

Dykstra's Algorithm, ADMM, and Coordinate Descent: Connections, Insights, and Extensions

Ryan Tibshirani

Depts. of Statistics & Machine Learning
Carnegie Mellon University

<http://www.stat.cmu.edu/~ryantibs/talks/dykstra-cd-2018.pdf>

How it began: additive models

Given data $(x_i, y_i) \in \mathbb{R}^d \times \mathbb{R}$, $i = 1, \dots, n$, for (potentially) large d .

How it began: additive models

Given data $(x_i, y_i) \in \mathbb{R}^d \times \mathbb{R}$, $i = 1, \dots, n$, for (potentially) large d . Consider an **additive model** fitted using **higher-order total variation** regularization, the solution of

$$\min_{f_1, \dots, f_d} \frac{1}{2} \sum_{i=1}^n \left(y_i - \sum_{j=1}^d f_j(x_{ij}) \right)^2 + \lambda \sum_{j=1}^d \text{TV}(f_j^{(k)})$$

How it began: additive models

Given data $(x_i, y_i) \in \mathbb{R}^d \times \mathbb{R}$, $i = 1, \dots, n$, for (potentially) large d . Consider an **additive model** fitted using **higher-order total variation** regularization, the solution of

$$\min_{f_1, \dots, f_d} \frac{1}{2} \sum_{i=1}^n \left(y_i - \sum_{j=1}^d f_j(x_{ij}) \right)^2 + \lambda \sum_{j=1}^d \text{TV}(f_j^{(k)})$$

For computational sake, we can approximate this by

$$\min_{\theta_1, \dots, \theta_d \in \mathbb{R}^n} \frac{1}{2} \sum_{i=1}^n \left(y_i - \sum_{j=1}^d \theta_{ij} \right)^2 + \lambda \sum_{j=1}^d \|\Delta_j^{(k+1)} \theta_j\|_1$$

where each $\Delta_j^{(k+1)}$, $j = 1, \dots, d$ is a special $(k+1)$ st order discrete derivative operator.

How it began: additive models

Given data $(x_i, y_i) \in \mathbb{R}^d \times \mathbb{R}$, $i = 1, \dots, n$, for (potentially) large d . Consider an **additive model** fitted using **higher-order total variation** regularization, the solution of

$$\min_{f_1, \dots, f_d} \frac{1}{2} \sum_{i=1}^n \left(y_i - \sum_{j=1}^d f_j(x_{ij}) \right)^2 + \lambda \sum_{j=1}^d \text{TV}(f_j^{(k)})$$

For computational sake, we can approximate this by

$$\min_{\theta_1, \dots, \theta_d \in \mathbb{R}^n} \frac{1}{2} \sum_{i=1}^n \left(y_i - \sum_{j=1}^d \theta_{ij} \right)^2 + \lambda \sum_{j=1}^d \|\Delta_j^{(k+1)} \theta_j\|_1$$

where each $\Delta_j^{(k+1)}$, $j = 1, \dots, d$ is a special $(k+1)$ st order discrete derivative operator. We call this k th order **additive trend filtering**

Additive trend filtering

Why additive trend filtering?

Additive trend filtering

Why additive trend filtering? Briefly: (1) computationally efficient,

Additive trend filtering

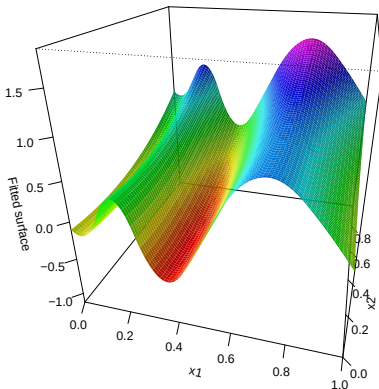
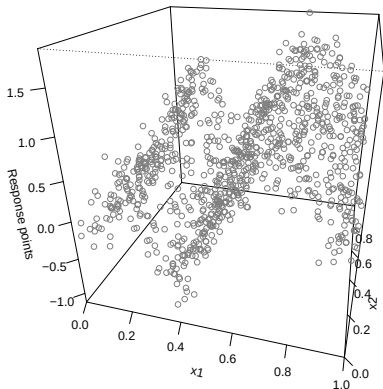
Why additive trend filtering? Briefly: (1) computationally efficient, (2) locally adaptive,

