Mutual Transfer Learning for Massive Data

Ching-Wei Cheng[1], **Xingye Qiao**[2*] and Guang Cheng[1]

1. Department of Statistics, Purdue University
2. Department of Mathematical Sciences, Binghamton University

ICML 2020

Mutual Transfer Learning?

- Data from a new domain (labeled or unlabeled) are too limited.

Mutual Transfer Learning?

- Data from a new domain (labeled or unlabeled) are too limited.
  - **Transfer learning**: use abundant data from a source domain to improve the learning performance (including prediction and inference) for a target domain.

Mutual Transfer Learning?

- Data from a new domain (labeled or unlabeled) are too limited.
  - **Transfer learning**: use abundant data from a source domain to improve the learning performance (including prediction and inference) for a target domain.
  - Typically, the target and the source domains are known and fixed.

Mutual Transfer Learning?

- Data from a new domain (labeled or unlabeled) are too limited.
  - **Transfer learning**: use abundant data from a source domain to improve the learning performance (including prediction and inference) for a target domain.
  - Typically, the target and the source domains are known and fixed.
- In this paper, every data domain could potentially be the target of interest, and it could also be a useful source to help the learning in other data domains - **mutual transfer learning**.

What makes it interesting?

- Given a target domain, not every domain can be a successful source; only data sets that are similar enough to be thought as from the same population are useful sources for each other

What makes it interesting?

- Given a target domain, not every domain can be a successful source; only data sets that are similar enough to be thought as from the same population are useful sources for each other
  - Suggests a **mutual learnability** structure

What makes it interesting?

- Given a target domain, not every domain can be a successful source; only data sets that are similar enough to be thought as from the same population are useful sources for each other
  - Suggests a **mutual learnability** structure
- *How to identify useful sources?*

What makes it interesting?

- Given a target domain, not every domain can be a successful source; only data sets that are similar enough to be thought as from the same population are useful sources for each other
  - Suggests a **mutual learnability** structure
- *How to identify useful sources?*
- A confidence distribution (CD) fusion approach is proposed to recover such **mutual learnability** relation in the transfer learning regime

What makes it interesting?

- Given a target domain, not every domain can be a successful source; only data sets that are similar enough to be thought as from the same population are useful sources for each other
  - Suggests a **mutual learnability** structure
- *How to identify useful sources?*
- A confidence distribution (CD) fusion approach is proposed to recover such **mutual learnability** relation in the transfer learning regime
  - Achieves the same oracle statistical inferential accuracy as if the true mutual learnability structure were known.

What makes it interesting?

- Given a target domain, not every domain can be a successful source; only data sets that are similar enough to be thought as from the same population are useful sources for each other
  - Suggests a **mutual learnability** structure
- *How to identify useful sources?*
- A confidence distribution (CD) fusion approach is proposed to recover such **mutual learnability** relation in the transfer learning regime
  - Achieves the same oracle statistical inferential accuracy as if the true mutual learnability structure were known.
  - Implemented in an efficient parallel fashion to deal with large-scale data.

Big Climate Data

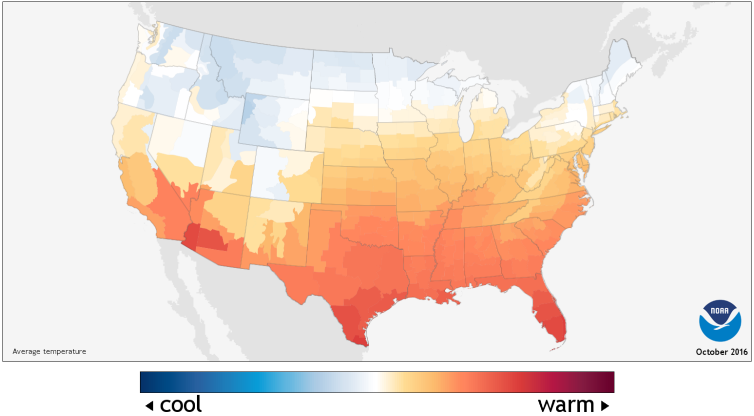
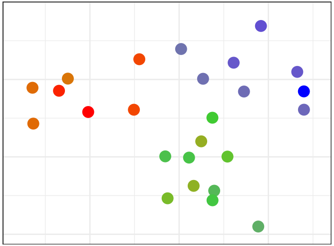


Figure: U.S. average temperature map in October, 2016.

- 503,616 monthly observations from 344 climate divisions (data units) from January 1895 to December 2016

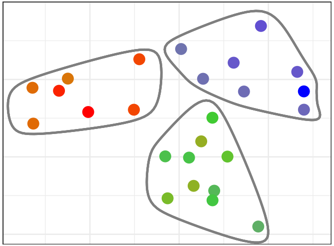
Big Data with Two-Layer Heterogeneity

Big Data typically consists of multiple datasets ("data units") that are collected in different time periods, at different locations and using different approaches



Big Data with Two-Layer Heterogeneity

Big Data typically consists of multiple datasets ("data units") that are collected in different time periods, at different locations and using different approaches



Two-layer heterogeneity:

- 1^{st} layer: Subpopulation heterogeneity
- 2^{nd} layer: Within-subpopulation heterogeneity
  (Units are still different within subpopulations)

In this work, we propose a Mutual Transfer Learning (MTL) model

Goals:

Model:

Method:

In this work, we propose a Mutual Transfer Learning (MTL) model

Goals:

- Mutual learnability structure recovery (which domains are useful?)
- The best possible statistical estimation and inference
- Scalable for massive data

Model:

Method:

