Alternating Direction Method of Multipliers

a talk by

Vinícius and Prof. Daniel Palomar
The Hong Kong University of Science and Technology

ELEC5470/IEDA6100A - Convex Optimization

December, 2020

Contents

Optimization Algorithms

- Gradient Descent
- Newton
- Interior Point Methods (IPM)
- Block Coordinate Descent (BCD)
- Majorization-Minimization (MM)
- Block Majorization-Minimization (BMM)
- Successive Convex Approximation (SCA)

Optimization Algorithms

- Gradient Descent
- Newton
- Interior Point Methods (IPM)
- Block Coordinate Descent (BCD)
- Majorization-Minimization (MM)
- Block Majorization-Minimization (BMM)
- Successive Convex Approximation (SCA)
- ...

Optimization Algorithms

- Gradient Descent
- Newton
- Interior Point Methods (IPM)
- Block Coordinate Descent (BCD)
- Majorization-Minimization (MM)
- Block Majorization-Minimization (BMM)
- Successive Convex Approximation (SCA)
- ...
- **Alternating Direction Method of Multipliers (ADMM)**

Reference

- Boyd *et al.* **Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers**. *Foundations and Trends in Machine Learning*. 2010.
- Available online for **free**: `https://web.stanford.edu/~boyd/papers/pdf/admm_distr_stats.pdf`
- Citations: 13519[1]

[1] as of Nov. 24th 2020

Optimization Problem

$$
\begin{aligned}
\underset{\boldsymbol{x},\,\boldsymbol{z}}{\text{minimize}} \quad & f(\boldsymbol{x}) + g(\boldsymbol{z}) \\
\text{subject to} \quad & \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{z} = \boldsymbol{c}
\end{aligned} \tag{1}
$$

- variables: $\boldsymbol{x} \in \mathbb{R}^n$ and $\boldsymbol{z} \in \mathbb{R}^m$
- parameters: $\boldsymbol{A} \in \mathbb{R}^{p\times n}$, $\boldsymbol{B} \in \mathbb{R}^{p\times m}$, and $\boldsymbol{c} \in \mathbb{R}^p$

Algorithm

- Augmented Lagrangian:

$$L_\rho(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{y}) = \underbrace{f(\boldsymbol{x}) + g(\boldsymbol{z}) + \langle \boldsymbol{y}, \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{z} - \boldsymbol{c} \rangle}_{\text{Lagrangian}} + \frac{\rho}{2} \|\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{z} - \boldsymbol{c}\|_{\mathrm{F}}^2 \tag{2}$$

- ADMM consists of the iterations:

$$\boldsymbol{x}^{k+1} := \operatorname*{arg\,min}_{\boldsymbol{x}} \; L_\rho(\boldsymbol{x}^k, \boldsymbol{z}^k, \boldsymbol{y}^k) \tag{3}$$

$$\boldsymbol{z}^{k+1} := \operatorname*{arg\,min}_{\boldsymbol{z}} \; L_\rho(\boldsymbol{x}^{k+1}, \boldsymbol{z}^k, \boldsymbol{y}^k) \tag{4}$$

$$\boldsymbol{y}^{k+1} := \boldsymbol{y}^k + \rho\left(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{z} - \boldsymbol{c}\right) \tag{5}$$

Are sparse solutions recoverable via ℓ_1-norm?

- theoretically:

Theorem

Let $\hat{\mathbf{\Theta}} \in \mathbb{R}^{p\times p}$ be the global minimum of (1) with $p > 3$. Define $s_1 = \max_k S_{kk}$ and $s_2 = \min_{ij} S_{ij}$. If the regularization parameter λ in (1) satisfies $\lambda \in [(2+2\sqrt{2})(p+1)(s_1 - s_2), +\infty)$, then the estimated graph weight $\hat{W}_{ij} = -\hat{\Theta}_{ij}$ obeys

$$\hat{W}_{ij} \geq \frac{1}{(s_1 - (p+1)s_2 + \lambda)p} > 0, \quad \forall\, i \neq j.$$

Proof

Please refer to our supplementary material 😄

Our framework for sparse graphs

- nonconvex formulation:

$$\underset{\boldsymbol{w}\geq\boldsymbol{0}}{\mathsf{minimize}} \quad \mathrm{tr}(\boldsymbol{S}\mathcal{L}\boldsymbol{w}) - \log\det(\mathcal{L}\boldsymbol{w} + \boldsymbol{J}) + \sum_i h_\lambda(w_i) \tag{6}$$

  $\mathcal{L}$ is the Laplacian operator and $h_\lambda(\cdot)$ is a nonconvex regularizer such as