Additive trend filtering

Why additive trend filtering? Briefly: (1) computationally efficient, (2) locally adaptive, (3) naturally extrapolates to smooth surface

Additive trend filtering

Why additive trend filtering? Briefly: (1) computationally efficient, (2) locally adaptive, (3) naturally extrapolates to smooth surface



(Sadhanala and T., 2017)

Trend filtering dual

The **dual** of the additive trend filtering problem:

$$\min_{u \in \mathbb{R}^n} \|y - u\|_2^2 \quad \text{s.t.} \quad u \in C_1 \cap \cdots \cap C_d,$$

where $C_j = \{(\Delta_j^{(k+1)})^T v_j : \|v_j\|_\infty \leq \lambda\}, \quad j = 1, \dots, d$

Trend filtering dual

The **dual** of the additive trend filtering problem:

$$\min_{u \in \mathbb{R}^n} \|y - u\|_2^2 \quad \text{s.t.} \quad u \in C_1 \cap \cdots \cap C_d,$$

$$\text{where } C_j = \{(\Delta_j^{(k+1)})^T v_j : \|v_j\|_\infty \leq \lambda\}, \quad j = 1, \dots, d$$

Primal and dual solutions $\hat{\theta}, \hat{u}$ related by $\hat{u} = y - \sum_{j=1}^d \hat{\theta}_j$

Trend filtering dual

The **dual** of the additive trend filtering problem:

$$\min_{u \in \mathbb{R}^n} \|y - u\|_2^2 \quad \text{s.t.} \quad u \in C_1 \cap \cdots \cap C_d,$$

$$\text{where } C_j = \{(\Delta_j^{(k+1)})^T v_j : \|v_j\|_\infty \leq \lambda\}, \quad j = 1, \dots, d$$

Primal and dual solutions $\hat{\theta}, \hat{u}$ related by $\hat{u} = y - \sum_{j=1}^d \hat{\theta}_j$

Dual problem is projection onto intersection of sets $C_1, \dots, C_d$.

Trend filtering dual

The **dual** of the additive trend filtering problem:

$$\min_{u \in \mathbb{R}^n} \|y - u\|_2^2 \quad \text{s.t.} \quad u \in C_1 \cap \cdots \cap C_d,$$

$$\text{where } C_j = \{(\Delta_j^{(k+1)})^T v_j : \|v_j\|_\infty \leq \lambda\}, \quad j = 1, \dots, d$$

Primal and dual solutions $\hat{\theta}, \hat{u}$ related by $\hat{u} = y - \sum_{j=1}^d \hat{\theta}_j$

Dual problem is projection onto intersection of sets $C_1, \dots, C_d$.

Motivating questions:

Trend filtering dual

The **dual** of the additive trend filtering problem:

$$\min_{u \in \mathbb{R}^n} \|y - u\|_2^2 \quad \text{s.t.} \quad u \in C_1 \cap \cdots \cap C_d,$$

$$\text{where } C_j = \{(\Delta_j^{(k+1)})^T v_j : \|v_j\|_\infty \leq \lambda\}, \quad j = 1, \dots, d$$

Primal and dual solutions $\hat{\theta}, \hat{u}$ related by $\hat{u} = y - \sum_{j=1}^d \hat{\theta}_j$

Dual problem is projection onto intersection of sets $C_1, \dots, C_d$.

Motivating questions:

- Is backfitting—i.e., block coordinate descent—some kind of **alternating projections** in the dual?

Trend filtering dual

The **dual** of the additive trend filtering problem:

$$\min_{u \in \mathbb{R}^n} \|y - u\|_2^2 \quad \text{s.t.} \quad u \in C_1 \cap \cdots \cap C_d,$$

where $C_j = \{(\Delta_j^{(k+1)})^T v_j : \|v_j\|_\infty \leq \lambda\}, \quad j = 1, \dots, d$

Primal and dual solutions $\hat{\theta}, \hat{u}$ related by $\hat{u} = y - \sum_{j=1}^d \hat{\theta}_j$

Dual problem is projection onto intersection of sets $C_1, \dots, C_d$.

Motivating questions:

- Is backfitting—i.e., block coordinate descent—some kind of **alternating projections** in the dual?
- Can we port over methods for **parallel projections** to primal problem?

