1±8.8
Manifold	77.6±8.9	81.4±10.4	82.3±10.8
$n_u = 1000$			
Centered kernel **(RMT)**	83.6±2.4	84.6±2.4	88.7±9.4
Iterated centered kernel **(RMT)**	**84.8±3.8**	**88.0±5.5**	**96.4±3.0**
Laplacian	72.7±4.2	88.9±5.7	95.8±3.2
Iterated Laplacian	83.0±5.5	88.2±6.0	92.7±6.1
Manifold	77.7±5.8	85.0±9.0	90.6±8.1

Table: Comparison of classification accuracy (%) on German Traffic Sign datasets with $n_l = 10$. Computed over 1000 random iterations for $n_u = 100$ and 100 for $n_u = 1000$.

3. Making complex ML frameworks simple: Multitask Learning

➸ **Problem:** k classification tasks with data $X = [X_1, \ldots, X_k]$, $X_i = [X_i^{(1)}, X_i^{(2)}] \in \mathbb{R}^{p \times n_i}$ and labels $y_i \in \mathbb{R}^{n_i}$ for each task i.

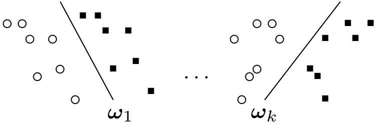
➳ **Problem:** k classification tasks with data $X = [X_1, \ldots, X_k]$, $X_i = [X_i^{(1)}, X_i^{(2)}] \in \mathbb{R}^{p \times n_i}$ and labels $y_i \in \mathbb{R}^{n_i}$ for each task i.



$$\min_{(\omega_i, b_i)} \frac{1}{2}\|\omega_i\|^2 + \frac{\gamma_i}{2}\|\xi_i\|^2$$

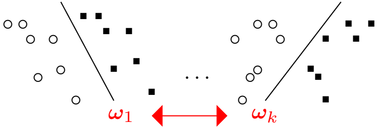
$$\xi_i = y_i - (X_i^\mathsf{T}\omega_i + b_i 1_{n_i}), \quad \forall i \in \{1, \ldots, k\}.$$

➳ **Problem:** k classification tasks with data $X = [X_1, \ldots, X_k]$, $X_i = [X_i^{(1)}, X_i^{(2)}] \in \mathbb{R}^{p \times n_i}$ and labels $y_i \in \mathbb{R}^{n_i}$ for each task i.



$$\min_{(\boldsymbol{\omega}_i, b_i)} \frac{1}{2}\|\boldsymbol{\omega}_i\|^2 + \frac{\gamma_i}{2}\|\xi_i\|^2$$

$$\xi_i = y_i - (X_i^\mathsf{T}\boldsymbol{\omega}_i + b_i 1_{n_i}), \quad \forall i \in \{1, \ldots, k\}.$$



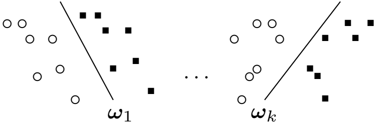
Relatedness Assumption: $\boldsymbol{\omega}_i = \boldsymbol{\omega}_0 + \mathbf{v}_i$

$$\min_{(\boldsymbol{\omega}_0, \mathbf{v}_i, b_i)} \frac{1}{2\lambda}\|\boldsymbol{\omega}_0\|^2 + \frac{1}{2}\sum_{i=1}^{k} \frac{\|\mathbf{v}_i\|^2}{\gamma_i} + \frac{1}{2}\sum_{i=1}^{k}\|\xi_i\|^2$$

$$\xi_i = y_i - (X_i^\mathsf{T}\boldsymbol{\omega}_i + b_i 1_{n_i}), \quad \forall i \in \{1, \ldots, k\}.$$

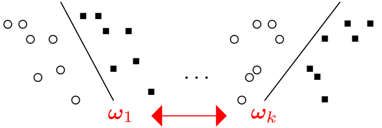
Multitask Learning Analysis and Improvement

➼ **Problem:** k classification tasks with data $X = [X_1, \ldots, X_k]$, $X_i = [X_i^{(1)}, X_i^{(2)}] \in \mathbb{R}^{p \times n_i}$ and labels $y_i \in \mathbb{R}^{n_i}$ for each task i.



$$\min_{(\omega_i, b_i)} \frac{1}{2}\|\omega_i\|^2 + \frac{\gamma_i}{2}\|\xi_i\|^2$$

$$\xi_i = y_i - (X_i^\mathsf{T}\omega_i + b_i 1_{n_i}), \quad \forall i \in \{1, \ldots, k\}.$$



Relatedness Assumption: $\omega_i = \omega_0 + \mathbf{v}_i$

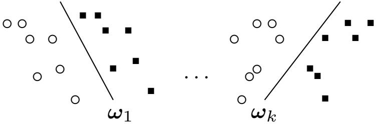
$$\min_{(\omega_0, \mathbf{v}_i, b_i)} \frac{1}{2\lambda}\|\omega_0\|^2 + \frac{1}{2}\sum_{i=1}^{k} \frac{\|\mathbf{v}_i\|^2}{\gamma_i} + \frac{1}{2}\sum_{i=1}^{k}\|\xi_i\|^2$$

$$\xi_i = y_i - (X_i^\mathsf{T}\omega_i + b_i 1_{n_i}), \quad \forall i \in \{1, \ldots, k\}.$$

➼ **Classification score**: for test data $\mathbf{x}$ for Task i: $g_i(\mathbf{x}) = \mathbf{x}^\mathsf{T}\omega_i + b_i$

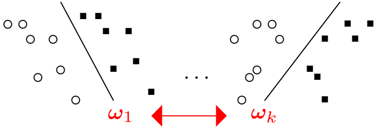
Multitask Learning Analysis and Improvement

➻ **Problem:** k classification tasks with data $X = [X_1, \ldots, X_k]$, $X_i = [X_i^{(1)}, X_i^{(2)}] \in \mathbb{R}^{p \times n_i}$ and labels $y_i \in \mathbb{R}^{n_i}$ for each task i.



$$\min_{(\omega_i, b_i)} \frac{1}{2}\|\omega_i\|^2 + \frac{\gamma_i}{2}\|\xi_i\|^2$$

$$\xi_i = y_i - (X_i^{\mathsf{T}}\omega_i + b_i 1_{n_i}), \quad \forall i \in \{1, \ldots, k\}.$$



Relatedness Assumption: $\omega_i = \omega_0 + \mathbf{v}_i$

$$\min_{(\omega_0, \mathbf{v}_i, b_i)} \frac{1}{2\lambda}\|\omega_0\|^2 + \frac{1}{2}\sum_{i=1}^{k}\frac{\|\mathbf{v}_i\|^2}{\gamma_i} + \frac{1}{2}\sum_{i=1}^{k}\|\xi_i\|^2$$

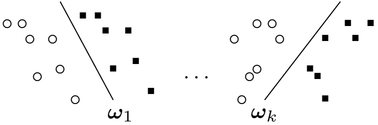
$$\xi_i = y_i - (X_i^{\mathsf{T}}\omega_i + b_i 1_{n_i}), \quad \forall i \in \{1, \ldots, k\}.$$