In this work, we propose a Mutual Transfer Learning (MTL) model

Goals:

- Mutual learnability structure recovery (which domains are useful?)
- The best possible statistical estimation and inference
- Scalable for massive data

Model:

- MTL is based on linear mixed-effects model (LMM) using regression as examples
- MTL can be easily generalized to other response data types

Method:

In this work, we propose a Mutual Transfer Learning (MTL) model

Goals:

- Mutual learnability structure recovery (which domains are useful?)
- The best possible statistical estimation and inference
- Scalable for massive data

Model:

- MTL is based on linear mixed-effects model (LMM) using regression as examples
- MTL can be easily generalized to other response data types

Method:

- Confidence distribution (CD) fusion approach

Outline

MTL: Unit Level

LMM for the i-th data unit

$$\underset{n_i \times 1}{\boldsymbol{y}_i} = \underset{n_i \times p}{\boldsymbol{x}_i} \boldsymbol{\beta} + \underset{n_i \times q}{\boldsymbol{z}_i} (\boldsymbol{\theta}_i + \boldsymbol{u}_i) + \boldsymbol{\varepsilon}_i$$

- $\boldsymbol{\beta} \in \mathbb{R}^p$ is the coefficients for the global feature vector $\boldsymbol{x}_i$
- $\boldsymbol{\theta}_i \in \mathbb{R}^q$ is the coefficients for heterogeneous feature vector $\boldsymbol{z}_i$
- $\boldsymbol{u}_i$ is the unit-specific random effect with $\mathsf{E}[\boldsymbol{u}_i] = \mathbf{0}$ and $\mathsf{Cov}(\boldsymbol{u}_i) = \sigma_u^2 \boldsymbol{I}$
- $\boldsymbol{\varepsilon}_i$ is the error vector with $\mathsf{E}[\boldsymbol{\varepsilon}_i] = \mathbf{0}$ and $\mathsf{Cov}(\boldsymbol{\varepsilon}_i) = \sigma_\varepsilon^2 \boldsymbol{I}$

MTL: Subpopulation Level

LMM for the i-th data unit in the s-th subpopulation

$$\underset{n_i \times 1}{\boldsymbol{y}_i} = \underset{n_i \times p}{\boldsymbol{x}_i} \boldsymbol{\beta} + \underset{n_i \times q}{\boldsymbol{z}_i} (\boldsymbol{\alpha}_s + \boldsymbol{u}_i) + \boldsymbol{\varepsilon}_i$$

- $\boldsymbol{\beta} \in \mathbb{R}^p$ is the coefficients for the global feature vector $\boldsymbol{x}_i$
- $\boldsymbol{\theta}_i \in \mathbb{R}^q$ is the coefficients for heterogeneous feature vector $\boldsymbol{z}_i$
  - Assume $\boldsymbol{\theta}_i \equiv \boldsymbol{\alpha}_s$ if unit i belongs to subpopulation s
    $\Rightarrow$ Need to reveal learnability structure
- $\boldsymbol{u}_i$ is the unit-specific random effect with $\mathsf{E}[\boldsymbol{u}_i] = \mathbf{0}$ and $\mathsf{Cov}(\boldsymbol{u}_i) = \sigma_u^2 \boldsymbol{I}$
  - Within-subpopulation heterogeneity
- $\boldsymbol{\varepsilon}_i$ is the error vector with $\mathsf{E}[\boldsymbol{\varepsilon}_i] = \mathbf{0}$ and $\mathsf{Cov}(\boldsymbol{\varepsilon}_i) = \sigma_\varepsilon^2 \boldsymbol{I}$

THEM: Matrix Form

Matrix form with M data units
($N := \sum_{i=1}^{M} n_i$)

$$
\begin{array}{ccccccc}
\boldsymbol{Y} & = & \boldsymbol{X} & \boldsymbol{\beta} + & \boldsymbol{Z} & (\quad \boldsymbol{\Theta} \quad + \quad \boldsymbol{U} \quad) + & \boldsymbol{\mathcal{E}} \\
\begin{pmatrix} \boldsymbol{y}_1 \\ \vdots \\ \boldsymbol{y}_M \end{pmatrix} & = & \begin{bmatrix} \boldsymbol{x}_1 \\ \vdots \\ \boldsymbol{x}_M \end{bmatrix} & \boldsymbol{\beta} + & \begin{bmatrix} \boldsymbol{z}_1 & & \\ & \ddots & \\ & & \boldsymbol{z}_M \end{bmatrix} & \left(\begin{pmatrix} \boldsymbol{\theta}_1 \\ \vdots \\ \boldsymbol{\theta}_M \end{pmatrix} + \begin{pmatrix} \boldsymbol{u}_1 \\ \vdots \\ \boldsymbol{u}_M \end{pmatrix} \right) + & \begin{pmatrix} \boldsymbol{\varepsilon}_1 \\ \vdots \\ \boldsymbol{\varepsilon}_M \end{pmatrix} \\
N \times 1 & & N \times p & & N \times Mq & Mq \times 1 &
\end{array}
$$

