Exponentiated Gradient Updates for Joint Sparsity Pattern Recovery from Multiple Measurement Vectors

Saurabh Khanna

Electrical Communication Engineering Dept.
Indian Institute of Science, Bangalore

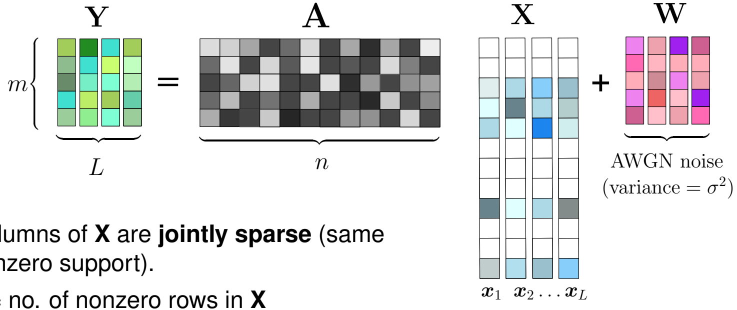
12th August, 2017

Outline

- Joint sparse support recovery problem
- Covariance matching framework for support recovery
- Matrix Exponentiated Gradient (MEG) Updates
- Two covariance matching algorithms based on MEG updates using
  - Log-Det Bregman divergence
  - Von-Neumann Bregman diverergence
- Numerical experiments
- Conclusions

Joint Sparse Support Recovery

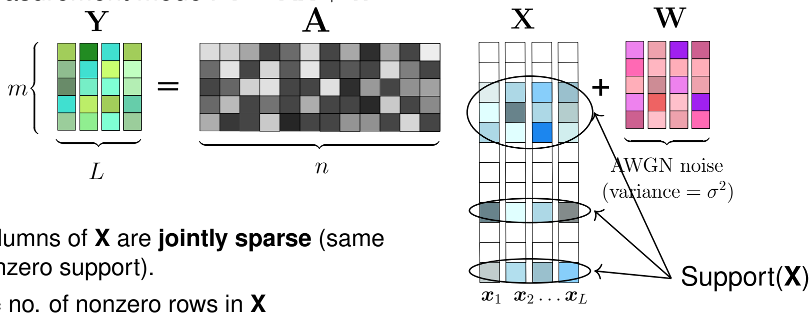
- Measurement model: $\mathbf{Y} = \mathbf{A}\mathbf{X} + \mathbf{W}$



- Columns of **X** are **jointly sparse** (same nonzero support).
- k = no. of nonzero rows in **X**
- Columns of **Y** are called MMVs
- No inter/intra vector correlations in **X**

Joint Sparse Support Recovery

- Measurement model: $\mathbf{Y} = \mathbf{AX} + \mathbf{W}$

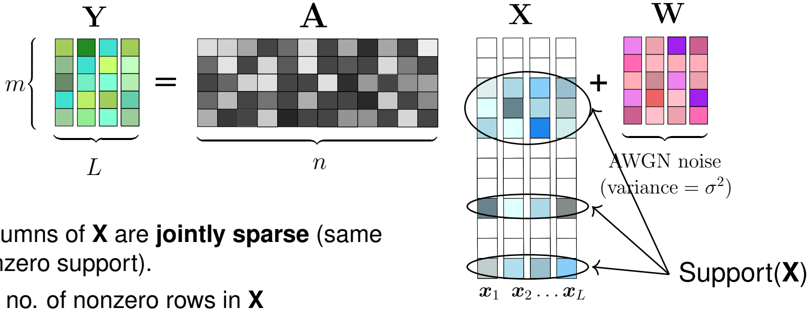


- Columns of **X** are **jointly sparse** (same nonzero support).
- *k* = no. of nonzero rows in **X**
- Columns of **Y** are called MMVs
- No inter/intra vector correlations in **X**

- Joint Sparse Support Recovery (JSSR) problem
  - Recover support(**X**) from $\left\{\mathbf{Y}, \mathbf{A}, \sigma^2\right\}$

