LAGRANGIAN METHODS
WHY LAGRANGIAN METHODS?

Smooth functions
\[
\text{minimize } f(x)
\]

Non-differentiable
\[
\text{minimize } f(x)
\]

Constrained problems?
\[
\text{minimize } f(x) \\
\text{subject to } g(x) \leq 0 \\
\text{ } h(x) = 0
\]

Gradient descent
Newton’s method
Quasi-newton
Conjugate gradients etc…

Proximal methods

Lagrangian methods
LAGRANGIAN

Simple case

minimize \( f(x) \)
subject to \( Ax + b = 0 \)

“Saddle-point” form

\[
\min_x \max_\lambda f(x) + \langle \lambda, Ax + b \rangle
\]

Lagrangian
LAGRANGIAN APPROACH

minimize $f(x)$ subject to $Ax + b = 0$

Dual approach: maximize dual function

$$d(\lambda) = \min_x f(x) + \langle \lambda, Ax + b \rangle$$

using conjugate

$$d(\lambda) = \langle b, \lambda \rangle - f^*(-A^T \lambda)$$

Lagrangian approach: directly find saddle point of Lagrangian

$$\max_{\lambda} \min_x f(x) + \langle \lambda, Ax + b \rangle$$
UZAWA'S METHOD

\[
\begin{align*}
\text{minimize} & \quad f(x) \quad \text{subject to} \quad Ax + b = 0 \\
\max_{\lambda} \min_{x} & \quad f(x) + \langle \lambda, Ax + b \rangle
\end{align*}
\]

\[
\begin{align*}
\text{minimize} & \quad x^{k+1} = \arg \min_{x} f(x) + \langle \lambda^{k}, Ax + b \rangle \\
\text{gradient ascent} & \quad \lambda^{k+1} = \lambda^{k} + \tau(Ax^{k+1} + b)
\end{align*}
\]
UZAWA’S METHOD

minimize \( f(x) \) subject to \( Ax + b = 0 \)

Uzawa’s method

\[
\begin{align*}
\text{minimize} \quad & x^{k+1} = \arg \min_x f(x) + \langle \lambda^k, Ax + b \rangle \\
\text{gradient ascent} \quad & \lambda^{k+1} = \lambda^k + \tau(Ax^{k+1} + b)
\end{align*}
\]

dual

\[
d(\lambda) = \langle b, \lambda \rangle - f^*(-A^T \lambda)
\]

optimality

\[
\begin{align*}
0 & \in \partial f(x^{k+1}) + A^T \lambda^k \\
-A^T \lambda^k & \in \partial f(x^{k+1}) \\
x^{k+1} & \in \partial f^*(-A^T \lambda^k) \\
A x^{k+1} + b & \in A \partial f^*(-A^T \lambda^k) + b
\end{align*}
\]
GRADIENT ASCENT

minimize \( f(x) \) subject to \( Ax + b = 0 \)

Uzawa's method

minimize \( x^{k+1} = \arg \min_x f(x) + \langle \lambda^k, Ax + b \rangle \)

gradient ascent \( \lambda^{k+1} = \lambda^k + \tau (Ax^{k+1} + b) \)

dual \( d(\lambda) = \langle b, \lambda \rangle - f^*(-A^T \lambda) \)

\( Ax^{k+1} + b \in A \partial f^*(-A^T \lambda^k) + b \)

dual gradient

\( \lambda^{k+1} = \lambda^k + \tau \partial d(\lambda^k) \)

stepsize restriction?
CONVERGENCE

\[ Ax^{k+1} + b \in A\partial f^*(-A^T \lambda^k) + b \]

**dual gradient**

**Nifty Theorem: Strong Convexity = Smooth Dual**

If \( f \) is strongly convex with constant \( m \), then \( f^* \) has Lipschitz continuous gradient with constant \( L = 1/m \).

gradient ascent \[ \lambda^{k+1} = \lambda^k + \tau(Ax + b) \]

\[ L_{dual} = \|A^T A\| L_{f^*} = \frac{\|A^T A\|_{op}}{m} \]

\[ \tau \leq 2/L_{dual} = \frac{2m}{\|A^T A\|_{op}} \]

Problem: requires **strong** convexity!
AUGMENTED LAGRANGIAN

idea: add curvature to the primal problem

minimize $f(x)$
subject to $Ax + b = 0$

Lagrangian

$L(x, \lambda) = f(x) + \langle \lambda, Ax + b \rangle$

Augmented Lagrangian

$L_\tau(x, \lambda) = f(x) + \langle \lambda, Ax + b \rangle + \frac{\tau}{2} \|Ax + b\|^2$

• optimality for $\lambda$: $Ax + b = 0$
• reduced energy: $f(x)$
• saddle point = solution to constrained problem
METHOD OF MULTIPLIERS

minimize $f(x)$
subject to $Ax + b = 0$

Augmented Lagrangian

$$\min_x \max_{\lambda} f(x) + \langle \lambda, Ax + b \rangle + \frac{\tau}{2} \| Ax + b \|^2$$

Method of Multipliers

minimize

$$x^{k+1} = \arg \min_x f(x) + \langle \lambda^k, Ax + b \rangle + \frac{\tau}{2} \| Ax + b \|^2$$

gradient step

$$\lambda^{k+1} = \lambda^k + \tau (Ax^{k+1} + b)$$
WHY IS THIS BETTER?

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad Ax + b = 0
\end{align*}
\]

**Method of Multipliers**

\[
\begin{align*}
\text{minimize} & \quad x^{k+1} = \arg \min_x f(x) + \langle \lambda^k, Ax + b \rangle + \frac{\tau}{2} \| Ax + b \|^2 \\
\text{gradient step} & \quad \lambda^{k+1} = \lambda^k + \tau (Ax^{k+1} + b)
\end{align*}
\]

dual

\[
\begin{align*}
d(\lambda) = \langle \lambda, b \rangle - f^*(-A\lambda) \\
0 \in \partial f(x^{k+1}) + A^T \lambda^k + \tau A^T (Ax^{k+1} + b) \\
0 \in \partial f(x^{k+1}) + A^T (\lambda^k + \tau (Ax^{k+1} + b)) \\
0 \in \partial f(x^{k+1}) + A^T \lambda^{k+1}
\end{align*}
\]
WHY IS THIS BETTER?

minimize \( f(x) \)
subject to \( Ax + b = 0 \)

\[
d(\lambda) = \langle \lambda, b \rangle - f^*(-A^T \lambda)
\]

iterates satisfy…

\[
0 \in \partial f(x^{k+1}) + A^T \lambda^{k+1}
\]

\[
-A^T \lambda^{k+1} \in \partial f(x^{k+1})
\]

\[
x^{k+1} \in \partial f^*(-A^T \lambda^{k+1})
\]

\[
Ax^{k+1} + b \in A\partial f^*(-A^T \lambda^{k+1}) + b
\]

dual gradient
MM=BACKWARD GRADIENT

minimize \( f(x) \)
subject to \( Ax + b = 0 \)

dual
\[
d(\lambda) = \langle \lambda, b \rangle - f^*(-A^T\lambda)
\]

method of multipliers
minimize \( x^{k+1} = \text{arg min}_x f(x) + \langle \lambda^k, Ax + b \rangle + \frac{\tau}{2} \|Ax + b\|^2 \)
gradient step \( \lambda^{k+1} = \lambda^k + \tau (Ax^{k+1} + b) \)

\[
Ax^{k+1} + b \in A\partial f^*(-A^T\lambda^{k+1}) + b
\]

backward gradient ascent
\[
\lambda^{k+1} = \lambda^k + \tau \partial d(\lambda^{k+1})
\]
CONVERGENCE

minimize $f(x)$
subject to $Ax + b = 0$

backward gradient ascent

$\lambda^{k+1} = \lambda^k + \tau \partial d(\lambda^{k+1})$

Works for any stepsize!