Outline

Outline:

- Dykstra, ADMM, and CD
- Coordinate descent for the lasso
- Parallel coordinate descent
- Nonquadratic loss: equivalences
- Nonquadratic loss: parallel methods
- Back to additive models

Dykstra, ADMM, and CD

Dijkstra's algorithm

Best approximation problem (projection):
given closed, convex sets $C_1, \dots, C_d \subseteq \mathbb{R}^n$,
nonempty intersection, and $y \in \mathbb{R}^n$, solve

$$\min_{u \in \mathbb{R}^n} \|y - u\|_2^2 \quad \text{s.t.} \quad u \in C_1 \cap \dots \cap C_d$$

Dykstra's algorithm

Best approximation problem (projection):
given closed, convex sets $C_1, \dots, C_d \subseteq \mathbb{R}^n$,
nonempty intersection, and $y \in \mathbb{R}^n$, solve

$$\min_{u \in \mathbb{R}^n} \|y - u\|_2^2 \quad \text{s.t.} \quad u \in C_1 \cap \dots \cap C_d$$

Dykstra's algorithm (Dykstra, 1983; Boyle
and Dykstra, 1986):

Dykstra's algorithm

Best approximation problem (projection):
given closed, convex sets $C_1, \dots, C_d \subseteq \mathbb{R}^n$,
nonempty intersection, and $y \in \mathbb{R}^n$, solve

$$\min_{u \in \mathbb{R}^n} \|y - u\|_2^2 \quad \text{s.t.} \quad u \in C_1 \cap \dots \cap C_d$$

Dykstra's algorithm (Dykstra, 1983; Boyle
and Dykstra, 1986):



Richard Dykstra, U.
of Iowa

Dykstra's algorithm

Best approximation problem (projection):
given closed, convex sets $C_1, \dots, C_d \subseteq \mathbb{R}^n$,
nonempty intersection, and $y \in \mathbb{R}^n$, solve

$$\min_{u \in \mathbb{R}^n} \|y - u\|_2^2 \quad \text{s.t.} \quad u \in C_1 \cap \dots \cap C_d$$

Dykstra's algorithm (Dykstra, 1983; Boyle
and Dykstra, 1986): initialize $u_d^{(0)} = y$,
 $z_1^{(0)} = \dots = z_d^{(0)} = 0$, and repeat

$$\left. \begin{aligned} u_0^{(k)} &= u_d^{(k-1)} \\ u_i^{(k)} &= P_{C_i}(u_{i-1}^{(k)} + z_i^{(k-1)}) \\ z_i^{(k)} &= u_{i-1}^{(k)} + z_i^{(k-1)} - u_i^{(k)} \end{aligned} \right\} \quad i = 1, \dots, d$$



Richard Dykstra, U.
of Iowa

Dijkstra's algorithm (cont.)

Some notes:

- Dijkstra is a statistician! (Main work is on shape constraints)

Dijkstra's algorithm (cont.)

Some notes:

- Dijkstra is a statistician! (Main work is on shape constraints)
- Cf. the simpler **alternating projections algorithm** (von Neumann, 1950; Halperin, 1962) which does not generally converge

Dykstra's algorithm (cont.)

Some notes:

- Dykstra is a statistician! (Main work is on shape constraints)
- Cf. the simpler **alternating projections algorithm** (von Neumann, 1950; Halperin, 1962) which does not generally converge
- When sets are affine spaces, equivalent to **Hildreth's algorithm** (Hildreth, 1957) for quadratic programming

Dykstra's algorithm (cont.)

Some notes:

- Dykstra is a statistician! (Main work is on shape constraints)
- Cf. the simpler **alternating projections algorithm** (von Neumann, 1950; Halperin, 1962) which does not generally converge
- When sets are affine spaces, equivalent to **Hildreth's algorithm** (Hildreth, 1957) for quadratic programming
- Theory and extensions thoroughly developed over the years by Bauschke, Borwein, Bregman, Censor, Combettes, Deutsch