Joint Sparse Support Recovery

- Measurement model: $\mathbf{Y} = \mathbf{AX} + \mathbf{W}$



- Columns of **X** are **jointly sparse** (same nonzero support).
- k = no. of nonzero rows in **X**
- Columns of **Y** are called MMVs
- No inter/intra vector correlations in **X**

- Joint Sparse Support Recovery (JSSR) problem
  - Recover support(**X**) from $\{\mathbf{Y}, \mathbf{A}, \sigma^2\}$

- Computational complexity of support recovery should scale reasonably with m, n, k and L

Covariance Matching Framework for Support Recovery

- MMV model: $\mathbf{Y} = \mathbf{A}\mathbf{X} + \mathbf{W}$

Covariance Matching Framework for Support Recovery

- MMV model: $\mathbf{Y} = \mathbf{AX} + \mathbf{W}$
  - $\mathbf{x}_j \sim \mathcal{N}(0, \mathrm{diag}(\boldsymbol{\gamma}))$

Covariance Matching Framework for Support Recovery

- MMV model: $\mathbf{Y} = \mathbf{AX} + \mathbf{W}$
  - $\mathbf{x}_j \sim \mathcal{N}(0, \mathrm{diag}(\boldsymbol{\gamma}))$
  - $\mathbf{y}_j \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_m + \mathbf{A}\Gamma\mathbf{A}^T)$

Covariance Matching Framework for Support Recovery

- MMV model: $\mathbf{Y} = \mathbf{AX} + \mathbf{W}$
  - $\mathbf{x}_j \sim \mathcal{N}(0, \mathrm{diag}(\gamma))$
  - $\mathbf{y}_j \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_m + \mathbf{A}\Gamma\mathbf{A}^T)$
- Covariance matrices:

Covariance Matching Framework for Support Recovery

- MMV model: $\mathbf{Y} = \mathbf{AX} + \mathbf{W}$
  - $\mathbf{x}_j \sim \mathcal{N}(0, \mathrm{diag}(\boldsymbol{\gamma}))$
  - $\mathbf{y}_j \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_m + \mathbf{A}\Gamma\mathbf{A}^T)$
- Covariance matrices:
  - Empirical $\mathbf{R}_{\mathbf{Y}} = \frac{1}{L}\mathbf{Y}\mathbf{Y}^T$

Covariance Matching Framework for Support Recovery

- MMV model: $\mathbf{Y} = \mathbf{AX} + \mathbf{W}$
  - $\mathbf{x}_j \sim \mathcal{N}(0, \mathrm{diag}(\gamma))$
  - $\mathbf{y}_j \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_m + \mathbf{A}\Gamma\mathbf{A}^T)$
- Covariance matrices:
  - Empirical $\mathbf{R}_\mathbf{Y} = \frac{1}{L}\mathbf{Y}\mathbf{Y}^T$
  - Parameterized $\boldsymbol{\Sigma}_\gamma = \sigma^2 \mathbf{I}_m + \mathbf{A}\Gamma\mathbf{A}^T$

Covariance Matching Framework for Support Recovery

- MMV model: $\mathbf{Y} = \mathbf{AX} + \mathbf{W}$
  - $\mathbf{x}_j \sim \mathcal{N}(0, \mathrm{diag}(\boldsymbol{\gamma}))$
  - $\mathbf{y}_j \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_m + \mathbf{A}\Gamma\mathbf{A}^T)$
- Covariance matrices:
  - Empirical $\mathbf{R}_\mathbf{Y} = \frac{1}{L}\mathbf{Y}\mathbf{Y}^T$
  - Parameterized $\boldsymbol{\Sigma}_\gamma = \sigma^2 \mathbf{I}_m + \mathbf{A}\Gamma\mathbf{A}^T$
- Covariance Matching Principle:

$$\hat{\gamma} = \underset{\boldsymbol{\gamma} \in \mathbb{R}_+^n}{\text{arg min}} \ \textbf{distance}\Big(\underbrace{\mathbf{R}_\mathbf{Y}}_{\substack{\text{empirical} \\ \text{MMV covariance}}} , \underbrace{\sigma^2 \mathbf{I} + \mathbf{A}\Gamma\mathbf{A}^T}_{\substack{\text{parameterized} \\ \text{MMV covariance}}} \Big)$$