problem: requires solution of problem on every iteration

$x^{k+1} = \arg \min_x f(x) + \langle \lambda^k, Ax + b \rangle + \frac{\tau}{2} \|Ax + b\|^2$

Can we solve the whole problem in one shot?
SPLIT OBJECTIVE

just like we did for FBS…

\[
\begin{align*}
\text{minimize} \quad & f(x) + g(y) \\
\text{subject to} \quad & Ax + By + c = 0
\end{align*}
\]

\[
L_\tau(x, y, \lambda) = f(x) + g(y) + \langle \lambda, Ax + By + c \rangle + \frac{\tau}{2} \|Ax + By + c\|^2
\]

method of multipliers

\[
x^{k+1}, y^{k+1} = \arg\min_{x, y} f(x) + g(y) + \langle \lambda^k, Ax + By + c \rangle + \frac{\tau}{2} \|Ax + By + c\|^2
\]

\[
\lambda^{k+1} = \lambda^k + \tau (Ax^{k+1} + By^{k+1} + c)
\]
ADMM

minimize \( f(x) + g(y) \)

subject to \( Ax + By + c = 0 \)

method of multipliers

\[
x^{k+1}, y^{k+1} = \arg \min_{x,y} f(x) + g(y) + \langle \lambda^k, Ax + By + c \rangle + \frac{\tau}{2} \|Ax + By + c\|^2
\]

\[
\lambda^{k+1} = \lambda^k + \tau (Ax^{k+1} + By^{k+1} + c)
\]

alternating direction method of multipliers

\[
x^{k+1} = \arg \min_x f(x) + \langle \lambda^k, Ax \rangle + \frac{\tau}{2} \|Ax + By^k + c\|^2
\]

\[
y^{k+1} = \arg \min_y g(y) + \langle \lambda^k, By \rangle + \frac{\tau}{2} \|Ax^{k+1} + By + c\|^2
\]

\[
\lambda^{k+1} = \lambda^k + \tau (Ax^{k+1} + By^{k+1} + c)
\]
RESIDUALS

Lagrangian

\[ L(x, y, \lambda) = f(x) + g(y) + \langle \lambda, Ax + By + c \rangle \]

How do we measure closeness to a saddle point?

optimality conditions

**x-residual:** \[ 0 \in \partial f(x) + A^T \lambda \]

**y-residual:** \[ 0 \in \partial g(x) + B^T \lambda \]

**\lambda-residual:** \[ 0 = Ax + By + c \]
RESIDUALS

Lagrangian

\[ L(x, y, \lambda) = f(x) + g(y) + \langle \lambda, Ax + By + c \rangle \]

alternating direction method of multipliers

\[ x^{k+1} = \arg \min_x f(x) + \langle \lambda^k, Ax \rangle + \frac{\tau}{2} \| Ax + By^k + c \|^2 \]

\[ y^{k+1} = \arg \min_y g(y) + \langle \lambda^k, By \rangle + \frac{\tau}{2} \| Ax^{k+1} + By + c \|^2 \]

\[ \lambda^{k+1} = \lambda^k + \tau(Ax^{k+1} + By^{k+1} + c) \]

\[ 0 \in \partial g(y^{k+1}) + B^T \lambda^k + \tau B^T (Ax^{k+1} + By^{k+1} + c) \]

\[ 0 \in \partial g(y^{k+1}) + B^T (\lambda^k + \tau(Ax^{k+1} + By^{k+1} + c)) \]

**y-residual:** \[ 0 \in \partial g(y^{k+1}) + B^T \lambda^{k+1} \]
RESIDUALS

Lagrangian

\[ L(x, y, \lambda) = f(x) + g(y) + \langle \lambda, Ax + By + c \rangle \]

 alternating direction method of multipliers

\[ x^{k+1} = \arg \min_x f(x) + \langle \lambda^k, Ax \rangle + \frac{\tau}{2} \|Ax + By^k + c\|^2 \]
\[ y^{k+1} = \arg \min_y g(y) + \langle \lambda^k, By \rangle + \frac{\tau}{2} \|Ax^{k+1} + By + c\|^2 \]
\[ \lambda^{k+1} = \lambda^k + \tau (Ax^{k+1} + By^{k+1} + c) \]

0 \in \partial f(x^{k+1}) + A^T (\lambda^k + \tau (Ax^{k+1} + By^k + c))

0 \in \partial f(x^{k+1}) + A^T (\lambda^k + \tau (Ax^{k+1} + By^{k+1} + c)) + \tau A^T By^k - \tau A^T By^{k+1}

0 \in \partial f(x^{k+1}) + A^T \lambda^{k+1} + \tau A^T By^k - \tau A^T By^{k+1}

x-residual: \[ \tau A^T By^{k+1} - \tau A^T By^k \in \partial f(x^{k+1}) + A^T \lambda^{k+1} \]
RESIDUALS

Lagrangian

\[ L(x, y, \lambda) = f(x) + g(y) + \langle \lambda, Ax + By + c \rangle \]

alternating direction method of multipliers

\[ x^{k+1} = \arg \min_x f(x) + \langle \lambda^k, Ax \rangle + \frac{\tau}{2} \| Ax + By^k + c \|^2 \]

\[ y^{k+1} = \arg \min_y g(y) + \langle \lambda^k, By \rangle + \frac{\tau}{2} \| Ax^{k+1} + By + c \|^2 \]

\[ \lambda^{k+1} = \lambda^k + \tau (Ax^{k+1} + By^{k+1} + c) \]

lambda-residual: \[ Ax^{k+1} + By^{k+1} + c \]
CONVERGENCE

Lagrangian

\[ L(x, y, \lambda) = f(x) + g(y) + \langle \lambda, Ax + By + c \rangle \]

x (primal) residual: \[ \tau A^T By^{k+1} - \tau A^T By^k \]

lambda (dual) residual: \[ Ax^{k+1} + By^{k+1} + c \]

**Theorem** (He and Yuan ‘12)

For any fixed stepsize \( \tau \), ADMM converges in the residuals with rate

\[ \|\tau A^T By^{k+1} - \tau A^T By^k\|^2 + \|Ax^{k+1} + By^{k+1} + c\|^2 < O \left( \frac{1}{k} \right) \]
LASSO

\[
\text{minimize } \mu |x| + \frac{1}{2} \|Ax - b\|^2
\]

"split Bregman" form

\[
\text{minimize } \mu |y| + \frac{1}{2} \|Ax - b\|^2
\]

subject to \( x - y = 0 \)

augmented Lagrangian

\[
\mu |y| + \frac{1}{2} \|Ax - b\|^2 + \langle \lambda, x - y \rangle + \frac{\tau}{2} \|x - y\|^2
\]
**LASSO**

minimize \( \mu |x| + \frac{1}{2} \|Ax - b\|^2 \)

augmented Lagrangian

\[
\mu |y| + \frac{1}{2} \|Ax - b\|^2 + \langle \lambda, x - y \rangle + \frac{\tau}{2} \|x - y\|^2
\]

**ADMM lasso**

\[
x^{k+1} = \arg \min_x \frac{1}{2} \|Ax - b\|^2 + \langle \lambda^k, x \rangle + \frac{\tau}{2} \|x - y^k\|^2
\]

\[
y^{k+1} = \arg \min_y \mu |y| - \langle \lambda^k, y \rangle + \frac{\tau}{2} \|x^{k+1} - y\|^2
\]

\[
\lambda^{k+1} = \lambda^k + \tau(x^{k+1} - y^{k+1})
\]

how do you solve these sub-problems?
EXAMPLE: SPLIT BREGMAN

\[
\text{minimize} \quad \mu |\nabla x| + \frac{1}{2} \|Ax - f\|^2
\]