THEM: Matrix Form

Matrix form with M data units coming from S subpopulations (**oracle**) $(N := \sum_{i=1}^{M} n_i)$

$$
\begin{array}{ccccccc}
\boldsymbol{Y} & = & \boldsymbol{X} & \boldsymbol{\beta} + & \boldsymbol{Z} & (\quad \boldsymbol{\Theta} \quad + \quad \boldsymbol{U} \quad) + & \boldsymbol{\mathcal{E}} \\
\begin{pmatrix} \boldsymbol{y}_1 \\ \vdots \\ \boldsymbol{y}_M \end{pmatrix} & = & \begin{bmatrix} \boldsymbol{x}_1 \\ \vdots \\ \boldsymbol{x}_M \end{bmatrix} & \boldsymbol{\beta} + & \begin{bmatrix} \boldsymbol{z}_1 & & \\ & \ddots & \\ & & \boldsymbol{z}_M \end{bmatrix} & (\quad \boldsymbol{A\alpha} \quad + \begin{pmatrix} \boldsymbol{u}_1 \\ \vdots \\ \boldsymbol{u}_M \end{pmatrix}) + & \begin{pmatrix} \boldsymbol{\varepsilon}_1 \\ \vdots \\ \boldsymbol{\varepsilon}_M \end{pmatrix} \\
N \times 1 & & N \times p & & N \times Mq & Mq \times 1 &
\end{array}
$$

- Exists an (unknown) label matrix $\boldsymbol{A}_{Mq \times Sq}$ such that $\boldsymbol{\Theta} = \boldsymbol{A\alpha}$ with $\boldsymbol{\alpha} = \begin{pmatrix} \boldsymbol{\alpha}_1 \\ \vdots \\ \boldsymbol{\alpha}_S \end{pmatrix}_{Sq \times 1}$
  - $\boldsymbol{\theta}_i \equiv \boldsymbol{\alpha}_s$ if unit i belongs to subpopulation s
  - Only S different values of $\boldsymbol{\theta}_i$'s

A Naive Full-Data Estimator

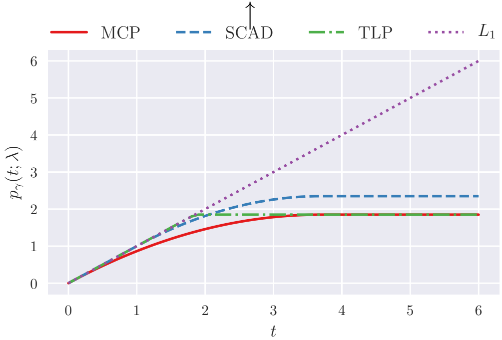
$$\begin{pmatrix} \widehat{\boldsymbol{\beta}}(\lambda) \\ \widehat{\boldsymbol{\Theta}}(\lambda) \end{pmatrix} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p, \boldsymbol{\Theta}\in\mathbb{R}^{Mq}}{\arg\min} \; Q_N(\boldsymbol{\beta}, \boldsymbol{\Theta}) \text{ where}$$

$$Q_N(\boldsymbol{\beta}, \boldsymbol{\Theta}) = \left\{ \underbrace{\frac{1}{2} \sum_{i=1}^{M} (\boldsymbol{y}_i - \boldsymbol{x}_i\boldsymbol{\beta} - \boldsymbol{z}_i\boldsymbol{\theta}_i)^\top \boldsymbol{W}_i (\boldsymbol{y}_i - \boldsymbol{x}_i\boldsymbol{\beta} - \boldsymbol{z}_i\boldsymbol{\theta}_i)}_{\text{generalized least squares (GLS) based on full data}} + \underbrace{\sum_{1\le i<j\le M} p_\gamma \left(\|\boldsymbol{\theta}_i - \boldsymbol{\theta}_j\|\,; \lambda\right)}_{\text{pairwise concave fusion penalty}} \right\}$$

- $\boldsymbol{W}_i = \mathrm{Cov}(\boldsymbol{y}_i|\boldsymbol{x}_i, \boldsymbol{z}_i)^{-1} = \left(\sigma_\varepsilon^2 \boldsymbol{I}_{n_i} + \sigma_u^2 \boldsymbol{z}_i \boldsymbol{z}_i^\top\right)^{-1}$
- $\lambda > 0$ is a tuning parameter
- $\gamma > 0$ determines the concavity of the penalty

Concave Penalty Function $p_\gamma(t;\lambda)$

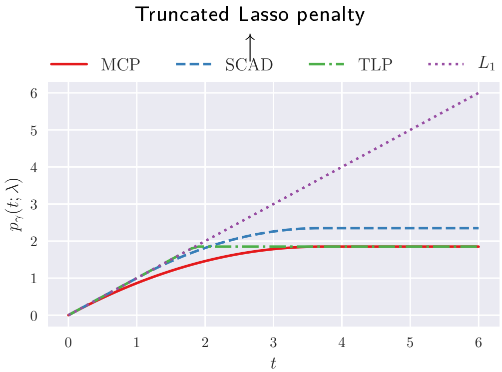
Truncated Lasso penalty



In this graph,

- $\lambda = 1$
- $\gamma = 3.7$ for MCP and SCAD and $\gamma = 1.85$ for TLP

Concave Penalty Function $p_\gamma(t;\lambda)$



In our analysis,

- $\lambda > 0$ is chosen by modified BIC (Wang et al., 2009)
- $\gamma = 3.7$ for MCP and SCAD and $\gamma = 1.85$ for TLP

Computation Barrier

$$Q_N(\boldsymbol{\beta}, \boldsymbol{\Theta}) = \left\{ \frac{1}{2} \sum_{i=1}^{M} (\boldsymbol{y}_i - \boldsymbol{x}_i \boldsymbol{\beta} - \boldsymbol{z}_i \boldsymbol{\theta}_i)^\top \boldsymbol{W}_i (\boldsymbol{y}_i - \boldsymbol{x}_i \boldsymbol{\beta} - \boldsymbol{z}_i \boldsymbol{\theta}_i) + \sum_{1 \le i < j \le M} p_\gamma \left(\|\boldsymbol{\theta}_i - \boldsymbol{\theta}_j\| ; \lambda \right) \right\}$$