Covariance Matching Framework for Support Recovery

- MMV model: $\mathbf{Y} = \mathbf{AX} + \mathbf{W}$
  - $\mathbf{x}_j \sim \mathcal{N}(0, \mathrm{diag}(\gamma))$
  - $\mathbf{y}_j \sim \mathcal{N}(0, \sigma^2\mathbf{I}_m + \mathbf{A}\Gamma\mathbf{A}^T)$
- Covariance matrices:
  - Empirical $\mathbf{R}_\mathbf{Y} = \frac{1}{L}\mathbf{Y}\mathbf{Y}^T$
  - Parameterized $\mathbf{\Sigma}_\gamma = \sigma^2\mathbf{I}_m + \mathbf{A}\Gamma\mathbf{A}^T$
- Covariance Matching Principle:

$$\hat{\gamma} = \underset{\gamma \in \mathbb{R}_+^n}{\text{arg min}} \ \textbf{distance}\Bigg(\underbrace{\mathbf{R}_\mathbf{Y}}_{\substack{\text{empirical} \\ \text{MMV covariance}}}, \underbrace{\sigma^2\mathbf{I} + \mathbf{A}\Gamma\mathbf{A}^T}_{\substack{\text{parameterized} \\ \text{MMV covariance}}} \Bigg)$$

- Distance = Log-Det Bregman divergence, we get MSBL

$$\hat{\gamma} = \underset{\gamma \in \mathbb{R}_+^n}{\text{arg min}} \ \mathcal{D}^{\text{Bregman}}_{-\log\det}\left(\mathbf{R}_\mathbf{Y}, \sigma^2\mathbf{I} + \mathbf{A}\Gamma\mathbf{A}^T\right)$$

Covariance Matching Framework for Support Recovery

- MMV model: $\mathbf{Y} = \mathbf{AX} + \mathbf{W}$
  - $\mathbf{x}_j \sim \mathcal{N}(0, \mathrm{diag}(\gamma))$
  - $\mathbf{y}_j \sim \mathcal{N}(0, \sigma^2\mathbf{I}_m + \mathbf{A}\Gamma\mathbf{A}^T)$
- Covariance matrices:
  - Empirical $\mathbf{R_Y} = \frac{1}{L}\mathbf{YY}^T$
  - Parameterized $\Sigma_\gamma = \sigma^2\mathbf{I}_m + \mathbf{A}\Gamma\mathbf{A}^T$
- Covariance Matching Principle:

$$\hat{\gamma} = \arg\min_{\gamma\in\mathbb{R}_+^n} \ \textbf{distance}\left(\underbrace{\mathbf{R_Y}}_{\substack{\text{empirical}\\ \text{MMV covariance}}} , \underbrace{\sigma^2\mathbf{I} + \mathbf{A}\Gamma\mathbf{A}^T}_{\substack{\text{parameterized}\\ \text{MMV covariance}}} \right)$$

- Distance = Log-Det Bregman divergence, we get MSBL

$$\hat{\gamma} = \arg\min_{\gamma\in\mathbb{R}_+^n} \ \mathcal{D}^{\text{Bregman}}_{-\log\det}\left(\mathbf{R_Y}, \sigma^2\mathbf{I} + \mathbf{A}\Gamma\mathbf{A}^T\right)$$

- Distance = Frobenius matrix norm, we get Co-LASSO

$$\hat{\gamma} = \arg\min_{\gamma\in\mathbb{R}_+^n} \ ||\gamma||_1 \quad \text{subj. to. } \mathbf{R_Y} = \sigma^2\mathbf{I} + \mathbf{A}\Gamma\mathbf{A}^T$$

Matrix Exponentiated Gradient (MEG) updates

- MEG updates were introduced by Kivinen and Warmuth in 1997.
  - Seminal paper: `Exponentiated gradient vs gradient descent for linear predictors`
- In most learning algorithms we need to learn a parameter vector from data
- Often, the parameter vector is structured
  - sparsity
  - non-negative
  - this work considers parameters to be a symmetric positive definite matrix
- Parameters are found my minimizing some kind of loss function $L(.)$
- Prior approach: project to feasible parameter set after every gradient descent update
- Goal is to design updates which preserve symmetry and positive definiteness