“split Bregman” form

\[
\text{minimize} \quad \mu |y| + \frac{1}{2} \|Ax - f\|^2
\]

subject to \( \nabla x - y = 0 \)

augmented Lagrangian

\[
\mu |y| + \frac{1}{2} \|Ax - f\|^2 + \langle \lambda, \nabla x - y \rangle + \frac{\tau}{2} \|\nabla x - y\|^2
\]
EXAMPLE: SPLIT BREGMAN

\[ \text{minimize} \quad \mu |\nabla x| + \frac{1}{2} \|Ax - f\|^2 \]

\[ \mu |y| + \frac{1}{2} \|Ax - f\|^2 + \langle \lambda, \nabla x - y \rangle + \frac{\tau}{2} \|\nabla x - y\|^2 \]

**Split Bregman TV**

\[ x^{k+1} = \arg \min_x \frac{1}{2} \|Ax - f\|^2 + \langle \lambda^k, \nabla x \rangle + \frac{\tau}{2} \|\nabla x - y^k\|^2 \]

\[ y^{k+1} = \arg \min_y \mu |y| - \langle \lambda^k, y \rangle + \frac{\tau}{2} \|\nabla x^{k+1} - y\|^2 \]

\[ \lambda^{k+1} = \lambda^k + \tau (\nabla x^{k+1} - y^{k+1}) \]
**X-UPDATE**

\[
\text{minimize} \quad \mu |\nabla x| + \frac{1}{2} \|Ax - f\|^2
\]

\[
\mu |y| + \frac{1}{2} \|Ax - f\|^2 + \langle \lambda, \nabla x - y \rangle + \frac{\tau}{2} \|\nabla x - y\|^2
\]

**x-update**

\[
x^{k+1} = \arg \min_x \frac{1}{2} \|Ax - f\|^2 + \langle \lambda^k, \nabla x \rangle + \frac{\tau}{2} \|\nabla x - y^k\|^2
\]

**optimality condition**

\[
A^T (Ax - f) + \nabla^T \lambda^k + \tau \nabla^T (\nabla x - y^k) = 0
\]

**linear system**

\[
(A^T A + \tau \nabla^T \nabla)x = A^T f - \nabla^T \lambda^k + \tau \nabla^T y^k
\]
minimize $\mu |\nabla x| + \frac{1}{2} \|Ax - f\|^2$

linear system

$$(A^T A + \tau \nabla^T \nabla)x = A^T f - \nabla^T \lambda^k + \tau \nabla^T y^k$$

deblurring: $A = F^H DF$

$$(F^T D^H D F + \tau \nabla^T \nabla)x = \text{rhs}$$

$$(F^T |D|^2 F + \tau \nabla^T \nabla)x = \text{rhs}$$

$\nabla = F^H K F$

$$(F^T |D|^2 F + \tau F^T |K|^2 F)x = \text{rhs}$$

$$F^T (|D|^2 + \tau |K|^2)Fx = \text{rhs}$$

$$x^{k+1} = F^T (|D|^2 + \tau |K|^2)^{-1} F(\text{rhs})$$
Y-UPDATE

minimize \( \mu \| \nabla x \| + \frac{1}{2} \| Ax - f \|^2 \)

\[ \mu \| y \| + \frac{1}{2} \| Ax - f \|^2 + \langle \lambda, \nabla x - y \rangle + \frac{\tau}{2} \| \nabla x - y \|^2 \]

y-update

\[ y^{k+1} = \arg \min_y \mu \| y \| - \langle \lambda^k, y \rangle + \frac{\tau}{2} \| \nabla x^{k+1} - y \|^2 \]

complete square

\[ y^{k+1} = \arg \min_y \mu \| y \| + \frac{\tau}{2} \| y - \nabla x^{k+1} \| - \frac{1}{\tau} \lambda^k \|^2 \]

proximal operator

\[ y^{k+1} = \text{shrink}(\nabla x^{k+1} + \frac{1}{\tau} \lambda^k, \mu / \tau) \]
DEBLURRING ALGORITHM

\[ \minimize \mu |\nabla x| + \frac{1}{2} \|Ax - f\|^2 \]

\[ \mu |y| + \frac{1}{2} \|Ax - f\|^2 + \langle \lambda, \nabla x - y \rangle + \frac{\tau}{2} \|\nabla x - y\|^2 \]

Split Bregman Deblurring

\[ x^{k+1} = F^H(|D|^2 + \tau |K|^2)^{-1} F(F^H D^H F f - \nabla^T \lambda^k + \tau \nabla^T y^k) \]

\[ y^{k+1} = \text{shrink}(\nabla x^{k+1} + \frac{1}{\tau} \lambda^k, \mu / \tau) \]

\[ \lambda^{k+1} = \lambda^k + \tau (\nabla x^{k+1} - y^{k+1}) \]
Graphical model:
• nodes are random variables
• dependent variables connected by edge

Example: “gene finding”
• measure mRNA expression levels
• try to find mRNA strands regulated by same genes

Variables are conditionally independent if there is no path from one to another
\[ x_i = \begin{pmatrix} a_i \\ b_i \\ c_i \\ d_i \end{pmatrix} \]

inverse covariance matrix

\[ p(x) = \frac{1}{(2\pi)^p |\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2} x^T \Sigma^{-1} x\right) \]
LIKELIHOOD MODEL

\[ p(x) = \frac{1}{(2\pi)^p |\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2} x^T \Sigma^{-1} x \right) \]

\[ l(\Sigma) = \prod_i \frac{1}{(2\pi)^p |\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2} x_i^T \Sigma^{-1} x_i \right) \]

\[ ll(\Sigma) = -np \log(2\pi) - \frac{n}{2} \log \det \Sigma - \frac{1}{2} \sum_i x_i^T \Sigma^{-1} x_i \]

\[ \log \det \Sigma + \frac{1}{n} \sum_i x_i^T \Sigma^{-1} x_i \]
LIKELIHOOD MODEL

\[ \log \det \Sigma + \frac{1}{n} \sum_i x_i^T \Sigma^{-1} x_i \]

\[ \frac{1}{n} \sum_i x_i^T \Sigma^{-1} x_i = \frac{1}{n} \sum_i \langle x_i x_i^T, \Sigma^{-1} \rangle = \langle \frac{1}{n} \sum_i x_i x_i^T, \Sigma^{-1} \rangle = \langle S, \Sigma^{-1} \rangle \]

negative log likelihood

\[ \log \det \Sigma + \langle S, \Sigma^{-1} \rangle \]

sparse model

\[ \minimize_{\Sigma^{-1}} \log \det \Sigma + \langle S, \Sigma^{-1} \rangle + |\Sigma^{-1}| \]

change variables

\[ \minimize_X - \log \det X + \langle S, X \rangle + |X| \]
ADMM FOR SICS

\[
\text{minimize } \quad - \log \det X + \langle S, X \rangle + |X| \\
\text{subject to } \quad X = Y
\]

split Bregman form

\[
\text{minimize } \quad - \log \det X + \langle S, X \rangle + |Y| \\
\]

augmented Lagrangian

\[
\max_{\lambda} \min_{X,Y} \quad - \log \det X + \langle S, X \rangle + |Y| + \langle \lambda, X - Y \rangle + \frac{\tau}{2} \|X - Y\|^2
\]
ADMM FOR SICS

augmented Lagrangian

\[
\max_{X,Y} \min_{\lambda} \log \det X + \langle S, X \rangle + |Y| + \langle \lambda, X - Y \rangle + \frac{\tau}{2} \|X - Y\|^2
\]

step 1: minimize for \(Y\)

\[
\min_Y \quad |Y| + \langle \lambda, X - Y \rangle + \frac{\tau}{2} \|Y - X\|^2
\]

\[
\min_Y \quad |Y| + \frac{\tau}{2} \|Y - X - \frac{1}{\tau} \lambda\|^2
\]