➻ **Classification score**: for test data $\mathbf{x}$ for Task i: $g_i(\mathbf{x}) = \mathbf{x}^{\mathsf{T}}\omega_i + b_i$

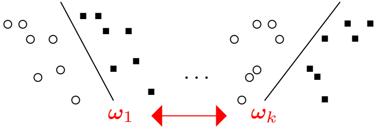
➻ **Key elements:**

- labels $[y_i]_\ell \in \{\pm 1\}$ (binary) or $[y_i]_\ell = (0, \cdots, 0, 1, 0, \cdots, 0)^{\mathsf{T}}$ (multiclass)

- **Problem:** k classification tasks with data $X = [X_1, \ldots, X_k]$, $X_i = [X_i^{(1)}, X_i^{(2)}] \in \mathbb{R}^{p \times n_i}$ and labels $y_i \in \mathbb{R}^{n_i}$ for each task i.



$$\min_{(\boldsymbol{\omega}_i, b_i)} \frac{1}{2}\|\boldsymbol{\omega}_i\|^2 + \frac{\gamma_i}{2}\|\xi_i\|^2$$
$$\xi_i = y_i - (X_i^\mathsf{T}\boldsymbol{\omega}_i + b_i 1_{n_i}), \quad \forall i \in \{1, \ldots, k\}.$$



Relatedness Assumption: $\boldsymbol{\omega}_i = \boldsymbol{\omega}_0 + \mathbf{v}_i$

$$\min_{(\boldsymbol{\omega}_0, \mathbf{v}_i, b_i)} \frac{1}{2\lambda}\|\boldsymbol{\omega}_0\|^2 + \frac{1}{2}\sum_{i=1}^{k}\frac{\|\mathbf{v}_i\|^2}{\gamma_i} + \frac{1}{2}\sum_{i=1}^{k}\|\xi_i\|^2$$
$$\xi_i = y_i - (X_i^\mathsf{T}\boldsymbol{\omega}_i + b_i 1_{n_i}), \quad \forall i \in \{1, \ldots, k\}.$$

- **Classification score**: for test data $\mathbf{x}$ for Task i: $g_i(\mathbf{x}) = \mathbf{x}^\mathsf{T}\boldsymbol{\omega}_i + b_i$
- **Key elements:**
  - labels $[y_i]_\ell \in \{\pm 1\}$ (binary) or $[y_i]_\ell = (0, \cdots, 0, 1, 0, \cdots, 0)^\mathsf{T}$ (multiclass)
  - decision score $g_i(\mathbf{x}) \gtrless 0$.

Theorem (Asymptotics of $g_i(\mathbf{x})$ **[Tiomoko,Tiomoko,Couillet'21]**)

For $\mathbf{x} \sim \mathcal{N}(\mu_{ij}, I_p)$ *(Task* i*, Class* j*), and* $[y_{(i,j)}]_\ell = \tilde{y}_{ij} \in \mathbb{R}$ *constant in each class,*

$$g_i(\mathbf{x}) - G_{ij} \to 0, \quad G_{ij} \sim \mathcal{N}(m_{ij}, {\sigma_i}^2)$$

Theorem (Asymptotics of $g_i(\mathbf{x})$ [Tiomoko,Tiomoko,Couillet'21])

For $\mathbf{x} \sim \mathcal{N}(\mu_{ij}, I_p)$ (Task i, Class j), and $[y_{(i,j)}]_\ell = \tilde{y}_{ij} \in \mathbb{R}$ constant in each class,

$$g_i(\mathbf{x}) - G_{ij} \to 0, \quad G_{ij} \sim \mathcal{N}(m_{ij}, {\sigma_i}^2)$$

where, for $m = [m_{11}, \ldots, m_{k2}]^\mathsf{T}$,

$$m = \tilde{y} - \mathcal{D}_{\bar{\Delta}}^{-\frac{1}{2}} \Gamma \mathcal{D}_{\bar{\Delta}}^{\frac{1}{2}} \mathring{\tilde{y}}$$

$$\Gamma = \left(I_{2k} + \left(\mathcal{A} \otimes \mathbf{1}_2 \mathbf{1}_2^\mathsf{T}\right) \odot \mathcal{M}\right)^{-1}$$

$$\mathcal{A} = \left(I_k + \mathcal{D}_{\tilde{\Delta}}^{-\frac{1}{2}} \left(\mathcal{D}_\gamma + \lambda \mathbf{1}_k \mathbf{1}_k^\mathsf{T}\right)^{-1} \mathcal{D}_{\tilde{\Delta}}^{-\frac{1}{2}}\right)^{-1}.$$

Theorem (Asymptotics of $g_i(\mathbf{x})$ [Tiomoko,Tiomoko,Couillet'21])

For $\mathbf{x} \sim \mathcal{N}(\mu_{ij}, I_p)$ (Task i, Class j), and $[y_{(i,j)}]_\ell = \tilde{y}_{ij} \in \mathbb{R}$ constant in each class,

$$g_i(\mathbf{x}) - G_{ij} \to 0, \quad G_{ij} \sim \mathcal{N}(m_{ij}, {\sigma_i}^2)$$

where, for $m = [m_{11}, \ldots, m_{k2}]^\mathsf{T}$,

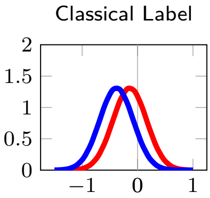
$$\begin{aligned} m &= \tilde{y} - \mathcal{D}_{\tilde{\Delta}}^{-\frac{1}{2}} \Gamma \mathcal{D}_{\tilde{\Delta}}^{\frac{1}{2}} \mathring{\tilde{y}} \\ \Gamma &= \left(I_{2k} + \left(\mathcal{A} \otimes \mathbf{1}_2 \mathbf{1}_2^\mathsf{T} \right) \odot \mathcal{M} \right)^{-1} \\ \mathcal{A} &= \left(I_k + \mathcal{D}_{\tilde{\Delta}}^{-\frac{1}{2}} \left(\mathcal{D}_\gamma + \lambda \mathbf{1}_k \mathbf{1}_k^\mathsf{T} \right)^{-1} \mathcal{D}_{\tilde{\Delta}}^{-\frac{1}{2}} \right)^{-1}. \end{aligned}$$

Theorem (Optimal (small dimensional) $\tilde{y} \in \mathbb{R}^{2k}$)

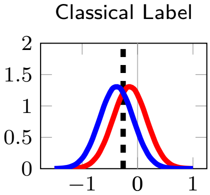
Optimal $\tilde{y} = \tilde{y}^{\star(i)}$ for Task i,

$$\tilde{y}^{\star(i)} = \operatorname{argmax}_{\tilde{y} \in \mathbb{R}^{2k}} \frac{(m_{i1} - m_{i2})^2}{\sigma_i^2} = \mathcal{D}_{\tilde{\Delta}}^{-\frac{1}{2}} \Gamma^{-1} \mathcal{H}[(\mathcal{A} \otimes \mathbf{1}_2 \mathbf{1}_2^\mathsf{T}) \odot \mathcal{M}] \mathcal{D}_{\tilde{\Delta}}^{-\frac{1}{2}} (e_{i1}^{[2k]} - e_{i2}^{[2k]})$$

with $e_{ij}^{[2k]} = (0, \ldots, 0, 1, 0, \ldots, 0)^\mathsf{T}$ with 1 in position (i,j).