Matrix Exponentiated Gradient (MEG) updates

- Canonical problem:
  - Find a symmetric positive definite matrix that satisfies a number of linear inequality constraints

Matrix Exponentiated Gradient (MEG) updates

- Canonical problem:
  - Find a symmetric positive definite matrix that satisfies a number of linear inequality constraints
  - .... like covariance matching constraints!
- Matrix basics:

Matrix Exponentiated Gradient (MEG) updates

- Canonical problem:
  - Find a symmetric positive definite matrix that satisfies a number of linear inequality constraints
  - .... like covariance matching constraints!
- Matrix basics:
  - Let $\mathbf{A}$ admits eigenvalue decomposition $\mathbf{A} = \mathbf{U}\Lambda\mathbf{U}^T$

Matrix Exponentiated Gradient (MEG) updates

- Canonical problem:
  - Find a symmetric positive definite matrix that satisfies a number of linear inequality constraints
  - .... like covariance matching constraints!
- Matrix basics:
  - Let $\mathbf{A}$ admits eigenvalue decomposition $\mathbf{A} = \mathbf{U}\Lambda\mathbf{U}^T$
  - $\log(\mathbf{A}) = \mathbf{U}(\log(\Lambda))\mathbf{U}^T$

Matrix Exponentiated Gradient (MEG) updates

- Canonical problem:
  - Find a symmetric positive definite matrix that satisfies a number of linear inequality constraints
  - .... like covariance matching constraints!
- Matrix basics:
  - Let $\mathbf{A}$ admits eigenvalue decomposition $\mathbf{A} = \mathbf{U}\Lambda\mathbf{U}^T$
  - $\log(\mathbf{A}) = \mathbf{U}(\log(\Lambda))\mathbf{U}^T$
  - $\exp(\mathbf{A}) = \mathbf{U}(\exp(\Lambda))\mathbf{U}^T$

Bregman divergences

- Let F be a real-valued strictly convex differentiable function on a subset of matrices in $\mathbb{R}^{n \times n}$
- $f(\mathbf{W}) = \nabla F(\mathbf{W})$
- Bregman divergence between two matrix parameters $\bar{\mathbf{W}}$ and $\mathbf{W}$ is defined as

$$\mathcal{D}_F(\bar{\mathbf{W}}, \mathbf{W}) = F(\bar{\mathbf{W}}) - \underbrace{F(\mathbf{W}) - \mathrm{tr}\left(f(\mathbf{W})^T(\bar{\mathbf{W}} - \mathbf{W})\right)}_{\text{first order approx. of } F(\bar{\mathbf{W}}) \text{ around } \mathbf{W}}$$

- Due to strict convexity of F, we have $\mathcal{D}_F(\bar{\mathbf{W}}, \mathbf{W}) \geq 0$
- $F(\mathbf{W}) = -\log|\mathbf{W}|$ gives Log-Det Bregman matrix divergence

$$\mathcal{D}^{\text{Bregman}}_{-\log\det}(\bar{\mathbf{W}}, \mathbf{W}) = \log\frac{|\mathbf{W}|}{|\bar{\mathbf{W}}|} + \mathrm{tr}\left(\mathbf{W}^{-1}\bar{\mathbf{W}}\right) - n$$

- $F(\mathbf{W}) = \mathrm{tr}\left(\mathbf{W}\log\mathbf{W} - \mathbf{W}\right)$ gives Von-Neumann matrix divergence

$$\mathcal{D}^{\text{Bregman}}_{\text{von-Neumann}}(\bar{\mathbf{W}}, \mathbf{W}) = \mathrm{tr}\left(\bar{\mathbf{W}}\log\bar{\mathbf{W}} - \bar{\mathbf{W}}\log\mathbf{W} - \bar{\mathbf{W}} + \mathbf{W}\right)$$