- Communication cost: each local machine passes
  - an $n_i \times (p + q + 1)$ data matrix $(\boldsymbol{y}_i, \boldsymbol{x}_i, \boldsymbol{z}_i)$ and
  - an $n_i \times n_i$ weight matrix $\boldsymbol{W}_i$

  to a centralized computer node

▸ Communication cost for CD fusion

Computation Barrier

Replace it using the CD approach of Liu et al. (2015)
to combine unit GLS estimates
↑

$$Q_N(\boldsymbol{\beta}, \boldsymbol{\Theta}) = \left\{ \frac{1}{2} \sum_{i=1}^{M} (\boldsymbol{y}_i - \boldsymbol{x}_i \boldsymbol{\beta} - \boldsymbol{z}_i \boldsymbol{\theta}_i)^\top \boldsymbol{W}_i (\boldsymbol{y}_i - \boldsymbol{x}_i \boldsymbol{\beta} - \boldsymbol{z}_i \boldsymbol{\theta}_i) + \sum_{1 \le i < j \le M} p_\gamma \left(\|\boldsymbol{\theta}_i - \boldsymbol{\theta}_j\| \,; \lambda \right) \right\}$$

- Communication cost: each local machine passes
  - an $n_i \times (p + q + 1)$ data matrix $(\boldsymbol{y}_i, \boldsymbol{x}_i, \boldsymbol{z}_i)$ and
  - an $n_i \times n_i$ weight matrix $\boldsymbol{W}_i$

  to a centralized computer node

▸ Communication cost for CD fusion

Unit GLS Estimates

- Unit GLS estimates are defined as

$$\begin{pmatrix} \widehat{\boldsymbol{\beta}}_i \\ \widehat{\boldsymbol{\theta}}_i \end{pmatrix} = \left[(\boldsymbol{x}_i, \boldsymbol{z}_i)^\top \boldsymbol{W}_i (\boldsymbol{x}_i, \boldsymbol{z}_i) \right]^{-1} (\boldsymbol{x}_i, \boldsymbol{z}_i)^\top \boldsymbol{W}_i \boldsymbol{y}_i \xrightarrow{D} \mathcal{N} \left(\begin{pmatrix} \boldsymbol{\beta}_0 \\ \boldsymbol{\theta}_{i,0} \end{pmatrix}, \underbrace{\left[(\boldsymbol{x}_i, \boldsymbol{z}_i)^\top \boldsymbol{W}_i (\boldsymbol{x}_i, \boldsymbol{z}_i) \right]^{-1}}_{\boldsymbol{\Sigma}_i} \right)$$

  where $\boldsymbol{W}_i = \mathrm{Cov}(\boldsymbol{y}_i | \boldsymbol{x}_i, \boldsymbol{z}_i)^{-1} = \left(\sigma_\varepsilon^2 \boldsymbol{I}_{n_i} + \sigma_u^2 \boldsymbol{z}_i \boldsymbol{z}_i^\top \right)^{-1}$
  - σ_u^2 and σ_ε^2 can be consistently estimated through restricted maximum likelihood (REML) method.
  - For simplicity, we assume σ_u^2 and σ_ε^2 (and thus $\boldsymbol{W}_i$'s) are known

CD Fusion Approach: Unit CD Density

- Unit GLS estimates are defined as

$$\begin{pmatrix} \widehat{\boldsymbol{\beta}}_i \\ \widehat{\boldsymbol{\theta}}_i \end{pmatrix} = \left[(\boldsymbol{x}_i, \boldsymbol{z}_i)^\top \boldsymbol{W}_i (\boldsymbol{x}_i, \boldsymbol{z}_i) \right]^{-1} (\boldsymbol{x}_i, \boldsymbol{z}_i)^\top \boldsymbol{W}_i \boldsymbol{y}_i \xrightarrow{D} \mathcal{N}\left(\begin{pmatrix} \boldsymbol{\beta}_0 \\ \boldsymbol{\theta}_{i,0} \end{pmatrix}, \underbrace{\left[(\boldsymbol{x}_i, \boldsymbol{z}_i)^\top \boldsymbol{W}_i (\boldsymbol{x}_i, \boldsymbol{z}_i) \right]^{-1}}_{\boldsymbol{\Sigma}_i} \right)$$

  where $\boldsymbol{W}_i = \text{Cov}(\boldsymbol{y}_i | \boldsymbol{x}_i, \boldsymbol{z}_i)^{-1} = \left(\sigma_\varepsilon^2 \boldsymbol{I}_{n_i} + \sigma_u^2 \boldsymbol{z}_i \boldsymbol{z}_i^\top \right)^{-1}$
  - σ_u^2 and σ_ε^2 can be consistently estimated through restricted maximum likelihood (REML) method.
  - For simplicity, we assume σ_u^2 and σ_ε^2 (and thus $\boldsymbol{W}_i$'s) are known
- CD density can be assigned by switching the roles of estimator and parameter of interest, i.e., define the unit CD density by

$$h_i(\boldsymbol{\beta}, \boldsymbol{\theta}_i) := \text{density of } \mathcal{N}\left(\begin{pmatrix} \widehat{\boldsymbol{\beta}}_i \\ \widehat{\boldsymbol{\theta}}_i \end{pmatrix}, \boldsymbol{\Sigma}_i \right)$$