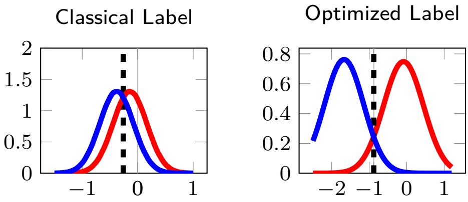


- Theoretical prediction
  (asymptotic stats of $g(\mathbf{x})$)

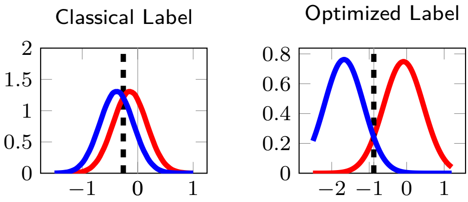


- Theoretical prediction
  (asymptotic stats of $g(\mathbf{x})$)
- Center the decision threshold!

Multi Task Learning Analysis and Improvement



- Theoretical prediction (asymptotic stats of $g(\mathbf{x})$)
- Center the decision threshold!
- **Label optimization** (kills negative transfer!)



- Theoretical prediction (asymptotic stats of $g(\mathbf{x})$)
- Center the decision threshold!
- **Label optimization** (kills negative transfer!)

Table: Classification accuracy over Office+Caltech256 database. c(Caltech), w(Webcam), a(Amazon), d(dslr) using VGG features.

S/T	c → w	w → c	c → a	a → c	w → a	a → d	d → a	w → d	c → d	d → c
LSSVM	90.70	**89.90**	92.90	90.00	93.80	78.70	93.50	95.00	85.00	**90.20**
MMDT	90.73	87.05	90.83	84.40	94.17	86.25	94.58	97.50	86.25	87.23
ILS	77.29	73.55	86.85	76.22	86.22	71.34	74.53	82.80	68.15	63.49
CDLS	*96.70*	*88.30*	*93.54*	*88.30*	**93.54**	*92.50*	*93.54*	93.75	*93.75*	*88.30*
RMT	**98.00**	**89.90**	**94.40**	**90.60**	**94.40**	**93.80**	**94.20**	**100**	**92.50**	89.90

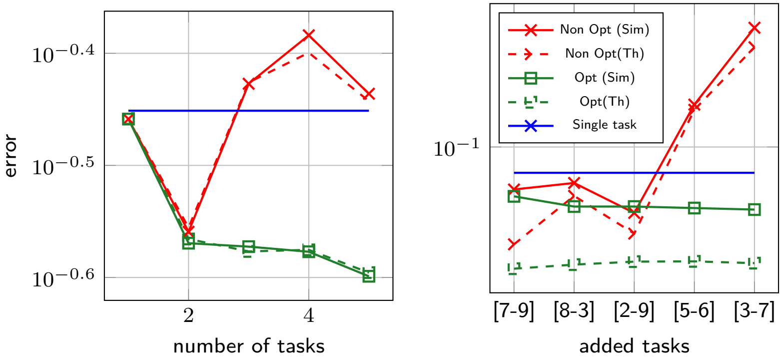
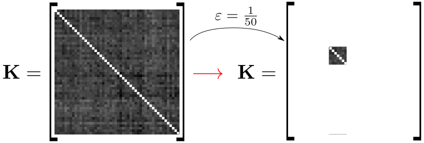


Figure: Classification accuracy for increasing number of tasks. **(Left)** Synthetic data with random correlation; **(Right)** MNIST (HOG features): $(1,4)$ as target, added task in x-axis; in both settings, $\gamma = \mathbf{1}_k$, $\lambda = 10$. ***Optimized scheme avoids negative transfer.***

4. Towards cheap “environment-friendly” learning

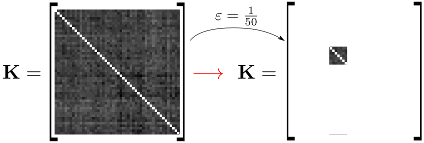
- **Computation cost reduction**: $(p, n \gg 1)$

$\rightarrow$ ε-subsampling $K \in \mathbb{R}^{n\varepsilon \times n\varepsilon}$

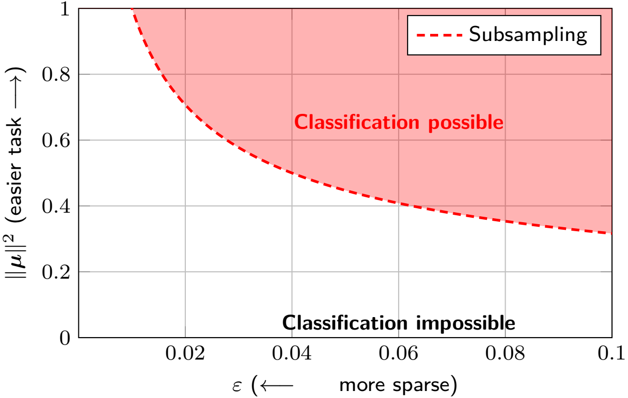


- **Computation cost reduction**: $(p, n \gg 1)$

  $\rightarrow$ ε-subsampling $K \in \mathbb{R}^{n\varepsilon \times n\varepsilon}$



- **Phase transition of spectral clustering**: $(x_i \sim \mathcal{N}(\pm\mu, I_p),\ n/p = 100)$,

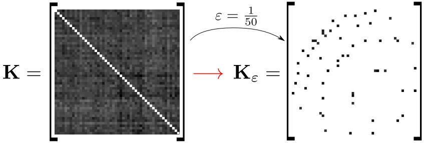


Towards efficient and cheap learning

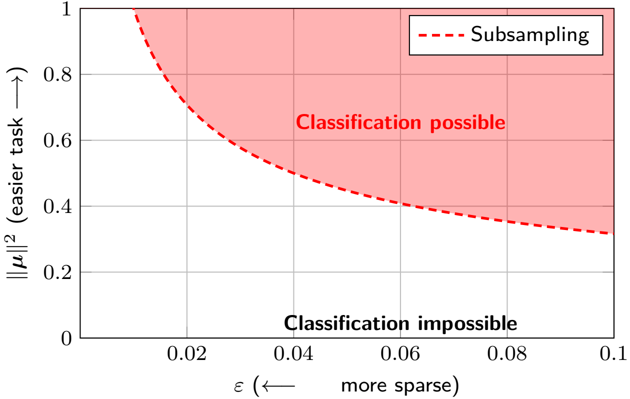
- **Computation cost reduction**: $(p, n \gg 1)$

  $\rightarrow$ ε-subsampling $K \in \mathbb{R}^{n\varepsilon \times n\varepsilon}$

  $\rightarrow$ $K_\varepsilon \equiv K \odot B$ with $B_{ij} \sim \mathrm{Bern}(\varepsilon)$ i.i.d.