solution

\[
Y^{k+1} = \text{shrink}(X^k + \tau^{-1} \lambda^k)
\]
ADMM FOR SICS

augmented Lagrangian

\[
\max_{\lambda} \min_{X,Y} \lambda \log \det X + \langle S, X \rangle + |Y| + \langle \lambda, X - Y \rangle + \frac{\tau}{2} \|X - Y\|^2
\]

step 2: minimize for X

\[
\min_{X} - \log \det X + \langle S, X \rangle + \langle \lambda, X \rangle + \frac{\tau}{2} \|X - Y\|^2
\]

\[
\min_{X} - \log \det X + \frac{\tau}{2} \|X - Y + \tau^{-1}(\lambda + S)\|^2
\]

\[
X = U \Lambda_X U^T \\
Z = U \Lambda_Z U^T
\]
ADMM FOR SICS

step 2: minimize for X

\[
\begin{align*}
\min_X & \quad - \log \det X + \langle S, X \rangle + \langle \lambda, X \rangle + \frac{\tau}{2} \| X - Y \|^2 \\
\min_X & \quad - \log \det X + \frac{\tau}{2} \| X - Y + \tau^{-1}(\lambda + S) \|^2 \\
X & = U \Lambda_X U^T \\
Z & = U \Lambda_Z U^T
\end{align*}
\]

\[
\sum_i - \log \sigma_X^i + \frac{\tau}{2} (\sigma_X^i - \sigma_Z^i)^2 \rightarrow -\frac{1}{\sigma_X^i} + \tau (\sigma_X^i - \sigma_Z^i) = 0
\]

\[
\sigma_X^i = \frac{\sigma_Z^i + \sqrt{(\sigma_Z^i)^2 + 4/\tau}}{2}
\]

final step: \( \lambda^{k+1} = \lambda^k + \tau(X - Y) \)
SCALED ADMM

minimize $f(x) + g(y)$

subject to $Ax + By + c = 0$

augmented Lagrangian

$L_\tau(x, y, \lambda) = f(x) + g(y) + \langle \lambda, Ax + By + c \rangle + \frac{\tau}{2}\|Ax + By + c\|^2$

scaled Lagrangian

$L_\tau(x, y, \lambda) = f(x) + g(y) + \frac{\tau}{2}\|Ax + By + c + \frac{1}{\tau}\lambda\|^2$

These differ by a constant. Why??

penalty function/spring interpretation
SCALED ADMM

minimize $f(x) + g(y)$

subject to $Ax + By + c = 0$

scaled Lagrangian

$L_\tau(x, y, \lambda) = f(x) + g(y) + \frac{\tau}{2}\|Ax + By + c + \frac{1}{\tau}\lambda\|^2$

$\hat{\lambda} \leftarrow \lambda$

$L_\tau(x, y, \hat{\lambda}) = f(x) + g(y) + \frac{\tau}{2}\|Ax + By + c + \hat{\lambda}\|^2$

scaled ADMM

$x^{k+1} = \arg\min_x f(x) + \frac{\tau}{2}\|Ax + By^{k} + c + \hat{\lambda}^k\|^2$

$y^{k+1} = \arg\min_y g(y) + \frac{\tau}{2}\|Ax^{k+1} + By + c + \hat{\lambda}^k\|^2$

$\hat{\lambda}^{k+1} = \hat{\lambda}^k + Ax^{k+1} + By^{k+1} + c$
DISTRIBUTED PROBLEMS

minimize \( g(x) + \sum_{i} f_i(x) \)

example: sparse least squares

minimize \( \mu |x| + \frac{1}{2} \|Ax - b\|^2 \)

\[
A = \begin{pmatrix}
A_1 \\
A_2 \\
\vdots \\
A_N
\end{pmatrix}
\]

minimize \( \mu |x| + \sum_{i} \frac{1}{2} \|A_i x - b_i\|^2 \)

data stored on different servers
CONSENSUS ADMM

minimize \quad g(x) + \sum_{i} f_i(x)

Central server holds global variables: \( z \)
Every client gets local copy of unknowns: \( x_i \)

minimize \quad g(z) + \sum_{i} f_i(x_i)

subject to \quad x_i = z, \ \forall i
CONSENSUS ADMM

\[
\begin{align*}
\text{minimize} & \quad g(z) + \sum_i f_i(x_i) \\
\text{subject to} & \quad x_i = z, \quad \forall i
\end{align*}
\]

scaled augmented Lagrangian

\[
L = g(z) + \sum_i f_i(x_i) + \sum_i \frac{\tau}{2} \|x_i - z + \lambda_i\|^2
\]

consensus ADMM

central server: \(z^{k+1} = \arg \min_z g(z) + \sum_i \frac{\tau}{2} \|x_i^k - z + \lambda_i^k\|^2\)

remote client: \(x_i^{k+1} = \arg \min_{x_i} f_i(x_i) + \frac{\tau}{2} \|x_i - z^{k+1} + \lambda_i^k\|^2\)

remote client: \(\lambda_i^{k+1} = \lambda_i^k + x_i^{k+1} - z^{k+1}\)
CENTRAL STEP

central server: \( \hat{z}^{k+1} = \arg \min_z g(z) + \sum_i \frac{\tau}{2} \| x_i^k - z + \lambda_i^k \|^2 \)

\( \hat{z}^{k+1} = \arg \min_z g(z) + \sum_i \frac{\tau}{2} \| z - (x_i^k + \lambda_i^k) \|^2 \)

\( \hat{z}^{k+1} = \arg \min_z g(z) + \frac{N\tau}{2} \| z - \frac{1}{N} \sum_i (x_i^k + \lambda_i^k) \|^2 \)

average of remote values \( \eta^k \)

\( \hat{z}^{k+1} = \arg \min_z g(z) + \frac{N\tau}{2} \| z - \eta^k \|^2 \)
EXAMPLE: LASSO

minimize \( \mu |x| + \sum_i \frac{1}{2} \| A_i x - b_i \|^2 \)

scaled augmented Lagrangian

\[ L = \mu |z| + \sum_i \frac{1}{2} \| A_i x_i - b_i \|^2 + \sum_i \frac{\tau}{2} \| x_i - z + \lambda_i \|^2 \]

consensus LASSO

MPI reduce: \( \eta^k = \frac{1}{N} \sum_i x_i^k + \lambda_i^k \)

central server: \( z^{k+1} = \arg \min_x \mu |z| + \frac{N \tau}{2} \| z - \eta \|^2 \)

remote client: \( x_i^{k+1} = \arg \min_{x_i} \frac{1}{2} \| A_i x_i - b_i \|^2 + \frac{\tau}{2} \| x_i - z^{k+1} + \lambda_i^k \|^2 \)

remote client: \( \lambda_i^{k+1} = \lambda_i^k + x_i^{k+1} - z^{k+1} \)
UNWRAPPED ADMM

minimize \quad g(x) + f(Ax) = g(x) + \sum_i f_i(A_ix)

Example: SVM

minimize \quad \frac{1}{2}\|x\|^2 + C h(Ax)