CD Fusion Approach: Combined CD Density

- Following Liu et al. (2015), the combined CD density is defined by

$$h(\boldsymbol{\beta}, \boldsymbol{\Theta}) := \prod_{i=1}^{M} h_i(\boldsymbol{\beta}, \boldsymbol{\theta}_i)$$

CD Fusion Approach: Combined CD Density

- Following Liu et al. (2015), the combined CD density is defined by

$$h(\boldsymbol{\beta}, \boldsymbol{\Theta}) := \prod_{i=1}^{M} h_i(\boldsymbol{\beta}, \boldsymbol{\theta}_i)$$

- By omitting additive constant terms, we have

$$-\log h(\boldsymbol{\beta}, \boldsymbol{\Theta}) \propto \sum_{i=1}^{M} \begin{pmatrix} \widehat{\boldsymbol{\beta}}_i - \boldsymbol{\beta} \\ \widehat{\boldsymbol{\theta}}_i - \boldsymbol{\theta}_i \end{pmatrix}^{\top} \boldsymbol{\Sigma}_i^{-1} \begin{pmatrix} \widehat{\boldsymbol{\beta}}_i - \boldsymbol{\beta} \\ \widehat{\boldsymbol{\theta}}_i - \boldsymbol{\theta}_i \end{pmatrix}$$

CD Fusion Estimator

$$\begin{pmatrix} \widehat{\boldsymbol{\beta}}_{\mathrm{CD}}(\lambda) \\ \widehat{\boldsymbol{\Theta}}_{\mathrm{CD}}(\lambda) \end{pmatrix} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p, \boldsymbol{\Theta}\in\mathbb{R}^{Mq}}{\arg\min} \; Q_N^{\mathrm{CD}}(\boldsymbol{\beta}, \boldsymbol{\Theta}) \text{ where}$$

$$Q_N^{\mathrm{CD}}(\boldsymbol{\beta}, \boldsymbol{\Theta}) = -\log \underbrace{h(\boldsymbol{\beta}, \boldsymbol{\Theta})}_{\text{Combined CD density}} + \underbrace{\sum_{1\le i<j\le M} p_\gamma\left(\|\boldsymbol{\theta}_i - \boldsymbol{\theta}_j\|\,;\lambda\right)}_{\text{pairwise concave fusion penalty}}$$

CD Fusion Estimator

$$\begin{pmatrix} \widehat{\boldsymbol{\beta}}_{\mathrm{CD}}(\lambda) \\ \widehat{\boldsymbol{\Theta}}_{\mathrm{CD}}(\lambda) \end{pmatrix} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p, \boldsymbol{\Theta}\in\mathbb{R}^{Mq}}{\arg\min}\ Q_N^{\mathrm{CD}}(\boldsymbol{\beta}, \boldsymbol{\Theta}) \text{ where}$$

$$Q_N^{\mathrm{CD}}(\boldsymbol{\beta}, \boldsymbol{\Theta}) = -\log \underbrace{h(\boldsymbol{\beta}, \boldsymbol{\Theta})}_{\text{Combined CD density}} + \underbrace{\sum_{1\le i<j\le M} p_\gamma\left(\|\boldsymbol{\theta}_i - \boldsymbol{\theta}_j\|\,;\lambda\right)}_{\text{pairwise concave fusion penalty}}$$

- Communication cost: each local machine passes
  - a $(p+q)$-vector $\left(\widehat{\boldsymbol{\beta}}_i^\top, \widehat{\boldsymbol{\theta}}_i^\top\right)^\top$ and
  - a $(p+q)\times(p+q)$ matrix $\boldsymbol{\Sigma}_i$

  to a centralized computer node

◂ Communication cost for the full-data approach

Oracle Estimator

The oracle estimator of $(\boldsymbol{\beta}, \boldsymbol{\alpha})$ is defined by the full-data GLS estimator given the true subpopulations

$$\begin{aligned}\begin{pmatrix}\widehat{\boldsymbol{\beta}}_{\mathrm{OR}}\\ \widehat{\boldsymbol{\alpha}}_{\mathrm{OR}}\end{pmatrix} &= \underset{\boldsymbol{\beta}\in\mathbb{R}^p, \boldsymbol{\alpha}\in\mathbb{R}^{Sq}}{\arg\min}\ \frac{1}{2}(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{Z}\boldsymbol{A}\boldsymbol{\alpha})^\top \boldsymbol{W}(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{Z}\boldsymbol{A}\boldsymbol{\alpha})\\ &= \left[(\boldsymbol{X}, \boldsymbol{Z}\boldsymbol{A})^\top \boldsymbol{W}(\boldsymbol{X}, \boldsymbol{Z}\boldsymbol{A})\right]^{-1}(\boldsymbol{X}, \boldsymbol{Z}\boldsymbol{A})^\top \boldsymbol{W}\boldsymbol{Y}\end{aligned}$$

where $\boldsymbol{W} = \mathrm{diag}\left(\boldsymbol{W}_1, \ldots, \boldsymbol{W}_M\right)$

- $\boldsymbol{A}$ is **unknown** in reality
- Not computable with massive sample size

Theoretical Guarantees

Regularity conditions on

- Random design matrices (sub-Gaussian tails and eigenvalue restrictions)
- Sub-Gaussian tails for random effects $\boldsymbol{U}$ and noises $\boldsymbol{\mathcal{E}}$
- Concave fusion penalty (satisfied by MCP, SCAD and TLP)

Theoretical Guarantees

Regularity conditions on

- Random design matrices (sub-Gaussian tails and eigenvalue restrictions)
- Sub-Gaussian tails for random effects $\boldsymbol{U}$ and noises $\boldsymbol{\mathcal{E}}$
- Concave fusion penalty (satisfied by MCP, SCAD and TLP)