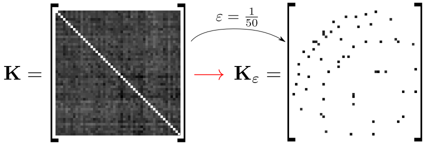
- **Phase transition of spectral clustering**: $(x_i \sim \mathcal{N}(\pm\mu, I_p),\ n/p = 100)$,



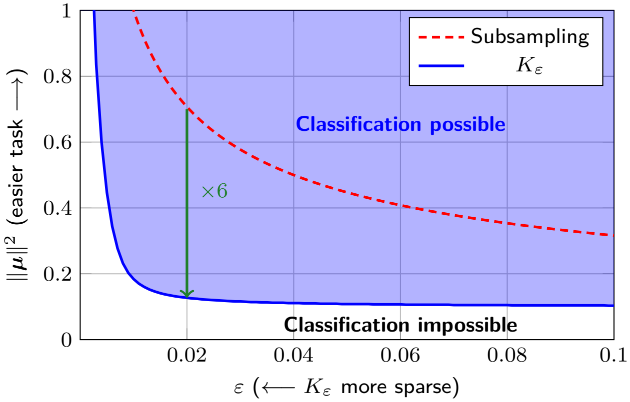
- **Computation cost reduction**: $(p, n \gg 1)$

  $\to$ ε-subsampling $K \in \mathbb{R}^{n\varepsilon \times n\varepsilon}$

  $\to$ $K_\varepsilon \equiv K \odot B$ with $B_{ij} \sim \mathrm{Bern}(\varepsilon)$ i.i.d.



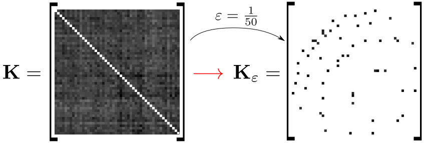
- **Phase transition of spectral clustering**: $(x_i \sim \mathcal{N}(\pm\mu, I_p),\ n/p = 100)$,



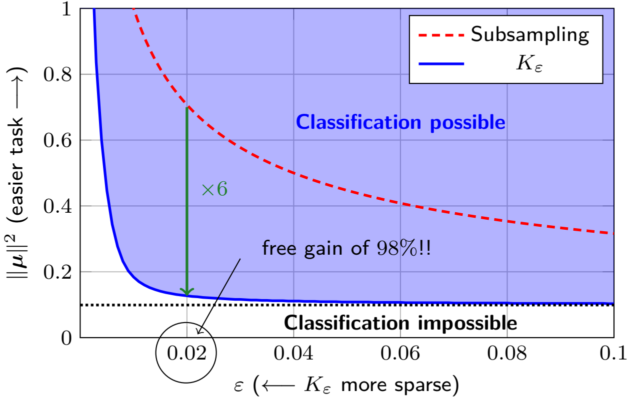
- **Computation cost reduction**: $(p, n \gg 1)$

$\rightarrow$ ε-subsampling $K \in \mathbb{R}^{n\varepsilon \times n\varepsilon}$

$\rightarrow$ $K_\varepsilon \equiv K \odot B$ with $B_{ij} \sim \text{Bern}(\varepsilon)$ i.i.d.



- **Phase transition of spectral clustering**: $(x_i \sim \mathcal{N}(\pm\mu, I_p),\ n/p = 100)$,



- **Going further: double-puncturing:**

$$K_{\varepsilon_S,\varepsilon_B} = \left\{ \frac{1}{p}(X \odot S)^\mathsf{T}(X \odot S) \right\} \odot B$$

- **Going further: double-puncturing:**

$$K_{\varepsilon_S,\varepsilon_B} = \left\{ \frac{1}{p}(X \odot S)^\mathsf{T}(X \odot S) \right\} \odot B$$

- **Spectrum of $K_{\varepsilon_S,\varepsilon_B}$:** mixture of semi-circle (pushed by B!) and MP-law ($X^\mathsf{T}X$)

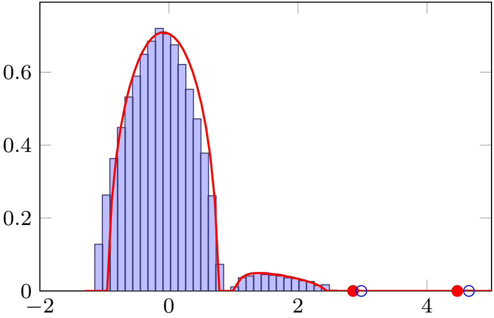


Figure: Two "humps" reminding the semi-circular and Marčenko-Pastur laws.

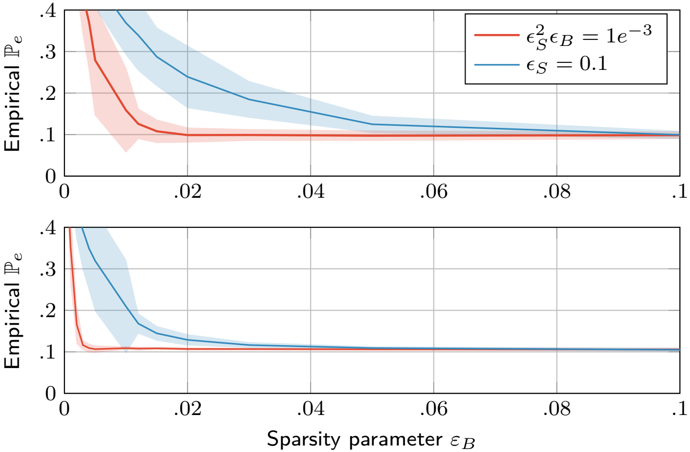


Figure: Empirical classification errors for 2-class (balanced) MNIST-fashion images (`trouser` vs `pullover`), with $n = 512$ (**top**) and $n = 2048$ (**bottom**). ***Note the "plateaus" predicted by theory!***.