  - *Minimax Concave Penalty (MCP)*
  - *Smoothly Clipped Absolute Deviation (SCAD)*

Our framework for sparse graphs

Algorithm 1: Connected sparse graph learning

Data: Sample covariance $\boldsymbol{S}$, $\lambda > 0$, $\hat{\boldsymbol{w}}^{(0)}$

Result: Laplacian estimation: $\mathcal{L}\hat{\boldsymbol{w}}^{(k)}$

```
k ← 1
while stopping criteria not met do
    ▷ update z_i^(k−1) = h'_λ(ŵ_i^(k−1)), for i = 1, ..., p(p − 1)/2
    ▷ update ŵ^(k) = arg min_{w≥0} − log det(ℒw + J) + tr(SℒW) + Σ_i z_i^(k−1) w_i
    ▷ k ← k + 1
end
```

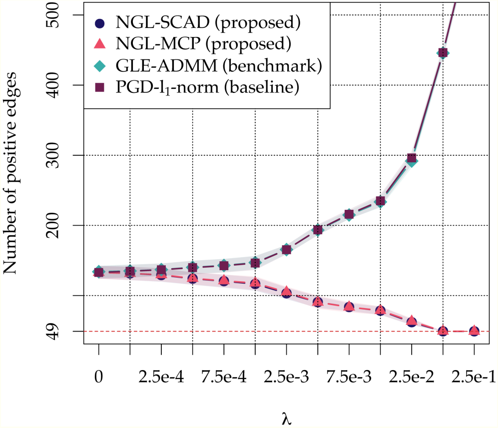
$k \leftarrow 1$

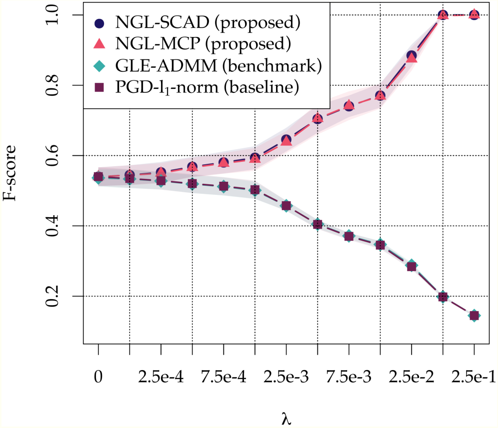
while *stopping criteria not met* **do**

- ▷ update $z_i^{(k-1)} = h'_\lambda(\hat{w}_i^{(k-1)})$, for $i = 1, \dots, p(p-1)/2$
- ▷ update $\hat{\boldsymbol{w}}^{(k)} = \arg\min_{\boldsymbol{w} \geq \boldsymbol{0}} - \log\det(\mathcal{L}\boldsymbol{w} + \boldsymbol{J}) + \mathrm{tr}(\boldsymbol{S}\mathcal{L}\boldsymbol{w}) + \sum_i z_i^{(k-1)} w_i$
- ▷ $k \leftarrow k + 1$

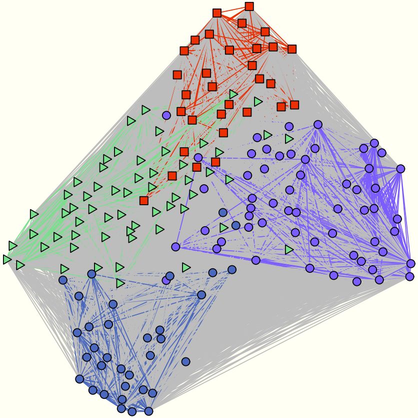
end

Sneak peek on the results: synthetic data

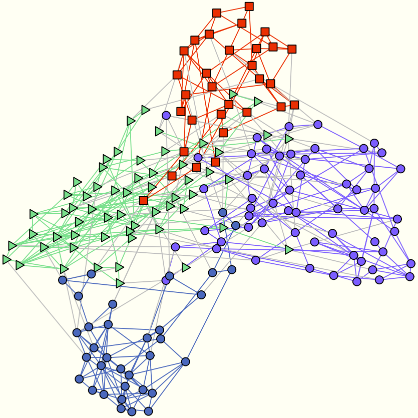




Sneak peek on the results: S&P 500 stocks



(a) GLE-ADMM (benchmark) $\lambda = 0$



(b) NGL-MCP (proposed) $\lambda = 0.5$

Reproducibility

- The code for the experiments can be found at
  `https://github.com/mirca/sparseGraph`

- Convex Research Group at HKUST:
  `https://www.danielppalomar.com`