Coordinate descent

Regularized regression problem: given $y \in \mathbb{R}^n$, $\Phi \in \mathbb{R}^{n \times p}$, and convex functions $h_i : \mathbb{R}^{p_i} \rightarrow \mathbb{R}$, $i = 1, \dots, d$, solve

$$\min_{w \in \mathbb{R}^p} \frac{1}{2} \|y - \Phi w\|_2^2 + \sum_{i=1}^d h_i(w_i),$$

with $w = (w_1, \dots, w_d)$ a block decomposition

Coordinate descent

Regularized regression problem: given $y \in \mathbb{R}^n$, $\Phi \in \mathbb{R}^{n \times p}$, and convex functions $h_i : \mathbb{R}^{p_i} \rightarrow \mathbb{R}$, $i = 1, \dots, d$, solve

$$\min_{w \in \mathbb{R}^p} \frac{1}{2} \|y - \Phi w\|_2^2 + \sum_{i=1}^d h_i(w_i),$$

with $w = (w_1, \dots, w_d)$ a block decomposition

Coordinate descent or **CD** (??; Wang, 1963):

Coordinate descent

Regularized regression problem: given $y \in \mathbb{R}^n$, $\Phi \in \mathbb{R}^{n \times p}$, and convex functions $h_i : \mathbb{R}^{p_i} \rightarrow \mathbb{R}$, $i = 1, \dots, d$, solve

$$\min_{w \in \mathbb{R}^p} \frac{1}{2} \|y - \Phi w\|_2^2 + \sum_{i=1}^d h_i(w_i),$$

with $w = (w_1, \dots, w_d)$ a block decomposition

Coordinate descent or **CD** (??; Warga, 1963):



Paul Tseng, U.
of Washington

Coordinate descent

Regularized regression problem: given $y \in \mathbb{R}^n$, $\Phi \in \mathbb{R}^{n \times p}$, and convex functions $h_i : \mathbb{R}^{p_i} \rightarrow \mathbb{R}$, $i = 1, \dots, d$, solve

$$\min_{w \in \mathbb{R}^p} \frac{1}{2} \|y - \Phi w\|_2^2 + \sum_{i=1}^d h_i(w_i),$$

with $w = (w_1, \dots, w_d)$ a block decomposition

Coordinate descent or **CD** (??; Warga, 1963):
initialize $w^{(0)} = 0$, and repeat

$$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} \frac{1}{2} \left\| y - \sum_{j < i} \Phi_j w_j^{(k)} - \sum_{j > i} \Phi_j w_j^{(k-1)} - \Phi_i w_i \right\|_2^2 + h_i(w_i),$$
$$i = 1, \dots, d$$



Paul Tseng, U.
of Washington

Coordinate descent (cont.)

Some notes:

- Dates back to the foundation of optimization as a discipline?

Coordinate descent (cont.)

Some notes:

- Dates back to the foundation of optimization as a discipline?
- For linear systems, equivalent to **Gauss-Seidel iterations**

Coordinate descent (cont.)

Some notes:

- Dates back to the foundation of optimization as a discipline?
- For linear systems, equivalent to **Gauss-Seidel iterations**
- Definitive theory by Bertsekas, Luo, and most notably Tseng

Coordinate descent (cont.)

Some notes:

- Dates back to the foundation of optimization as a discipline?
- For linear systems, equivalent to **Gauss-Seidel iterations**
- Definitive theory by Bertsekas, Luo, and most notably Tseng
- Huge revival of interest in machine learning and statistics over the last 10 years; see Wright (2015)

Coordinate descent (cont.)

Some notes:

- Dates back to the foundation of optimization as a discipline?
- For linear systems, equivalent to **Gauss-Seidel iterations**
- Definitive theory by Bertsekas, Luo, and most notably Tseng
- Huge revival of interest in machine learning and statistics over the last 10 years; see Wright (2015)
- Lots of interesting theory and extensions still being developed

Equivalence of Dykstra and CD

Suppose that, for $i = 1, \dots, d$,

$$h_i(v) = \max_{d \in D_i} \langle d, v \rangle, \quad C_i = (\Phi_i^T)^{-1}(D_i) = \{v \in \mathbb{R}^n : \Phi_i^T v \in D_i\}$$

Equivalence of Dykstra and CD

Suppose that, for $i = 1, \dots, d$,

$$h_i(v) = \max_{d \in D_i} \langle d, v \rangle, \quad C_i = (\Phi_i^T)^{-1}(D_i) = \{v \in \mathbb{R}^n : \Phi_i^T v \in D_i\}$$