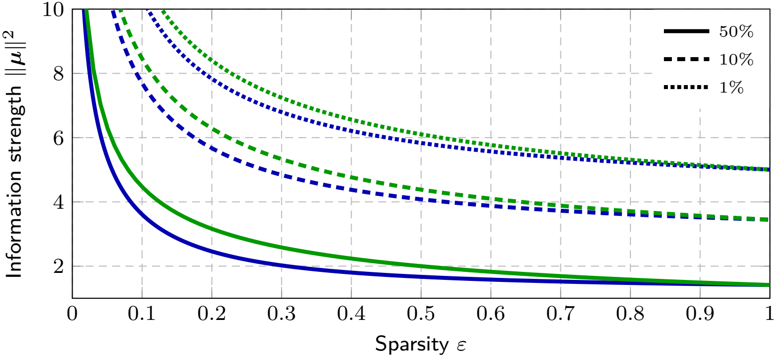
➽ **Smart sparsifying**: threshold (binary) kernels

$$\mathbf{K}_\varepsilon = \{f\left(\mathbf{K}_{ij}\right)\}_{i,j=1}^n \text{, with } f(t) = \begin{cases} t1_{|t|>s}, & \text{thresholding} \\ \operatorname{sgn}(t)1_{|t|>s}, & \text{thresholding \& binarization.} \end{cases}$$

- **Smart sparsifying**: threshold (binary) kernels

$$\mathbf{K}_\varepsilon = \{f(\mathbf{K}_{ij})\}_{i,j=1}^n, \text{ with } f(t) = \begin{cases} t1_{|t|>s}, & \text{thresholding} \\ \mathrm{sgn}(t)1_{|t|>s}, & \text{thresholding \& binarization.} \end{cases}$$

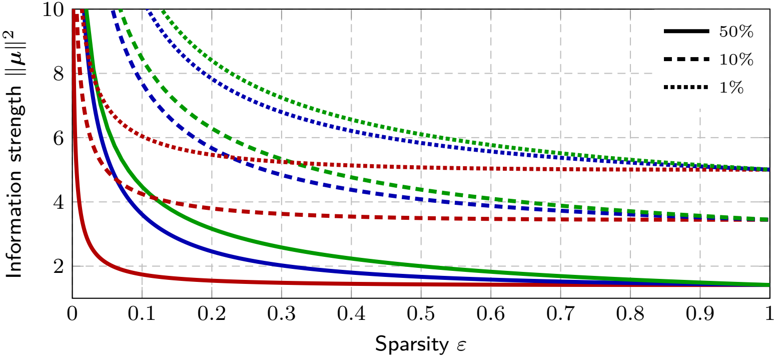
- **"Equi-performance" level comparison** (50% =‘phase transition’): subsampling (green), uniform random sparsity (blue) vs. thresholding (red), here for $n/p = 2$.



➽ **Smart sparsifying**: threshold (binary) kernels

$$\mathbf{K}_\varepsilon = \{f(\mathbf{K}_{ij})\}_{i,j=1}^n \text{, with } f(t) = \begin{cases} t1_{|t|>s}, & \text{thresholding} \\ \operatorname{sgn}(t)1_{|t|>s}, & \text{thresholding \& binarization.} \end{cases}$$

➽ **"Equi-performance" level comparison** ($50\% =$'phase transition'): subsampling (green), uniform random sparsity (blue) vs. thresholding (red), here for $n/p = 2$.



5. Using Random Matrices to Study... Random Tensors!

➸ **“Spiked” order-3 tensor model:** (can be generalized to order-d)

$$\mathcal{Y} = \lambda \, x \otimes x \times x + \frac{1}{\sqrt{N}} \mathcal{W}.$$

for $x \in \mathbb{R}^n$ and $\mathcal{W}_{ijk} \sim \mathcal{N}(0, \sigma^2_{ijk})$ symmetric ($\sigma^2 = 1$ if i, j, k distinct).

Spiked models for random tensors

➳ **"Spiked" order-3 tensor model:** (can be generalized to order-d)

$$\mathcal{Y} = \lambda\, x \otimes x \times x + \frac{1}{\sqrt{N}} \mathcal{W}.$$

for $x \in \mathbb{R}^n$ and $\mathcal{W}_{ijk} \sim \mathcal{N}(0, \sigma^2_{ijk})$ symmetric $(\sigma^2 = 1$ if i, j, k distinct$)$.

➳ **Objective:** Solve the best rank-1 approximation (" tensor eigenvalue/eigenvector") problem:

$$\operatorname{argmin}_{\mu \in \mathbb{R},\, u \in \mathbb{S}^{N-1}} \|\mathcal{Y} - \mu\, u \otimes u \otimes u\|_F^2$$

➽ **"Spiked" order-3 tensor model:** (can be generalized to order-d)

$$\mathcal{Y} = \lambda\, x \otimes x \times x + \frac{1}{\sqrt{N}}\mathcal{W}.$$

for $x \in \mathbb{R}^n$ and $\mathcal{W}_{ijk} \sim \mathcal{N}(0, \sigma_{ijk}^2)$ symmetric ($\sigma^2 = 1$ if i, j, k distinct).

➽ **Objective:** Solve the best rank-1 approximation (" tensor eigenvalue/eigenvector") problem:

$$\operatorname{argmin}_{\mu \in \mathbb{R},\, u \in \mathbb{S}^{N-1}} \|\mathcal{Y} - \mu\, u \otimes u \otimes u\|_F^2$$

➽ **Key (obvious but super fundamental!) remark:** for μ, u ($\|u\| = 1$) as above,

$$\mathcal{Y}(u)u = \mu u, \qquad \mathcal{Y}(a) = \sum_{i,j} \mathcal{Y}_{ijk} a_i \in \mathbb{R}^{n \times n}$$

Spiked models for random tensors

➽ **“Spiked” order-3 tensor model:** (can be generalized to order-d)

$$\mathcal{Y} = \lambda\, x \otimes x \times x + \frac{1}{\sqrt{N}} \mathcal{W}.$$

for $x \in \mathbb{R}^n$ and $\mathcal{W}_{ijk} \sim \mathcal{N}(0, \sigma^2_{ijk})$ symmetric ($\sigma^2 = 1$ if i, j, k distinct).

➽ **Objective:** Solve the best rank-1 approximation (“ tensor eigenvalue/eigenvector”) problem:

$$\mathrm{argmin}_{\mu \in \mathbb{R},\, u \in \mathbb{S}^{N-1}} \|\mathcal{Y} - \mu\, u \otimes u \otimes u\|_F^2$$

➽ **Key (obvious but super fundamental!) remark:** for μ, u ($\|u\| = 1$) as above,

$$\mathcal{Y}(u)u = \mu u, \qquad \mathcal{Y}(a) = \sum_{i,j} \mathcal{Y}_{ijk} a_i \in \mathbb{R}^{n \times n}$$

⇒ This is **random matrix** spiked model!

➳ **Technical idea:** study the random matrix $\mathcal{Y}(u)$ through resolvent

$$Q(z) = \left(\mathcal{Y}(u) - zI_n\right)^{-1}$$

➸ **Technical idea:** study the random matrix $\boldsymbol{\mathcal{Y}}(u)$ through resolvent

$$Q(z) = \left(\boldsymbol{\mathcal{Y}}(u) - zI_n\right)^{-1}$$

But strong dependence in the entries of $\boldsymbol{\mathcal{Y}}(u)$!