Nice properties

Theorem 4.1

(Given S does not grow too fast)

$\uparrow$

CD Fusion Estimator $\approx$ Oracle Estimator

$\uparrow$

Asymptotic equivalence

Theoretical Guarantees

Regularity conditions on

- Random design matrices (sub-Gaussian tails and eigenvalue restrictions)
- Sub-Gaussian tails for random effects $\boldsymbol{U}$ and noises $\boldsymbol{\mathcal{E}}$
- Concave fusion penalty (satisfied by MCP, SCAD and TLP)

Nice properties

Theorem 4.1

(Given S does not grow too fast)

↑

CD Fusion Estimator $=$ Full-Data Estimator $\approx$ Oracle Estimator

↑ (under $=$): Exact

Theorem 2.1

↑ (under $\approx$): Asymptotic equivalence

Theorem 4.2

(Given a minimal signal condition)

Revisiting Our Goals

- Mutual learnability structure recovery

Sol: Pairwise fusion penalty to fuse unit level $\boldsymbol{\beta}_i$'s
Theoretical guarantees, provided that S does not grow too fast and a minimal signal condition

Revisiting Our Goals

- Mutual learnability structure recovery

Sol: Pairwise fusion penalty to fuse unit level $\boldsymbol{\beta}_i$'s
Theoretical guarantees, provided that S does not grow too fast and a minimal signal condition

- Accurate estimation and inference

Sol: Achieves the oracle level

Revisiting Our Goals

- Mutual learnability structure recovery

Sol: Pairwise fusion penalty to fuse unit level $\boldsymbol{\beta}_i$'s
Theoretical guarantees, provided that S does not grow too fast and a minimal signal condition

- Accurate estimation and inference

Sol: Achieves the oracle level

- Computable approach for massive data

Sol: CD approach to combine unit estimates
ADMM with parallel computing

Summary of simulation studies:

- The CD fusion approach behaves desirably with MCP, SCAD and TLP

Numerical Results

Summary of simulation studies:

- The CD fusion approach behaves desirably with MCP, SCAD and TLP
- MCP is recommended in general
  - Decent and stable performance
  - Fast (only slightly slower than SCAD)

Numerical Results

Summary of simulation studies:

- The CD fusion approach behaves desirably with MCP, SCAD and TLP
- MCP is recommended in general
  - Decent and stable performance
  - Fast (only slightly slower than SCAD)
- SCAD and TLP are unstable in some cases
- L_1 “fails” in all cases

Real Data Example: NOAA[1]'s nClimDiv

- Time period chosen: January 1895 to December 2016
- $N = 503{,}616$ observations from $M = 344$ climate divisions (data units)

[1] National Oceanic and Atmospheric Administration

Real Data Example: NOAA[1]'s nClimDiv

- Time period chosen: January 1895 to December 2016
- $N = 503{,}616$ observations from $M = 344$ climate divisions (data units)
- Response: monthly average temperature

[1] National Oceanic and Atmospheric Administration

Real Data Example: NOAA[1]'s nClimDiv

- Time period chosen: January 1895 to December 2016
- $N = 503{,}616$ observations from $M = 344$ climate divisions (data units)
- Response: monthly average temperature
- 8 candidate covariates

How to choose global features

  - $p = 5$ covariates as global effects $\boldsymbol{\beta}$
    - 3 dummy variables for seasonal effects: Summer, Fall and Winter
    - Palmer Drought Severity Index (PSDI)
    - Palmer Hydrological Drought Index (PHDI)
  - $q = 3$ covariates as heterogeneous effects $\boldsymbol{\theta}_i$'s
    - Intercept
    - Precipitation (PCPN)
    - Palmer Z Index (ZNDX)

[1] National Oceanic and Atmospheric Administration

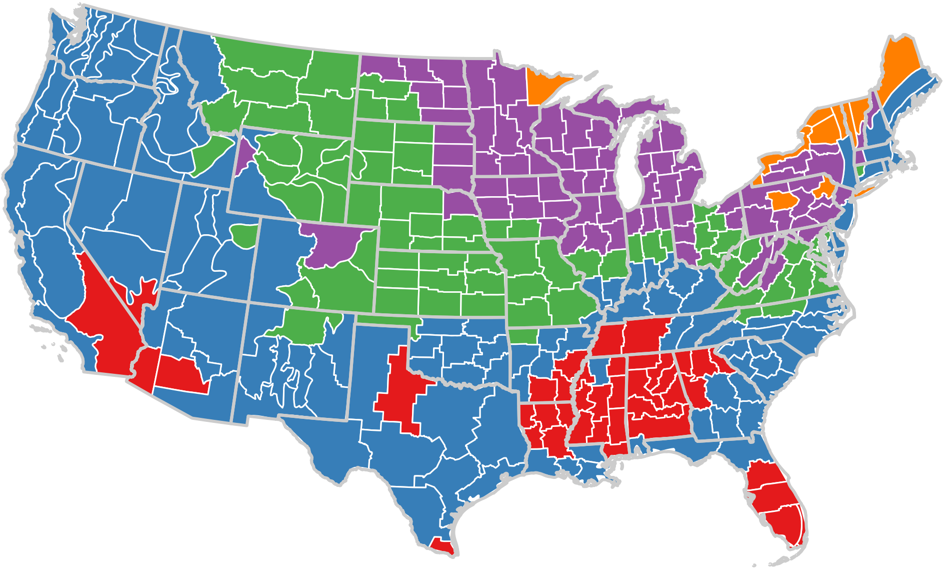
Real Data Example: NOAA[1]'s nClimDiv

- Time period chosen: January 1895 to December 2016
- $N = 503{,}616$ observations from $M = 344$ climate divisions (data units)
- Response: monthly average temperature
- 8 candidate covariates
  - $p = 5$ covariates as global effects $\boldsymbol{\beta}$
    - 3 dummy variables for seasonal effects: Summer, Fall and Winter
    - Palmer Drought Severity Index (PSDI)
    - Palmer Hydrological Drought Index (PHDI)
  - $q = 3$ covariates as heterogeneous effects $\boldsymbol{\theta}_i$'s
    - Intercept
    - Precipitation (PCPN)
    - Palmer Z Index (ZNDX)
- Only MCP is used in analysis