(Each h_i is a support function; includes seminorms as special case)

Equivalence of Dykstra and CD

Suppose that, for $i = 1, \dots, d$,

$$h_i(v) = \max_{d \in D_i} \langle d, v \rangle, \quad C_i = (\Phi_i^T)^{-1}(D_i) = \{v \in \mathbb{R}^n : \Phi_i^T v \in D_i\}$$

(Each h_i is a support function; includes seminorms as special case)

Then:

Equivalence of Dykstra and CD

Suppose that, for $i = 1, \dots, d$,

$$h_i(v) = \max_{d \in D_i} \langle d, v \rangle, \quad C_i = (\Phi_i^T)^{-1}(D_i) = \{v \in \mathbb{R}^n : \Phi_i^T v \in D_i\}$$

(Each h_i is a support function; includes seminorms as special case)

Then:

- Regularized regression problem and best approximation problem are **duals**.

Equivalence of Dykstra and CD

Suppose that, for $i = 1, \dots, d$,

$$h_i(v) = \max_{d \in D_i} \langle d, v \rangle, \quad C_i = (\Phi_i^T)^{-1}(D_i) = \{v \in \mathbb{R}^n : \Phi_i^T v \in D_i\}$$

(Each h_i is a support function; includes seminorms as special case)

Then:

- Regularized regression problem and best approximation problem are **duals**. Solutions $\hat{w}, \hat{u}$ related by $\hat{u} = y - \Phi \hat{w}$

Equivalence of Dykstra and CD

Suppose that, for $i = 1, \dots, d$,

$$h_i(v) = \max_{d \in D_i} \langle d, v \rangle, \quad C_i = (\Phi_i^T)^{-1}(D_i) = \{v \in \mathbb{R}^n : \Phi_i^T v \in D_i\}$$

(Each h_i is a support function; includes seminorms as special case)

Then:

- Regularized regression problem and best approximation problem are **duals**. Solutions $\hat{w}, \hat{u}$ related by $\hat{u} = y - \Phi \hat{w}$
- Coordinate descent and Dykstra's algorithm are **equivalent**,

Equivalence of Dykstra and CD

Suppose that, for $i = 1, \dots, d$,

$$h_i(v) = \max_{d \in D_i} \langle d, v \rangle, \quad C_i = (\Phi_i^T)^{-1}(D_i) = \{v \in \mathbb{R}^n : \Phi_i^T v \in D_i\}$$

(Each h_i is a support function; includes seminorms as special case)

Then:

- Regularized regression problem and best approximation problem are **duals**. Solutions $\hat{w}, \hat{u}$ related by $\hat{u} = y - \Phi \hat{w}$
- Coordinate descent and Dykstra's algorithm are **equivalent**, in that at all iterations

$$z_i^{(k)} = \Phi_i w_i^{(k)}, \quad u_i^{(k)} = y - \sum_{j \leq i} \Phi_j w_j^{(k)} - \sum_{j > i} \Phi_j w_j^{(k-1)},$$

$i = 1, \dots, d$

Equivalence of Dykstra and CD (cont.)

Dykstra, iteration i	Coordinate descent, iteration i
$u_i^{(k)} = P_{C_i}(u_{i-1}^{(k)} + z_i^{(k-1)})$ $z_i^{(k)} = u_{i-1}^{(k)} + z_i^{(k-1)} - u_i^{(k)}$	$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} \frac{1}{2} \left\ y - \sum_{j < i} \Phi_j w_j^{(k)} - \sum_{j > i} \Phi_j w_j^{(k-1)} - \Phi_i w_i \right\ _2^2 + h_i(w_i),$

Equivalence of Dykstra and CD (cont.)

Dykstra, iteration i	Coordinate descent, iteration i
$u_i^{(k)} = P_{C_i}(u_{i-1}^{(k)} + z_i^{(k-1)})$ $z_i^{(k)} = u_{i-1}^{(k)} + z_i^{(k-1)} - u_i^{(k)}$	$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} \frac{1}{2} \left\ y - \sum_{j < i} \Phi_j w_j^{(k)} - \sum_{j > i} \Phi_j w_j^{(k-1)} - \Phi_i w_i \right\ _2^2 + h_i(w_i),$

Key connection:

Equivalence of Dykstra and CD (cont.)