Spiked models for random tensors

➽ **Technical idea:** study the random matrix $\mathcal{Y}(u)$ through resolvent

$$Q(z) = \left(\mathcal{Y}(u) - zI_n\right)^{-1}$$

But strong dependence in the entries of $\mathcal{Y}(u)$!
➾ RMT tools can cope with it! (Stein and Nash-Poincaré method)

Spiked models for random tensors

➽ **Technical idea:** study the random matrix $\mathcal{Y}(u)$ through resolvent

$$Q(z) = \left(\mathcal{Y}(u) - zI_n\right)^{-1}$$

But strong dependence in the entries of $\mathcal{Y}(u)$!
⇨ RMT tools can cope with it! (Stein and Nash-Poincaré method)

Theorem (Spike Asymptotics **[Goulard,Couillet,Comon'21]**)

For $\lambda > \lambda_c$ (λ_c hard to identify...),

$$\mu \xrightarrow{\text{a.s.}} \mu^\infty(\lambda) = \phi(\mu^\infty(\lambda), \lambda)$$
$$|\langle u, x\rangle| \xrightarrow{\text{a.s.}} \alpha(\mu^\infty(\lambda), \lambda)$$

➽ **Technical idea:** study the random matrix $\mathcal{Y}(u)$ through resolvent

$$Q(z) = \left(\mathcal{Y}(u) - zI_n\right)^{-1}$$

But strong dependence in the entries of $\mathcal{Y}(u)$!
➩ RMT tools can cope with it! (Stein and Nash-Poincaré method)

Theorem (Spike Asymptotics [Goulard,Couillet,Comon'21])

For $\lambda > \lambda_c$ (λ_c hard to identify...),

$$\mu \xrightarrow{\text{a.s.}} \mu^\infty(\lambda) = \phi(\mu^\infty(\lambda), \lambda)$$
$$|\langle u, x\rangle| \xrightarrow{\text{a.s.}} \alpha(\mu^\infty(\lambda), \lambda)$$

where

$$\phi(\mu, \lambda) = \frac{\mu^2 - 4 - 4\,h(\mu/2)\,\lambda(\alpha(\mu,\lambda))^3}{-\mu/2 - h(\mu/2)},$$
$$\alpha(\mu, \lambda) = \frac{1}{\lambda}\,\frac{(h(\mu)+\mu)(h(\mu/2)+\mu/2) - 2/3}{\mu + h(\mu) - \mu/2 + h(\mu/2)},$$
$$h(\mu) = \sqrt{\mu^2 - 2/3}.$$

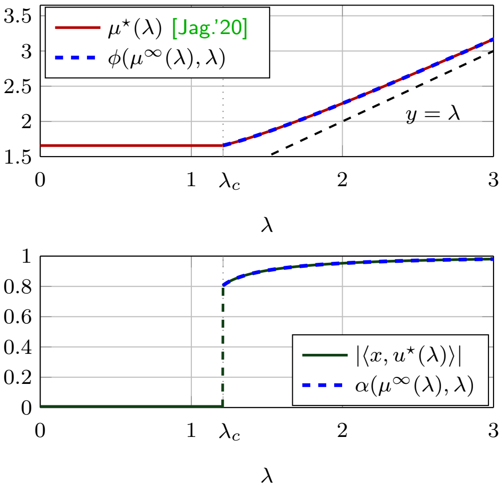


Figure: Our result vs. phy-stat results (with phase transition!) [Jagannath'20].

➳ **Remark**: existence of $\mu/2$ guarantees uniqueness... apparently!

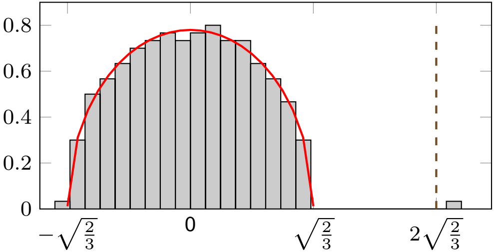


Figure: Eigenvalues of $\mathcal{Y}(u)$: spike always found **beyond** $2 \times \sqrt{\frac{2}{3}}$.

Takeaway Message 3

"RMT Also Grasps 'Real Data' Processing"

Beyond Gaussian Mixtures: results still valid for **concentrated random vectors.**

Beyond Gaussian Mixtures: results still valid for **concentrated random vectors.**

Definition (Concentrated Random Vector)

$x \in \mathbb{R}^p$ is concentrated if, for all Lipschitz $f : \mathbb{R}^p \to \mathbb{R}$, there exists $m_f \in \mathbb{R}$, such that

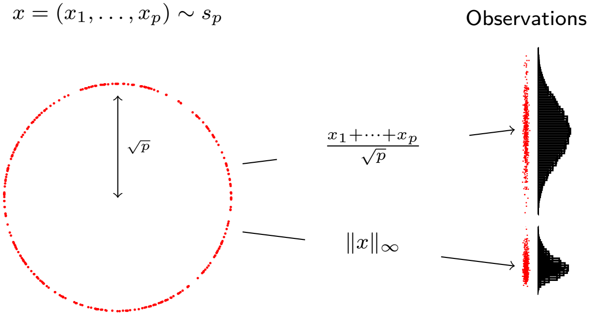
$$P\left(|f(x) - m_f| > \varepsilon\right) \leq e^{-g(\varepsilon)}, \quad g \text{ increasing function.}$$

Beyond Gaussian Mixtures: results still valid for **concentrated random vectors.**

Definition (Concentrated Random Vector)

$x \in \mathbb{R}^p$ is concentrated if, for all Lipschitz $f : \mathbb{R}^p \to \mathbb{R}$, there exists $m_f \in \mathbb{R}$, such that

$$P\left(|f(x) - m_f| > \varepsilon\right) \leq e^{-g(\varepsilon)}, \quad g \text{ increasing function.}$$



Theorem (**[Louart,C'18] [Seddik,C'19]** Kernel Universality)

For $x_i \sim \mathcal{L}(\mu_a, C_a)$ **concentrated random vector**, *under the conditions of* **[C-Benaych'16]**,

$$\|K - \hat{K}\| \xrightarrow{\text{a.s.}} 0, \quad \hat{K} = f(\tau)1_n 1_n^\mathsf{T} + f'(\tau)\frac{1}{p}ZZ^\mathsf{T} + JAJ^\mathsf{T} + *$$

with A only dependent on $f(\tau), f'(\tau), f''(\tau)$, $\mu_1, \ldots, \mu_k$, $C_1, \ldots, C_k$.

Theorem ([Louart,C'18] [Seddik,C'19] Kernel Universality)

For $x_i \sim \mathcal{L}(\mu_a, C_a)$ **concentrated random vector**, *under the conditions of* **[C-Benaych'16]**,

$$\|K - \hat{K}\| \xrightarrow{\text{a.s.}} 0, \quad \hat{K} = f(\tau)1_n 1_n^\mathsf{T} + f'(\tau)\frac{1}{p}ZZ^\mathsf{T} + JAJ^\mathsf{T} + *$$

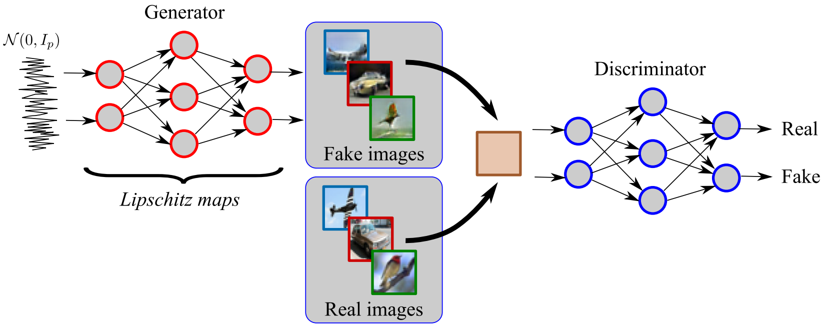
with A only dependent on $f(\tau), f'(\tau), f''(\tau)$, $\mu_1, \ldots, \mu_k$, $C_1, \ldots, C_k$.