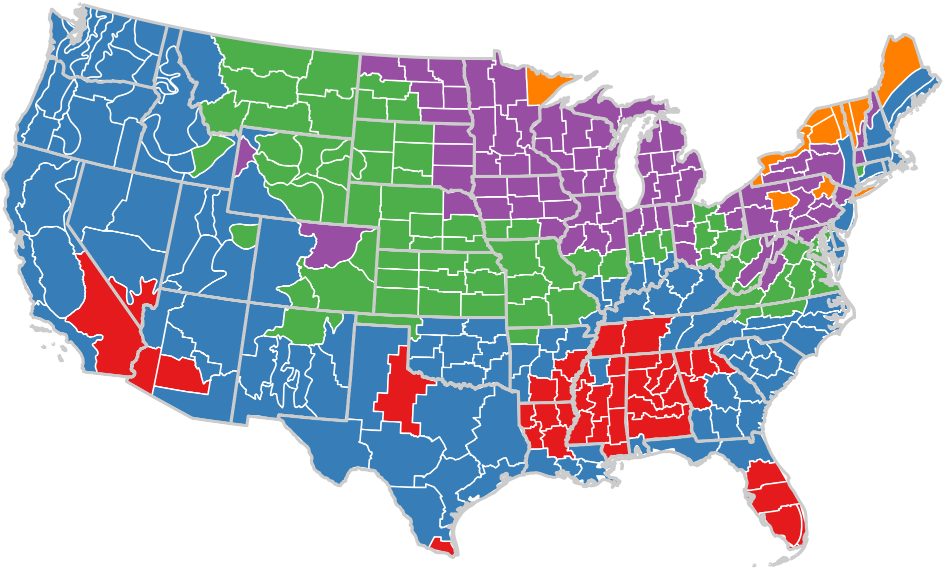
▸ How to choose global features

[1] National Oceanic and Atmospheric Administration

Real Data Example: Estimated Subpopulations ($\widehat{S} = 5$)

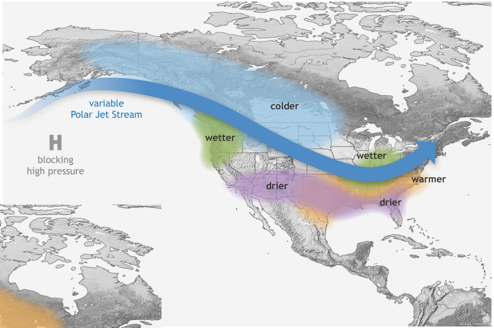


Real Data Example: Estimated Subpopulations ($\widehat{S} = 5$)

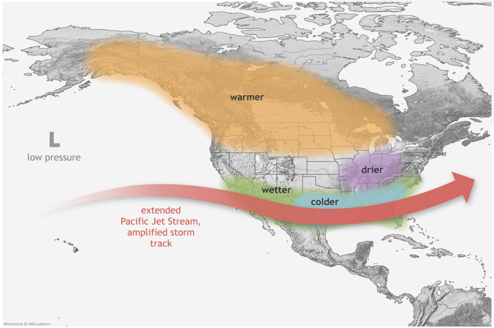


Wintertime ENSO Patterns

La Niña Winter Pattern



El Niño Winter Pattern



Real Data Example: Estimated Subpopulations and ENSO

Subpopulation [#(units)]	Corresponding ENSO Pattern
Red [41] and **Blue** [132]	Drier area in La Niña
Green [79]	Transition between wetter and drier in El Niño
Purple [81]	Drier area in El Niño

Orange [11] subpopulation is particularly curious cases...

- Extreme weather?

Real Data Example: THEM Estimates

Subpopulation Color [#(units)]	**Subpopulation Effects**		
	$\widehat{\alpha}_{\text{Intercept}}$	$\widehat{\alpha}_{\text{PCPN}}$	$\widehat{\alpha}_{\text{ZNDX}}$
Red [41]	64.97 (0.1320)	−0.37 (0.0952)	−0.07 (0.0954)
Blue [132]	49.53 (0.0714)	0.85 (0.0539)	−1.51 (0.0531)
Green [79]	35.32 (0.0891)	5.44 (0.0698)	−4.05 (0.0682)
Purple [81]	24.74 (0.0926)	7.28 (0.0686)	−5.16 (0.0675)
Orange [11]	9.90 (0.3232)	9.14 (0.1932)	−6.54 (0.1864)

Common Effects

$\widehat{\beta}_{\text{Summer}}$	$\widehat{\beta}_{\text{Fall}}$	$\widehat{\beta}_{\text{Winter}}$	$\widehat{\beta}_{\text{PDSI}}$	$\widehat{\beta}_{\text{PHDI}}$
18.26 (0.0261)	4.06 (0.0258)	−15.12 (0.0271)	0.18 (0.0098)	0.20 (0.0084)

Selected References

Liu, D., Liu, R. Y., and Xie, M. (2015), "Multivariate Meta-Analysis of Heterogeneous Studies Using Only Summary Statistics: Efficiency and Robustness," *Journal of the American Statistical Association*, 110, 326–340.