A = LD, \quad h = \text{hinge loss}
UNWRAPPED ADMM

\[ \text{minimize} \quad g(x) + f(Ax) = g(x) + \sum_i f_i(A_ix) \]

\[ z = Ax, \quad z_i = A_i x \]

unwrapped form

\[ \text{minimize} \quad g(x) + f(z) \]

subject to \quad z = Ax

scaled augmented Lagrangian

\[ L_\tau(x, z, \lambda) = \text{minimize} \quad g(x) + f(z) + \frac{\tau}{2} \|z - Ax + \lambda\|^2 \]

\[ L_\tau(x, z, \lambda) = \text{minimize} \quad g(x) + \sum_i f_i(z_i) + \frac{\tau}{2} \sum_i \|z_i - A_ix + \lambda_i\|^2 \]
UNWRAPPED ADMM

\[ L_\tau(x, z, \lambda) = \text{minimize} \quad g(x) + \sum_i f_i(z_i) + \frac{\tau}{2} \sum_i \| z_i - A_i x + \lambda_i \|^2 \]

unwrapped ADMM

remote client: \[ z_i^{k+1} = \arg \min_{z_i} f_i(z_i) + \frac{\tau}{2} \| z_i - A_i x^k + \lambda_i^k \|^2 \]

central server: \[ x^{k+1} = \arg \min_x g(x) + \frac{\tau}{2} \sum_i \| z_i^{k+1} - A_i x + \lambda_i^k \|^2 \]

remote client: \[ \lambda_i^{k+1} = \lambda_i^k + z_i^{k+1} - A_i x^{k+1} \]
REMOTE Z-STEP

each node only need a copy of $x$ from central server (MPI broadcast)

remote client: $z_{i}^{k+1} = \arg \min_{z_{i}} f_{i}(z_{i}) + \frac{T}{2} ||z_{i} - A_{i}x_{k} + \lambda_{i}^{k}||^{2}$

entries of $f$ are decoupled, closed form solutions easy
**GLOBAL X-STEP: TRANSPOSE REDUCTION**

central server:  
\[ x^{k+1} = \arg\min_x g(x) + \frac{\tau}{2} \sum_i \| z_i^{k+1} - A_i x + \lambda_i^k \|^2 \]

\[
\partial g(x) + \tau \sum_i A_i^T (A_i x - z_i^{k+1} - \lambda_i^k) = 0
\]

\[
(\partial g(x) + \tau \sum_i A_i^T A_i) x = \sum_i A_i^T (z_i^{k+1} + \lambda_i^k)
\]

computed \textbf{once} on each client and stored/factored on server

computed on each local client

MPI sum/reduce
EXAMPLE: SVM

\[
\begin{align*}
\text{minimize} & \quad \frac{1}{2} \|x\|^2 + Ch(Ax) \\
A &= LD, \quad h(y) = \max\{1 - y, 0\}
\end{align*}
\]

unwrap

\[
\begin{align*}
\text{minimize} & \quad \frac{1}{2} \|x\|^2 + Ch(z) \\
\text{subject to} & \quad z = Ax
\end{align*}
\]

Lagrangian

\[
\begin{align*}
\text{minimize} & \quad \frac{1}{2} \|x\|^2 + Ch(z) + \frac{\tau}{2} \|z - Ax + \lambda\|^2
\end{align*}
\]
NON-DISTRIBUTED SVM

\[
\begin{align*}
\text{minimize} & \quad \frac{1}{2} \|x\|^2 + Ch(z) + \frac{\tau}{2} \|z - Ax + \lambda\|^2 \\
\text{minimize for } z & \\
z^{k+1} & = Ch(z) + \frac{\tau}{2} \|z - Ax^k + \lambda^k\|^2 = \text{prox}_h(Ax^k - \lambda^k, C/\tau) \\
\text{prox}_h(u, \delta) & = \max\{u, \min\{u + \delta, 1\}\} \\
\text{minimize for } x & \\
(I + \tau A^T A)x^{k+1} & = \tau A^T (z^{k+1} + \lambda^k) \\
\text{dual gradient ascent} & \\
\lambda^{k+1} & = \lambda^k + z^{k+1} - Ax^{k+1}
\end{align*}
\]
DISTRIBUTED IMPLEMENTATION

setup phase

MPI Reduce: \( I + \tau A^T A = I + \tau \sum_i A_i^T A_i \)

solve phase

remote client: \( z_{i}^{k+1} = \text{prox}_{h}(A_i x^k - \lambda_i^k, C/\tau) \)

MPI Reduce: \( \tau A^T (z_i^{k+1} + \lambda_i^k) = \tau \sum_i A_i^T (z_i^{k+1} + \lambda_i^k) \)

central server: \( (I + \tau A^T A)x^{k+1} = \tau A^T (z_i^{k+1} + \lambda_i^k) \)

remote client: \( \lambda_{i}^{k+1} = \lambda_i^k + z_i^{k+1} - A_i x^{k+1} \)
EXPERIMENT: LOGISTIC REGRESSION

2K features, 50K data points/core

consensus

transpose reduction
EXPERIMENT: SVM

2K features, 50K data points/core

![Graph showing the relationship between the amount of data and total computation time. The graph indicates a linear increase in computation time as the amount of data increases.](image)
HETEROGENEOUS DATA

logistic regression

360M data points, 2K features, 7200 cores

homogeneous data

heterogeneous data

setup time
FAST ADMM

Gradient

\[ O \left( \frac{1}{k} \right) \]

Nesterov

\[ O \left( \frac{1}{k^2} \right) \]
FAST ADMM

Augmented Lagrangian

\[ L_\tau(x, y, \lambda) = f(x) + g(y) + \langle \lambda, Ax + By + c \rangle + \frac{\tau}{2} \| Ax + By + c \|^2 \]

\[ x^k = \arg \min_x f(x) + \langle \hat{\lambda}^k, Ax \rangle + \frac{\tau}{2} \| Ax + B\hat{y}^k + c \|^2 \]

\[ y^k = \arg \min_y g(y) + \langle \hat{\lambda}^k, By \rangle + \frac{\tau}{2} \| A\hat{x}^k + By + c \|^2 \]

\[ \lambda^k = \hat{\lambda}^k + \tau (Ax^k + BY^k + c) \quad \text{ADMM step} \]

\[ \alpha^{k+1} = \frac{1 + \sqrt{1 + 4(\alpha^k)^2}}{2} \]

\[ \hat{y}^{k+1} = y^k + \frac{\alpha^k - 1}{\alpha^{k+1}} (y^k - y^{k-1}) \]

\[ \hat{\lambda}^{k+1} = \lambda^k + \frac{\alpha^k - 1}{\alpha^{k+1}} (\lambda^k - \lambda^{k-1}) \]
FAST ADMM

Augmented Lagrangian

\[ L_\tau(x, y, \lambda) = f(x) + g(y) + \langle \lambda, Ax + By + c \rangle + \frac{\tau}{2} \| Ax + By + c \|^2 \]

\[ x^k = \arg \min_x f(x) + \langle \hat{\lambda}^k, Ax \rangle + \frac{\tau}{2} \| Ax + B\hat{y}^k + c \|^2 \]

\[ y^k = \arg \min_y g(y) + \langle \hat{\lambda}^k, By \rangle + \frac{\tau}{2} \| A\hat{x}^k + By + c \|^2 \]

\[ \lambda^k = \hat{\lambda}^k + \tau(Ax^k + BY^k + c) \]

\[ \alpha^{k+1} = \frac{1 + \sqrt{1 + 4(\alpha^k)^2}}{2} \]

\[ \hat{y}^{k+1} = y^k + \frac{\alpha^k - 1}{\alpha^{k+1}} (y^k - y^{k-1}) \]

\[ \hat{\lambda}^{k+1} = \lambda^k + \frac{\alpha^k - 1}{\alpha^{k+1}} (\lambda^k - \lambda^{k-1}) \]

momentum step
CONVERGENCE RESULTS

Theorem

Suppose \( f \) and \( g \) are strongly convex and that

\[
\tau^3 < \frac{\sigma_f \sigma_g^2}{\| A^T A \|_{op} \| B^T B \|_{op}^2}
\]

then fast ADMM converges with

\[
c_k < \frac{C \tau \| \hat{\lambda}^1 - \lambda^* \|^2}{(k + 2)^2}.
\]

RESTART SCHEME

- Restart: When \( c_{k+1} > c_k \)
- Reset acceleration factor: \( \alpha_k = 1 \)

