Exercises
Computational Intelligence Lab
SS 2017

Machine Learning Institute
Dept. of Computer Science, ETH Zürich
Prof. Dr. Thomas Hofmann
Web http://cil.inf.ethz.ch/

Series 12, June 2-3, 2017 (Robust Principal Component Analysis)

Problem 1 (ADMM for LASSO):

The *LASSO (least absolute shrinkage and selection operator)* is an L_1-regularized version of the linear least squares problem. It can be formulated as the following optimization problem:

$$\underset{\mathbf{x}}{\text{minimize}} \quad \frac{1}{2}\|A\mathbf{x}-\mathbf{b}\|_2^2 + \lambda\|\mathbf{x}\|_1$$

This optimization problem can also be written in ADMM form, where we explicitly split the regularization and the loss function:

$$\begin{array}{ll} \underset{\mathbf{x}_1,\mathbf{x}_2}{\text{minimize}} & \frac{1}{2}\|A\mathbf{x}_1-\mathbf{b}\|_2^2 + \lambda\|\mathbf{x}_2\|_1 \\ \text{subject to} & \mathbf{x}_1 - \mathbf{x}_2 = 0 \end{array}$$

We write down the augmented Lagrangian L_ρ for this problem:

$$L_\rho(\mathbf{x}_1,\mathbf{x}_2,\boldsymbol{\nu}) = \frac{1}{2}\|A\mathbf{x}_1-\mathbf{b}\|_2^2 + \lambda\|\mathbf{x}_2\|_1 + \boldsymbol{\nu}^T(\mathbf{x}_1-\mathbf{x}_2) + \frac{\rho}{2}\|\mathbf{x}_1-\mathbf{x}_2\|_2^2$$

Minimizing this functional with respect to $\mathbf{x}_1$ and $\mathbf{x}_2$ leads to the following update steps:

$$\begin{aligned} \underset{\mathbf{x}_1}{\operatorname{argmin}}\, L_\rho(\mathbf{x}_1,\mathbf{x}_2,\boldsymbol{\nu}) &= (A^TA+\rho I)^{-1}(A^T\mathbf{b}+\rho\mathbf{x}_2-\boldsymbol{\nu}) \\ \underset{\mathbf{x}_2}{\operatorname{argmin}}\, L_\rho(\mathbf{x}_1,\mathbf{x}_2,\boldsymbol{\nu}) &= S_{\lambda/\rho}(\mathbf{x}_1+\boldsymbol{\nu}/\rho) \end{aligned}$$

where $\mathcal{S}_\tau(x) := \operatorname{sign}(x)\max(|x|-\tau, 0)$ is the Shrinkage operator for a threshold value τ.

Adding the dual update step, we get an ADMM algorithm for LASSO:

$\mathbf{x}_1^{(0)} := \mathbf{x}_2^{(0)} := \boldsymbol{\nu}^{(0)} := 0,$ parameter $\rho > 0$
while not converged **do**
  $\mathbf{x}_1^{(t+1)} := (A^TA+\rho I)^{-1}(A^T\mathbf{b}+\rho\mathbf{x}_2^{(t)}-\boldsymbol{\nu}^{(t)})$
  $\mathbf{x}_2^{(t+1)} := S_{\lambda/\rho}(\mathbf{x}_1^{(t+1)}+\boldsymbol{\nu}^{(t)}/\rho)$
  $\boldsymbol{\nu}^{(t+1)} := \boldsymbol{\nu}^{(t)} + \rho\big(\mathbf{x}_1^{(t+1)} - \mathbf{x}_2^{(t+1)}\big)$
end while

1. Implement the ADMM algorithm for LASSO. You can use the following code to generate sample data:

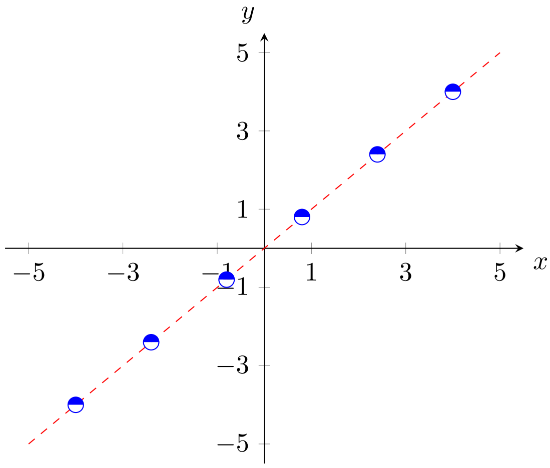
```
import numpy as np
M, N = 150, 500
p = 10 / N
x0 = np.random.randn(N)
x0[np.random.rand(N) > p] = 0
A = np.random.randn(M, N)
A /= np.sqrt(np.sum(A**2, axis=0))   # normalize columns
b = np.dot(A, x0) + np.sqrt(0.001) * np.random.randn(M)
```

2. Perform some experiments with the method: Choose different λ and ρ. What can you say about the performance?

3. Use $\lambda = \max(A^T\mathbf{b})$, $\rho = 1$. For this values of the parameters, generate plots showing the following values depending on the iteration number: $L_\rho(\mathbf{x}_1, \mathbf{x}_2, \boldsymbol{\nu})$; $\frac{1}{2}\|A\mathbf{x}_1 - \mathbf{b}\|_2^2 + \lambda\|\mathbf{x}_2\|_1$; $\|\mathbf{x}_1 - \mathbf{x}_2\|$. Give interpretations of the plots.

Problem 2 (RPCA Theory):

We are given the following set of points shown in the figure below. The red line is the principal component of this set, denoted u_1.



- Suggest a corrupted version of these points such that PCA can still estimate the principal component u_1.
- Suggest a corrupted version of these points such that Robust PCA estimates the principal component more reliably than PCA.

Problem 3 (RPCA Application):

We use robust PCA to extract the foreground and background of a video. The left-most image shows one frame extracted from the video with its corresponding background and foreground in the middle and right image respectively.

We see that each person in the video is considered as foreground, except the one in the rectangle. Can you explain why this is the case?