Wang, H., Li, B., and Leng, C. (2009), "Shrinkage Tuning Parameter Selection With A Diverging Number of Parameters," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 71, 671–683.

Thank you for your attention!

Backup Slides

Equivalence to the Full-Data Estimator - Theorem 2.1 in main paper

Theorem (Equivalence to the Full-Data Estimator)

$Q_N^{\mathrm{CD}}(\boldsymbol{\beta}, \boldsymbol{\Theta}) - Q_N(\boldsymbol{\beta}, \boldsymbol{\Theta}) = constant.$

- $\begin{pmatrix} \widehat{\boldsymbol{\beta}}_{\mathrm{CD}}(\lambda) \\ \widehat{\boldsymbol{\Theta}}_{\mathrm{CD}}(\lambda) \end{pmatrix} = \begin{pmatrix} \widehat{\boldsymbol{\beta}}(\lambda) \\ \widehat{\boldsymbol{\Theta}}(\lambda) \end{pmatrix}$ is a straightforward consequence

Return to Theoretical Guarantees

Properties of Oracle Estimator - Theorem 4.1 in main paper

$$g_{\min} = \min_{1 \le s \le S} \sum_{i \in \text{subpop } s} n_i$$ denotes the minimum sub-sample size

Theorem (Properties of the Oracle Estimator)

Suppose regularity conditions hold. If $g_{\min} \gg N^{3/4}(p + Sq)^{1/2}$, the oracle estimator is consistent and possesses asymptotic normality. Recall that p and q are parameter dimensions of $\boldsymbol{\beta}$ and $\boldsymbol{\theta}_i$, respectively.

- The above nice properties hold if
  - $g_{\min}$ diverges fast enough $\Rightarrow$ S cannot grow too fast
  - For example, (S, p, q) must satisfy $S\sqrt{p + Sq} = o(N^{1/4})$
  - Moreover, $S = o(N^{1/6})$ if p and q are fixed

Return to Theoretical Guarantees

Oracle Property of the CD Fusion Estimator - Theorem 4.2 in main paper

Theorem (Oracle Property)

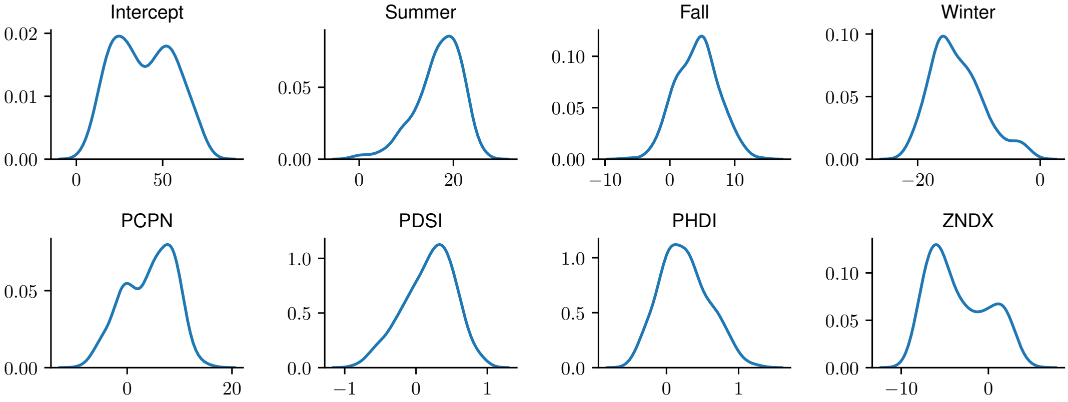
Suppose conditions in Theorem 2 and an additional minimal signal condition on $\min_{s \neq s'} \|\boldsymbol{\alpha}_s - \boldsymbol{\alpha}_{s'}\|$ *hold, then there exists a local minimizer* $\begin{pmatrix} \widehat{\boldsymbol{\beta}}(\lambda) \\ \widehat{\boldsymbol{\Theta}}(\lambda) \end{pmatrix}$ *of the objective function* $Q_N^{\mathrm{CD}}(\boldsymbol{\beta}, \boldsymbol{\Theta})$ *satisfying*

$$P\left(\begin{pmatrix} \widehat{\boldsymbol{\beta}}(\lambda) \\ \widehat{\boldsymbol{\Theta}}(\lambda) \end{pmatrix} = \begin{pmatrix} \widehat{\boldsymbol{\beta}}_{\mathrm{OR}} \\ \widehat{\boldsymbol{\Theta}}_{\mathrm{OR}} \end{pmatrix} \right) \to 1.$$

- $\begin{pmatrix} \widehat{\boldsymbol{\beta}}(\lambda) \\ \widehat{\boldsymbol{\alpha}}(\lambda) \end{pmatrix}$ possesses the same asymptotic distribution as $\begin{pmatrix} \widehat{\boldsymbol{\beta}}_{\mathrm{OR}} \\ \widehat{\boldsymbol{\Theta}}_{\mathrm{OR}} \end{pmatrix}$

Return to Theoretical Guarantees

- To use GLS, we need to determine heterogeneous effects through observing the kernel densities of the OLS estimates
- Intuitively, the distributions of heterogeneous effects are likely to form a multimodal or wide-spread shapes
- Kernel densities of the 344 OLS estimates obtained from the climate divisions



Go back to Real Data Example