**Theorem**

If \( f \) and \( g \) are arbitrary convex functions and the constrained problem is feasible then

\[
\lim_{k \to \infty} c_k = 0
\]

ELASTIC NET

\[
\min_u \lambda_1 |u| + \frac{\lambda_2}{2} \|u\|^2 + \frac{1}{2} \|Au - f\|^2
\]

Random A, Sparsity = 15/40

Goldstein, O'Donoghue, Setzer, Baraniuk. 2012
WHY PDHG?

minimize \( f(Ax) + g(x) \)

minimize \( f(y) + g(x) \)

subject to \( Ax - y = 0 \)

standard augmented Lagrangian

\[
L(x, y, \lambda) = f(y) + g(x) + \langle \lambda, Ax - y \rangle + \frac{\tau}{2} \|Ax - y\|^2
\]

how to minimize for \( x \)?
EXAMPLE: SPLIT BREGMAN

\[
\begin{align*}
\text{minimize} \quad & \mu |\nabla x| + \frac{1}{2} \|Ax - f\|^2 \\
& \mu |y| + \frac{1}{2} \|Ax - f\|^2 + \langle \lambda, \nabla x - y \rangle + \frac{\tau}{2} \|\nabla x - y\|^2
\end{align*}
\]

Split Bregman TV

\[
\begin{align*}
x^{k+1} &= \text{arg min}_x \frac{1}{2} \|Ax - f\|^2 + \langle \lambda^k, \nabla x \rangle + \frac{\tau}{2} \|\nabla x - y^k\|^2 \\
y^{k+1} &= \text{arg min}_y \mu |y| - \langle \lambda^k, y \rangle + \frac{\tau}{2} \|\nabla x^{k+1} - y\|^2 \\
\lambda^{k+1} &= \lambda^k + \tau (\nabla x^{k+1} - y^{k+1})
\end{align*}
\]

x-update

\[(A^T A + \tau \nabla^T \nabla)x = A^T f - \nabla^T \lambda^k + \tau \nabla^T y^k\]
PDHG: SADDLE-POINT PROBLEMS

\[
\min_{x \in X} \max_{y \in Y} f(x) + \langle Ax, y \rangle - g(y)
\]

• Convex functions: \( f \) and \( g \)

• “Link” term: \( \langle Ax, y \rangle \)

• We can evaluate “proximal operators”

\[
J_{\tau F}(\hat{x}) = \arg \min_{x \in X} f(x) + \frac{1}{2\tau} \| x - \hat{x} \|^2
\]

\[
J_{\sigma G}(\hat{y}) = \arg \min_{y \in Y} g(y) + \frac{1}{2\sigma} \| y - \hat{y} \|^2.
\]
HOW PDHG WORKS

$$\min_{x \in X} \max_{y \in Y} f(x) + \langle Ax, y \rangle - g(y)$$

Gradient descent

$$\hat{x} = x_k - \tau A^T y$$

Primal Proximal

$$x_{k+1} = \arg\min f(x) + \frac{1}{2\tau} \|x - \hat{x}\|^2$$

Predict

$$\bar{x} = x_{k+1} + (x_{k+1} - x_k)$$

Gradient Ascent

$$\hat{y} = y_k + \sigma A\bar{x}$$

Dual Proximal

$$y_{k+1} = \arg\min g(y) + \frac{1}{2\sigma} \|y - \hat{y}\|^2$$
FORMING THE SADDLE-POINT PROBLEM

\[ \begin{align*}
\text{minimize} & \quad f(x) + h(Ax) \\
\text{minimize} & \quad f(x) + h(y) \\
\text{subject to} & \quad Ax - y = 0 \\
\text{minimize} & \quad f(x) + \langle \lambda, Ax \rangle - \langle \lambda, y \rangle + h(y)
\end{align*} \]

reminder

\[ h^*(\lambda) = \max_y \langle \lambda, y \rangle - h(y) \quad \Rightarrow \quad -h^*(\lambda) = \min_y -\langle \lambda, y \rangle + h(y) \]

\[ \begin{align*}
\text{minimize} & \quad f(x) + \langle \lambda, Ax \rangle - h^*(\lambda)
\end{align*} \]
EXAMPLE: TV

\[
\min_{x \in X} \max_{y \in Y} f(x) + \langle Ax, y \rangle - g(y)
\]

• The problem:
  \[|\nabla x| + \frac{\mu}{2} \|Ax - f\|^2\]

Note:
  \[|x| = \max_{|y| \leq 1} yx\]

• Re-write TV:
  \[|\nabla x| = \max_{\|y\|_\infty \leq 1} \langle y, \nabla x \rangle\]
  \[= \max_y \langle y, \nabla x \rangle - \chi_\infty(y)\]

• Saddle-Point form:
  \[
  \max_y \min_x \frac{\mu}{2} \|Ax - f\|^2 + \langle y, \nabla x \rangle - \chi_\infty(y)
  \]
SCALED LASSO

\[
\text{minimize} \quad \mu |x| + \|Ax - b\|
\]

\[
\mu |x| = \max_{\|y\|_\infty \leq \mu} \langle y, x \rangle
\]

\[
= \max_{y} \langle y, x \rangle - \mathcal{X}_\infty^\mu (y)
\]

\[
\|Ax - b\| = \max_{\|z\| \leq 1} \langle z, Ax - b \rangle
\]

\[
= \max_{z} \langle z, Ax - b \rangle - \mathcal{X}_2 (z)
\]

saddle form

\[
\min_{x} \max_{y, z} \langle y, x \rangle + \langle z, Ax - b \rangle - \mathcal{X}_2 (z) - \mathcal{X}_\infty^\mu (y)
\]
SCALED LASSO

minimize \( \mu |x| + \| Ax - b \| \)

saddle form

\[
\min_{x} \max_{y,z} \langle y, x \rangle + \langle z, Ax - b \rangle - \mathcal{X}_2(z) - \mathcal{X}^\mu_\infty(y)
\]

minimize for \( x \)

\[
\hat{x} = x^k - \tau (y^k + A^T z^k)
\]

predict

\[
\bar{x} = x^{k+1} + (x^{k+1} - x^k)
\]

maximize dual: two sets of variables

\[
\hat{y} = y^k + \sigma \bar{x} \quad \rightarrow \quad y^{k+1} = \arg \min \mathcal{X}^\mu_\infty(y) + \frac{1}{2\sigma} \| y - \hat{y} \|^2
\]

\[
\hat{z} = z^k + \sigma (A \bar{x} - b) \quad \rightarrow \quad z^{k+1} = \arg \min \mathcal{X}_2(z) + \frac{1}{2\sigma} \| z - \hat{z} \|^2
\]
RESIDUALS

\[
\min \max_{x \in X, \ y \in Y} \ f(x) + \langle Ax, y \rangle - g(y)
\]

- Primal residual \( p(x, y) = \partial f(x) + A^T y \)
- Dual residual \( d(x, y) = \partial g(y) - Ax \)

\[
x^{k+1} = \text{arg min} \ f(x) + \frac{1}{2\tau} \| x - \hat{x} \|^2
\]
RESIDUALS

\[
\min_{x \in X} \max_{y \in Y} f(x) + \langle Ax, y \rangle - g(y)
\]

- **Primal residual**  \( p(x, y) = \partial f(x) + A^T y \)
- **Dual residual**  \( d(x, y) = \partial g(y) - Ax \)

\[
x^{k+1} = \arg \min f(x) + \frac{1}{2\tau} \| x - \hat{x} \|^2
\]

\[
0 \in \partial f(x^{k+1}) + \frac{1}{\tau} (x^{k+1} - \hat{x})
\]

\[
= \partial f(x^{k+1}) + \frac{1}{\tau} (x^{k+1} - x^k) + A^T y^k
\]

\[
p(x, y) = \frac{1}{\tau} (x^k - x^{k+1}) - A^T (y^k - y^{k+1})
\]
RESIDUALS

\[
\min_{x \in X} \max_{y \in Y} f(x) + \langle Ax, y \rangle - g(y)
\]