⇨ **Same result as [C-Benaych'16]... Universality of first two moments!**

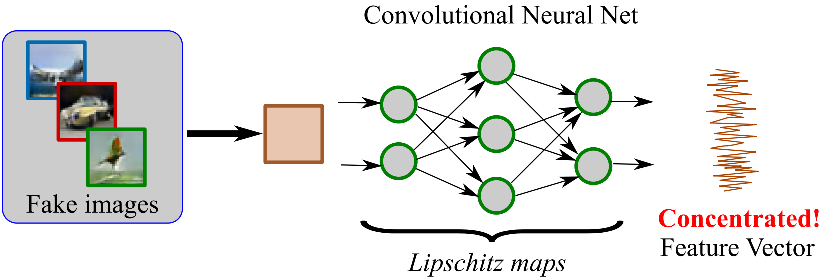
Ok. . . so what?

Key Finding. GAN-generated data **are concentrated random vectors!**

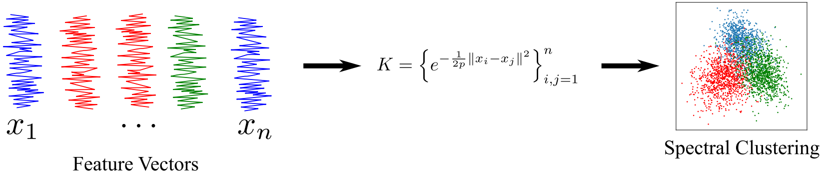
Key Finding. GAN-generated data **are concentrated random vectors!**



Ok. . . so what?


Fake images
Convolutional Neural Net
Lipschitz maps
Concentrated!
Feature Vector

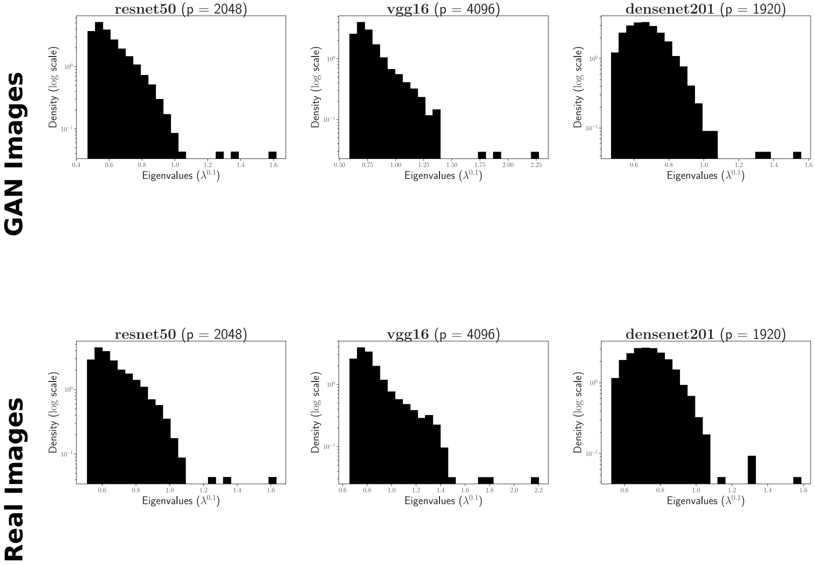
Ok. . . so what?



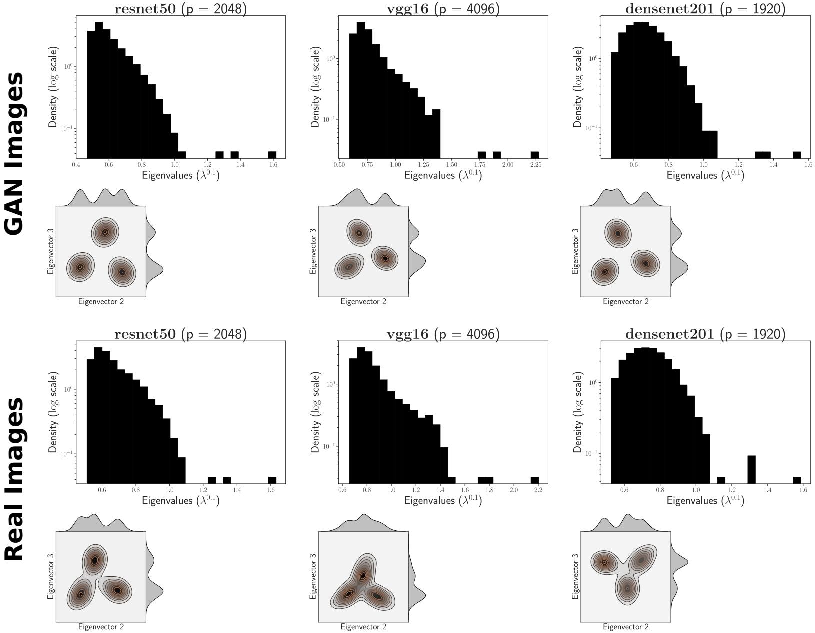
x_1 $\cdots$ x_n
Feature Vectors
$K = \left\{ e^{-\frac{1}{2p}\|x_i - x_j\|^2} \right\}_{i,j=1}^{n}$
Spectral Clustering

Results. [Seddik,C'19]