Dykstra, iteration i	Coordinate descent, iteration i
$u_i^{(k)} = P_{C_i}(u_{i-1}^{(k)} + z_i^{(k-1)})$ $z_i^{(k)} = u_{i-1}^{(k)} + z_i^{(k-1)} - u_i^{(k)}$	$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} \frac{1}{2} \left\ y - \sum_{j < i} \Phi_j w_j^{(k)} - \sum_{j > i} \Phi_j w_j^{(k-1)} - \Phi_i w_i \right\ _2^2 + h_i(w_i),$

Key connection: for $S \subseteq \mathbb{R}^q$, and $h_S(x) = \max_{s \in S} \langle s, x \rangle$, we have

$$(\text{Id} - P_S)(r) = \arg \min_{x \in \mathbb{R}^q} \frac{1}{2} \|r - x\|_2^2 + h_S(x)$$

Equivalence of Dykstra and CD (cont.)

Dykstra, iteration i	Coordinate descent, iteration i
$u_i^{(k)} = P_{C_i}(u_{i-1}^{(k)} + z_i^{(k-1)})$ $z_i^{(k)} = u_{i-1}^{(k)} + z_i^{(k-1)} - u_i^{(k)}$	$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} \frac{1}{2} \left\ y - \sum_{j < i} \Phi_j w_j^{(k)} - \sum_{j > i} \Phi_j w_j^{(k-1)} - \Phi_i w_i \right\ _2^2 + h_i(w_i),$

Key connection: for $S \subseteq \mathbb{R}^q$, and $h_S(x) = \max_{s \in S} \langle s, x \rangle$, we have

$$(\text{Id} - P_S)(r) = \arg \min_{x \in \mathbb{R}^q} \frac{1}{2} \|r - x\|_2^2 + h_S(x)$$

Proof sketch:

Equivalence of Dykstra and CD (cont.)

Dykstra, iteration i	Coordinate descent, iteration i
$u_i^{(k)} = P_{C_i}(u_{i-1}^{(k)} + z_i^{(k-1)})$ $z_i^{(k)} = u_{i-1}^{(k)} + z_i^{(k-1)} - u_i^{(k)}$	$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} \frac{1}{2} \left\ y - \sum_{j < i} \Phi_j w_j^{(k)} - \sum_{j > i} \Phi_j w_j^{(k-1)} - \Phi_i w_i \right\ _2^2 + h_i(w_i),$

Key connection: for $S \subseteq \mathbb{R}^q$, and $h_S(x) = \max_{s \in S} \langle s, x \rangle$, we have

$$(\text{Id} - P_S)(r) = \arg \min_{x \in \mathbb{R}^q} \frac{1}{2} \|r - x\|_2^2 + h_S(x)$$

Proof sketch:

- Rewrite $z_i^{(k)} = (\text{Id} - P_{C_i})(u_{i-1}^{(k)} + z_i^{(k-1)})$

Equivalence of Dykstra and CD (cont.)

Dykstra, iteration i	Coordinate descent, iteration i
$u_i^{(k)} = P_{C_i}(u_{i-1}^{(k)} + z_i^{(k-1)})$ $z_i^{(k)} = u_{i-1}^{(k)} + z_i^{(k-1)} - u_i^{(k)}$	$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} \frac{1}{2} \left\ y - \sum_{j < i} \Phi_j w_j^{(k)} - \sum_{j > i} \Phi_j w_j^{(k-1)} - \Phi_i w_i \right\ _2^2 + h_i(w_i),$

Key connection: for $S \subseteq \mathbb{R}^q$, and $h_S(x) = \max_{s \in S} \langle s, x \rangle$, we have

$$(\text{Id} - P_S)(r) = \arg \min_{x \in \mathbb{R}^q} \frac{1}{2} \|r - x\|_2^2 + h_S(x)$$

Proof sketch:

- Rewrite $z_i^{(k)} = (\text{Id} - P_{C_i})(u_{i-1}^{(k)} + z_i^{(k-1)})$
- By induction, $u_{i-1}^{(k)} + z_i^{(k-1)}$ is i th partial residual

Equivalence of Dykstra and CD (cont.)