- Primal residual \( p(x, y) = \partial f(x) + A^T y \)
- Dual residual \( d(x, y) = \partial g(y) - Ax \)

explicit formula

\[
p(x, y) = \frac{1}{\tau} (x^k - x^{k+1}) - A^T (y^k - y^{k+1})
\]

\[
d(x, y) = \frac{1}{\sigma} (y^k - y^{k+1}) - A^T (x^k - x^{k+1})
\]
EXAMPLE: DENOISING

\[
\min_x \frac{\mu}{2} \|x - f\|^2 + |\nabla x|
\]

\(\tau\) Big

\(\tau\) Small

Primal Residual
Dual Residual
RESIDUAL BALANCING

• Start with \( \tau \sigma < \frac{1}{\|A^T A\|} \)

• After each iteration:
  
  • If \( p_k/d_k \) large: increase \( \tau \) decrease \( \sigma \)
  
  • If \( p_k/d_k \) small: decrease \( \tau \) increase \( \sigma \)

• Stability condition is always satisfied, so method should converge
EXPERIMENT: DENOISING

Stability: $\tau \sigma < \frac{1}{8}$

$$\min_x \frac{\mu}{2} \|x - f\|^2 + |\nabla x|$$

$\tau = 1, \sigma = 1/10$ \hspace{1cm} $\tau = 1/10, \sigma = 1$

$\tau$ Big

$\tau$ Small
EXPERIMENT: DENOISING

Stability: $\tau\sigma < \frac{1}{8}$

Student Version of MATLAB

Primal Residual
Dual Residual

Stepsize Swap

Residuals

Iteration

$\tau = \frac{1}{10}$
THEOREM

\[ \phi_k = \max \left\{ \frac{\tau_k - \tau_{k+1}}{\tau_k}, \frac{\sigma_k - \sigma_{k+1}}{\sigma_k}, 0 \right\} \]

Theorem

PDHG converges if the following conditions hold:

A  The stepsizes \( \{\tau_k\} \) and \( \{\sigma_k\} \) are bounded.

B  The sequence \( \{\phi_k\} \) is summable, i.e., \( \sum_{k \geq 0} \phi_k < \infty \).

C  One of the following holds:

   C1  There is a constant \( L \) with

   \[ \tau_k \sigma_k < L < \rho(A^T A)^{-1}. \]

   C2  Either \( X \) or \( Y \) is bounded, and

   \[ \frac{\gamma}{\tau_k} \|x_{k+1} - x_k\|^2 + \frac{\gamma}{\sigma_k} \|y_{k+1} - y_k\|^2 > 2 \langle A(x_{k+1} - x_k), y_{k+1} - y_k \rangle. \]
BACKTRACKING

Pick $\tau, \sigma$ with $\tau \sigma > \|A^T A\|^{-1}$

$\gamma = 0.75$

While $p^k$ and $d^k$ are “big:”

Do PDHG, get $x^{k+1}$ and $y^{k+1}$

If $\gamma \frac{\|x^{k+1} - x_k\|^2}{\tau} + \gamma \frac{\|y^{k+1} - y_k\|^2}{\sigma} < 2\langle A(x^{k+1} - x_k), y^{k+1} - y_k \rangle$

$\tau \leftarrow \tau / 2, \quad \sigma \leftarrow \sigma / 2,$

Redo PDHG, get $x^{k+1}$ and $y^{k+1}$
BACKTRACKING

Pick $\tau, \sigma$ with $\tau \sigma > \|A^T A\|^{-1}$ \hspace{1cm} $\gamma = 0.75$

While $p^k$ and $d^k$ are “big:”

Do PDHG, get $x^{k+1}$ and $y^{k+1}$

If $\dfrac{\gamma}{\tau} \|x^{k+1} - x_k\|^2 + \dfrac{\gamma}{\sigma} \|y^{k+1} - y_k\|^2 < 2 \langle A(x^{k+1} - x_k), y^{k+1} - y_k \rangle$

$\tau \leftarrow \tau / 2$, $\sigma \leftarrow \sigma / 2$,

Redo PDHG, get $x^{k+1}$ and $y^{k+1}$
\(\alpha = 0.5, \eta = 0.95\)

Start with \(\alpha = 0.5, \eta = 0.95\)

While \(p^k\) and \(d^k\) are “big:”

Do PDHG and backtracking

If \(p^k > 2d^k\):

\[
\begin{align*}
\tau &\leftarrow \tau / (1 - \alpha) \\
\sigma &\leftarrow \sigma (1 - \alpha) \\
\alpha &\leftarrow \alpha \eta
\end{align*}
\]

If \(2p^k < d^k\):

\[
\begin{align*}
\tau &\leftarrow \tau (1 - \alpha) \\
\sigma &\leftarrow \sigma / (1 - \alpha) \\
\alpha &\leftarrow \alpha \eta
\end{align*}
\]

...also works for ADMM: He and Yuan ‘12
ADAPTIVITY

\[ \alpha = 0.5, \eta = 0.95 \]

Start with \( \alpha = 0.5, \eta = 0.95 \)

While \( p^k \) and \( d^k \) are "big:"

Do PDHG and backtracking

If \( p^k > 2d^k \):

\[
\begin{align*}
\tau &\leftarrow \tau/(1 - \alpha) \\
\sigma &\leftarrow \sigma(1 - \alpha) \\
\alpha &\leftarrow \alpha \eta
\end{align*}
\]

...also works for ADMM: He and Yuan '12
EXAMPLE: DENOISING

\[
\min_{x} \frac{\mu}{2} \|x - f\|^2 + |\nabla x|
\]
EXAMPLE: DENOISING

ROF Convergence Curves, $\mu = 0.05$

- Adapt: Backtrack
- Adapt: $\tau \sigma = L$
- Const: $\tau = \sqrt{L}$
- Const: $\tau$-final

Energy Gap

Residuals

Iteration
EXAMPLE: DENOISING

Primal Stepsize ($\tau_k$)

Adapt: Backtrack
Adapt: $\tau \sigma = L$
ADAPTIVE ADMM

Lagrangian

\[ L(x, y, \lambda) = f(x) + g(y) + \langle \lambda, Ax + By + c \rangle \]

dual-residual:

\[ \tau A^T B y^{k+1} - \tau A^T B y^k \]

primal-residual:

\[ A x^{k+1} + B y^{k+1} + c \]

**Theorem** (He and Yuan ‘12)

If we have

\[ \sum |t^{k+1} - t^k| < \infty, \]

then

\[ \lim_{k \to \infty} \| p^k \|^2 + \| d^k \|^2 = 0. \]
BB FOR ADMM

minimize \( H(u) + G(v) \)
subject to \( Au + Bv = b \)

Augmented Lagrangian

\[
\max_{\lambda} \min_{u,v} H(u) + G(v) + \langle \lambda, b - Au - Bv \rangle + \frac{\tau}{2} \|b - Au - Bv\|^2
\]

how to choose?
SPECTRAL STEPSIZE RULES

minimize \( H(u) + G(v) \)
subject to \( Au + Bv = b \)

Advantages
- Fast!  Superlinear for some problems
- Automated
- Spectral adaptive methods
- Gradient descent: Barzilai-Borwein
- Forward-backward: SpaRSA
- Constrained problems: ???