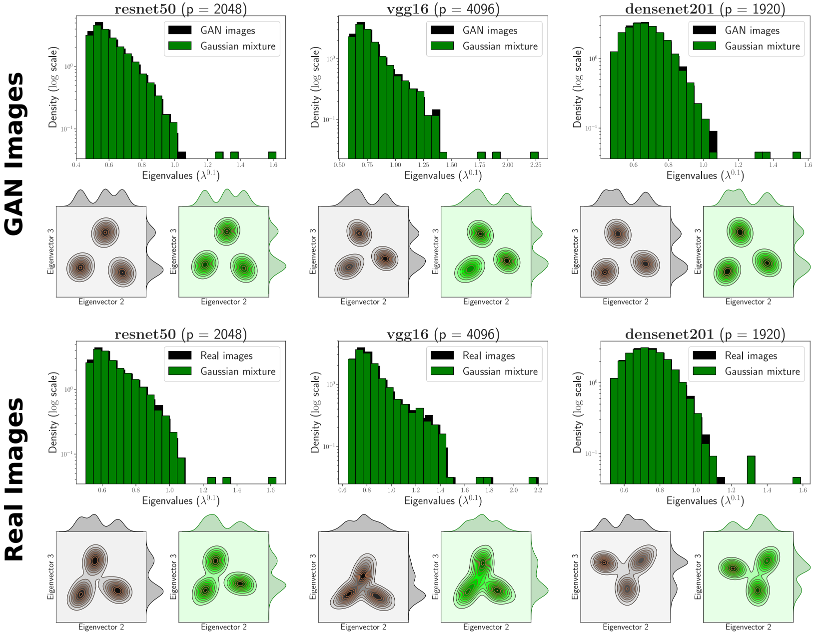
Gaussian, GAN, and real data



Gaussian, GAN, and real data

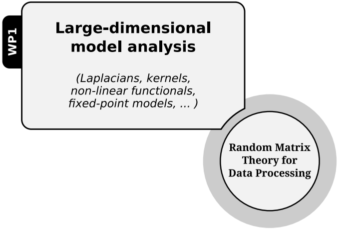


Gaussian, GAN, and real data

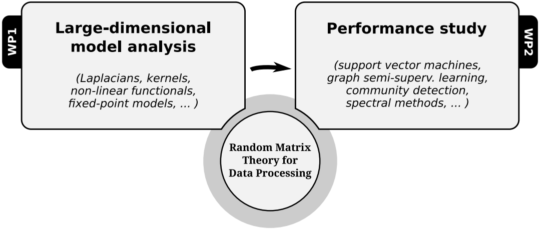


Our Research Activities:

Our Research Activities:

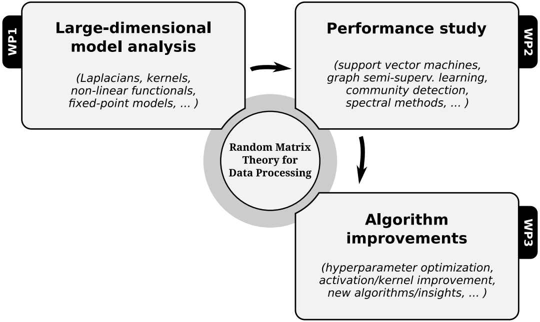


Our Research Activities:

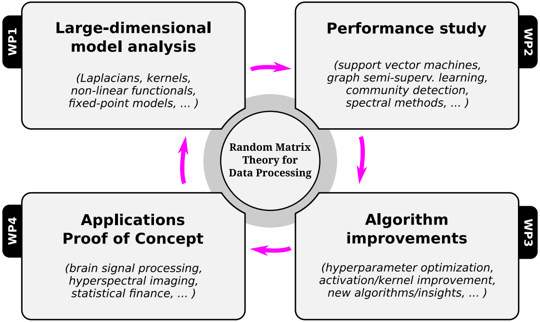


Conclusion

Our Research Activities:



Our Research Activities:



Our Team: the MIAI "LargeDATA" chair @ University Grenoble-Alpes

G. Besson
Institut Fourier
géométrie

F. Chatelain
GIPSA
statistiques

P. Comon
GIPSA
tenseurs

E. Gaussier
LIG
traitement langage

N. Le Bihan
GIPSA
stats, physique

N. Tremblay
GIPSA
graphes

S. Zozor
GIPSA
théorie de l'info

O. Michel
GIPSA
signal, physique

M. Seddik
Apprentissage
applis vision

L. Dall'Amico
Physique Stats
graphes

C. Louart
Mathématiques
concentration

M. Tiomoko
Apprentissage
transfer, SSL

H. Chakroun
Mathématiques
géométrie

C. Doz
Apprentissage
RMT et radar

T. Zarrouk
Apprentissage
RMT structuré

C. Séjourné
Apprentissage
RMT non convexe

B. Nabet
Finance
ML & fi-stats

H. Goulart
Trait. signal
tenseurs

Our Team: the MIAI "LargeDATA" chair @ University Grenoble-Alpes

G. Besson	*Institut Fourier*	*géométrie*
F. Chatelain	*GIPSA*	*statistiques*
P. Comon	*GIPSA*	*tenseurs*
E. Gaussier	*LIG*	*traitement langage*
N. Le Bihan	*GIPSA*	*stats, physique*
N. Tremblay	*GIPSA*	*graphes*
S. Zozor	*GIPSA*	*théorie de l'info*
O. Michel	*GIPSA*	*signal, physique*

M. Seddik	*Apprentissage*	*applis vision*
L. Dall'Amico	*Physique Stats*	*graphes*
C. Louart	*Mathématiques*	*concentration*
M. Tiomoko	*Apprentissage*	*transfer, SSL*
H. Chakroun	*Mathématiques*	*géométrie*
C. Doz	*Apprentissage*	*RMT et radar*
T. Zarrouk	*Apprentissage*	*RMT structuré*
C. Séjourné	*Apprentissage*	*RMT non convexe*
B. Nabet	*Finance*	*ML & fi-stats*
H. Goulart	*Trait. signal*	*tenseurs*

Join us !

Thank you!

- **[C',Chatelain,LeBihan'21]** R. Couillet, F. Chatelain, N. Le Bihan, "Two-way kernel matrix puncturing: towards resource-efficient PCA and spectral clustering", International Conference on Machine Learning (ICML'21), virtual conference, 2021. [article]
- **[Tiomoko,Tiomoko-C'21]** M. Tiomoko, H. Tiomoko, R. Couillet, "Deciphering and Optimizing Multi-Task and Transfer Learning: a Random Matrix Approach", International Conference on Learning Representations (ICLR'21), virtual conference, 2021. **Spotlight article.** [article]
- **[Liao,C,Mahoney'21]** Z. Liao, R. Couillet, M. Mahoney, "Sparse Quantized Spectral Clustering", International Conference on Learning Representations (ICLR'21), virtual conference, 2021. **Spotlight article.** [article]
- **[C-Benaych'16]** R. Couillet, Benaych-Georges, "Kernel Spectral Clustering of Large Dimensional Data", Electronic Journal of Statistics, vol. 10, no. 1, pp. 1393-1454, 2016. [article]
- **[Mai,C'21]** X. Mai, R. Couillet, "Consistent Semi-Supervised Graph Regularization for High Dimensional Data", (to appear) Journal of Machine Learning Research, 2021. [article]
- **[Louart,C'18]** C. Louart, Z. Liao, R. Couillet, "A Random Matrix Approach to Neural Networks", The Annals of Applied Probability, vol. 28, no. 2, pp. 1190-1248, 2018. [article]
- **[Seddik,C'19]** M. Seddik, M. Tamaazousti, R. Couillet, "Kernel Random Matrices of Large Concentrated Data: The Example of GAN-Generated Image", IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'19), Brighton, UK, 2019. [article]
- R. Couillet, M. Tiomoko, S. Zozor, E. Moisan, "Random matrix-improved estimation of covariance matrix distances", Journal of Multivariate Analysis, vol. 174, pp. 104531, 2019. [article]
- Z. Liao, R. Couillet, "A Large Dimensional Analysis of Least Squares Support Vector Machines", IEEE Transactions on Signal Processing, vol. 67, no.4, pp. 1065-1074, 2018. [article]