MEG updates

- Let $L_t(\mathbf{W})$ be a (time-varying) convex loss function
- Say, we aim to solve the following problem:

$$\mathbf{W}_{t+1} = \arg\min_{\mathbf{W}} \mathcal{D}_F(\mathbf{W}, \mathbf{W}_t) + \eta L_t(\mathbf{W})$$

  - want to stay close to old parameter $\mathbf{W}_t$
  - at the same time, achieve a small loss
    - ★ Learning rate η implements tradeoff between these two conflicting goals

- Due to convexity of the objective, $\mathbf{W}_{t+1}$ can be found via zero gradient optimality condition as

$$\mathbf{W}_{t+1} = f^{-1}\left(f(\mathbf{W}_t) - \eta \nabla_{\mathbf{W}} L_t(\mathbf{W}_{t+1})\right)$$

  - Unfortunately $\mathbf{W}_{t+1}$ not available in closed form
  - An approximation suggested by Kivinen and Warmuth fixes this issue!

$$\nabla_{\mathbf{W}} L_t(\mathbf{W}_{t+1}) \approx \nabla_{\mathbf{W}} L_t(\mathbf{W}_t)$$

- Final form of the MEG update:

$$\mathbf{W}_{t+1} = f^{-1}\left(f(\mathbf{W}_t) - \eta \nabla_{\mathbf{W}} L_t(\mathbf{W}_t)\right)$$

Two types of MEG updates

- Log-det divergence based MEG updates:
  - $F(\mathbf{W}) = -\log\det\mathbf{W}$
  - $f(\mathbf{W}) = -\mathbf{W}^{-1}$ and $f^{-1}(\mathbf{Q}) = \mathbf{Q}$

$$\mathbf{W}_{t+1} = -\left(-(\mathbf{W}_t)^{-1} - \eta\nabla_{\mathbf{W}} L_t(\mathbf{W}_t)\right)^{-1}$$

- Von-Neumann divergence based MEG updates:
  - $F(\mathbf{W}) = \mathbf{W}\log\mathbf{W} - \mathbf{W}$
  - $f(\mathbf{W}) = \log\mathbf{W}$ and $f^{-1}(\mathbf{Q}) = \exp\mathbf{Q}$

$$\mathbf{W}_{t+1} = \exp\left(\log\mathbf{W}_t - \eta\left(\nabla_{\mathbf{W}} L_t(\mathbf{W}_t)\right)\right)$$

Covariance matching MEG updates for support recov

- Find a sparse, nonnegative Γ which satisfies $\mathbf{R_Y} = \sigma^2 \mathbf{I}_m + \mathbf{A}\Gamma\mathbf{A}^T$
- Parameter space: set of all positive definite diagonal matrices
- Our loss function $L(\Gamma)$: $\left|\left|\left|\mathbf{R_Y} - (\sigma^2\mathbf{I} + \mathbf{A}\Gamma\mathbf{A}^T)\right|\right|\right|_F^2$
- $\nabla_\Gamma L(\Gamma)(i,i) = 2\mathbf{a}_i^T \left(\mathbf{A}\Gamma\mathbf{A}^T - (\mathbf{R_Y} - \sigma^2\mathbf{I})\right)\mathbf{a}_i$
- Log-Det divergence based MEG update:

$$\gamma_{t+1}(i) = \left(\frac{1}{\frac{1}{\gamma_t(i)} + 2\eta\mathbf{a}_i^T\left(\mathbf{A}\Gamma\mathbf{A}^T - (\mathbf{R_Y} - \sigma^2\mathbf{I})\right)\mathbf{a}_i}\right), \quad 1 \le i \le n$$

- Von-Neumann divergence based MEG update:

$$\gamma_{t+1}(i) = \gamma_t(i) \cdot e^{-2\eta\mathbf{a}_i^T\left(\mathbf{A}\Gamma\mathbf{A}^T - (\mathbf{R_Y} - \sigma^2\mathbf{I})\right)\mathbf{a}_i}, \quad 1 \le i \le n$$

Numerical experiments

Thank You.....Questions?