“Spectral” approximation

\[
\begin{align*}
    y &= \frac{\alpha}{2} \|x\|^2 \\
    y &= f(x)
\end{align*}
\]

optimal stepsize = \( \frac{1}{\alpha} \)
ADAPTIVE ADMM

\[
\begin{align*}
\text{minimize} & \quad H(u) + G(v) \\
\text{subject to} & \quad Au + Bv = b
\end{align*}
\]

dual problem: no constraints

\[
\begin{align*}
\min_\lambda & \quad H^*(A^T \lambda) - \langle \lambda, b \rangle + G^*(B^T \lambda) \\
& \quad \alpha \frac{1}{2} \| \lambda \|^2 \\
& \quad \beta \frac{1}{2} \| \lambda \|^2 \\
\text{optimal stepsize} & \quad \frac{1}{\sqrt{\alpha \beta}}
\end{align*}
\]
ADAPTIVE ADMM

\[
\begin{align*}
\text{minimize} & \quad H(u) + G(v) \\
\text{subject to} & \quad Au + Bv = b
\end{align*}
\]

\[
\begin{align*}
\min_{\lambda} & \quad H^*(A^T\lambda) - \langle \lambda, b \rangle + G^*(B^T\lambda) \\
\alpha & = \frac{1}{\sqrt{\alpha\beta}} \\
\beta & = \frac{\langle \lambda_k - \lambda_0 \rangle^T B (v_k - v_{k_0})}{\| \lambda_k - \lambda_0 \|^2} \\
\text{optimal stepsize} & = \frac{1}{\sqrt{\alpha\beta}} \\
\alpha & = \frac{\langle \hat{\lambda}_k - \hat{\lambda}_0 \rangle^T A (u_k - u_{k_0})}{\| \hat{\lambda}_k - \hat{\lambda}_0 \|^2}
\end{align*}
\]

curvatures are “free” given ADMM iterates
Zheng Xu, Mario Figueiredo, Tom Goldstein. “Adaptive ADMM with spectral penalty parameter selection.” 2017
WHAT’S A GRAPH CUT

cut function

\[ x : V \to \{0, 1\} \]

\[ \sum_{i,j} |x_i - x_j|w_{i,j} + \sum_i x_if(i) \]

must be positive (why?)

can be anything

\[ x = 1 \]

\[ x = 0 \]
MIN CUT PROBLEM

Cut graph into parts such that s and t lie in different chunks

graph cuts solves this

\[
\begin{align*}
\text{minimize} & \quad \sum_{i,j} |x_i - x_j| w_{i,j} + \sum_i x_i f(i) \\
\text{subject to} & \quad x_s = 1, x_t = 0
\end{align*}
\]
EXAMPLE: SEGMENTATION

\[ w_{ij} = \frac{e^{\|v_i - v_j\|^2 / \sigma^2}}{d(i, j)} \]

- similar pixels get big weight
- differing pixels get small weight
HISTOGRAM MODELS

user marks up image

build gaussian mixture model of regions

\[ f(i) = \log(p(v_i \in R_1)) - \log(p(v_i \in R_0)) \]

\[ \min_x \sum_{i,j} |x_i - x_j| w_{i,j} + \sum_i x_i f(i) \]

Rother et. al, “GrabCut,” ‘04
SAMPLE RESULTS

GrabCut (SIGGRAPH ‘04):
• iterative segmentation
• faded alpha channel at edge
• anti-aliasing

Rother et. al, “GrabCut,” ‘04
NUMERICS

graph cuts solves this

$$\text{minimize } \sum_{i,j} |x_i - x_j|w_{i,j} + \sum_i x_i f(i)$$

subject to $x_s = 1, x_t = 0$

classical methods: **dual** LP

dual(min cut) = max flow

drawback:
hard to implement
does not parallelize well
(bad for GPU)
PDHG APPROACH

Graph cuts solves this

\[
\text{minimize } \sum_{i,j} |x_i - x_j| w_{i,j} + \sum_i x_i f(i)
\]

Saddle-point form

\[
\max_{-w_{ij} \leq \lambda_{ij} \leq w_{ij}} \min_{0 \leq x_{ij} \leq 1} \sum_{i,j} \lambda_{ij} (x_i - x_j) + \sum_i x_i f(i)
\]

Use graph gradient

\[
\max_{-w_{ij} \leq \lambda_{ij} \leq w_{ij}} \min_{0 \leq x_{ij} \leq 1} \langle \lambda, \nabla x \rangle + \langle x, f \rangle
\]
PDHG APPROACH

use graph gradient

\[
\begin{align*}
\max_{-\lambda_{ij} \leq \lambda_{ij} \leq \lambda_{ij}} \min_{0 \leq x_{ij} \leq 1} & \langle \lambda, \nabla x \rangle + \langle x, f \rangle \\
\hat{x} &= x^k - \tau (\nabla_g^T \lambda^k + f) \\
x^{k+1} &= \min \{\max \{\hat{x}, 0\}, 1\} \\
\bar{x} &= x^{k+1} + (x^{k+1} - x^k) \\
\hat{\lambda} &= \lambda^k + \sigma \nabla_g \bar{x} \\
\lambda_{ij}^{k+1} &= \min \{\max \{\hat{\lambda}_{ij}, -w_{ij}\}, w_{ij}\}
\end{align*}
\]

minimize for \(x\)

maximize for \(y\)

All entries of primal/dual variables updated simultaneously (GPU)
WHY LINEARIZED PDHG?

**ADMM**

\[ L_\tau(x, \lambda) = f(x) + \langle \lambda, Ax + b \rangle + \frac{\tau}{2} \|Ax + b\|^2 \]

**PDHG**

\[
\min_{x \in X} \max_{y \in Y} f(x) + \langle Ax, y \rangle - g(y)
\]

What if you can’t do this?

note: linearization applies to all methods discussed above
MODIFIED PROX STEP

\[ f(x) + \frac{1}{2\tau} \|x - \hat{x}\|^2 \]

replace this distance metric

recall...

\[ f(x) \leq f(x^k) + \langle x - x^k, \nabla f(x^k) \rangle + \frac{1}{2\tau} \|x - x^k\|^2 \]

small enough

Valid distance metric

\[ d(x, x^k) = f(x^k) - f(x) + \langle x - x^k, \nabla f(x^k) \rangle + \frac{1}{2\tau} \|x - x^k\|^2 \]
MODIFIED PROX STEP

\[ f(x) + \frac{1}{2\tau} \|x - \hat{x}\|^2 \]

new prox step

\[ x^{k+1} = \min f(x) + d(x, x^k) \]

\[ d(x, x^k) = f(x^k) - f(x) + \langle x - x^k, \nabla f(x^k) \rangle + \frac{1}{2\tau} \|x - x^k\|^2 \]

simplify

\[ x^{k+1} = \arg \min f(x^k) + \langle x - x^k, \nabla f(x^k) \rangle + \frac{1}{2\tau} \|x - x^k\|^2 \]

\[ x^{k+1} = x^k - \tau \nabla f(x^k) \]
LINEARIZED PDHG

\[
\max_y \min_x h(x) + f(x) + \langle Ax, y \rangle - g(y)
\]

**Gradient Descent**
\[
\hat{x} = x^k - \tau(\nabla f(x^k) + A^T y^k)
\]

**Primal Proximal**
\[
x^{k+1} = \text{prox}_h(\hat{x}, \tau)
\]

**Predict**
\[
\bar{x} = x_{k+1} + (x_{k+1} - x_k)
\]

**Gradient Ascent**
\[
\hat{y} = y_k + \tau A\bar{x}
\]

**Dual Proximal**
\[
y_{k+1} = \arg\min_y g(y) + \frac{1}{2\sigma} \|y - \hat{y}\|^2
\]
The iterates of linearized PDHG converge if the stepsizes satisfy the stability condition

\[
\frac{1}{\tau} - \sigma \| A^T A \|_{op} \geq \frac{L_f}{2}.
\]

**Theorem**

The iterates of linearized PDHG converge if the stepsizes satisfy the stability condition

\[
\max_y \min_x h(x) + f(x) + \langle Ax, y \rangle - g(y)
\]