Dykstra, iteration i	Coordinate descent, iteration i
$u_i^{(k)} = P_{C_i}(u_{i-1}^{(k)} + z_i^{(k-1)})$ $z_i^{(k)} = u_{i-1}^{(k)} + z_i^{(k-1)} - u_i^{(k)}$	$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} \frac{1}{2} \left\ y - \sum_{j < i} \Phi_j w_j^{(k)} - \sum_{j > i} \Phi_j w_j^{(k-1)} - \Phi_i w_i \right\ _2^2 + h_i(w_i),$

Key connection: for $S \subseteq \mathbb{R}^q$, and $h_S(x) = \max_{s \in S} \langle s, x \rangle$, we have

$$(\text{Id} - P_S)(r) = \arg \min_{x \in \mathbb{R}^q} \frac{1}{2} \|r - x\|_2^2 + h_S(x)$$

Proof sketch:

- Rewrite $z_i^{(k)} = (\text{Id} - P_{C_i})(u_{i-1}^{(k)} + z_i^{(k-1)})$
- By induction, $u_{i-1}^{(k)} + z_i^{(k-1)}$ is i th partial residual
- Therefore by key fact, it follows $z_i^{(k)} = \Phi_i w_i^{(k)}$

Equivalence of Dykstra and CD (cont.)

Short history:

- Dates back to Han (1988); Gaffke and Mathar (1989), for the case $\Phi = I$

Equivalence of Dykstra and CD (cont.)

Short history:

- Dates back to Han (1988); Gaffke and Mathar (1989), for the case $\Phi = I$
- Han was presumably unaware of Dykstra's work, and reinvented Dykstra's algorithm

Equivalence of Dykstra and CD (cont.)

Short history:

- Dates back to Han (1988); Gaffke and Mathar (1989), for the case $\Phi = I$
- Han was presumably unaware of Dykstra's work, and reinvented Dykstra's algorithm
- Tseng was also presumably unaware of Dykstra's work!

Equivalence of Dykstra and CD (cont.)

Short history:

- Dates back to Han (1988); Gaffke and Mathar (1989), for the case $\Phi = I$
- Han was presumably unaware of Dykstra's work, and reinvented Dykstra's algorithm
- Tseng was also presumably unaware of Dykstra's work! He was inspired by Han's work, and wrote 1993 paper accordingly.

Equivalence of Dykstra and CD (cont.)

Short history:

- Dates back to Han (1988); Gaffke and Mathar (1989), for the case $\Phi = I$
- Han was presumably unaware of Dykstra's work, and reinvented Dykstra's algorithm
- Tseng was also presumably unaware of Dykstra's work! He was inspired by Han's work, and wrote 1993 paper accordingly. Led eventually to his seminal 2001 paper

Alternating direction method of multipliers

For convex $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $g : \mathbb{R}^m \rightarrow \mathbb{R}$, matrices $A \in \mathbb{R}^{p \times n}$, $B \in \mathbb{R}^{p \times m}$, and $c \in \mathbb{R}^p$, consider

$$\min_{x \in \mathbb{R}^n, y \in \mathbb{R}^m} f(x) + g(y) \quad \text{s.t.} \quad Ax + By = c$$

Alternating direction method of multipliers

For convex $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $g : \mathbb{R}^m \rightarrow \mathbb{R}$, matrices $A \in \mathbb{R}^{p \times n}$, $B \in \mathbb{R}^{p \times m}$, and $c \in \mathbb{R}^p$, consider

$$\min_{x \in \mathbb{R}^n, y \in \mathbb{R}^m} f(x) + g(y) \quad \text{s.t.} \quad Ax + By = c$$

Define augmented Lagrangian: $L(x, y, v) = f(x) + g(y) + \frac{\rho}{2} \|Ax + By - c + v\|_2^2 - \frac{\rho}{2} \|v\|_2^2$

Alternating direction method of multipliers

For convex $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $g : \mathbb{R}^m \rightarrow \mathbb{R}$, matrices $A \in \mathbb{R}^{p \times n}$, $B \in \mathbb{R}^{p \times m}$, and $c \in \mathbb{R}^p$, consider

$$\min_{x \in \mathbb{R}^n, y \in \mathbb{R}^m} f(x) + g(y) \quad \text{s.t.} \quad Ax + By = c$$

Define augmented Lagrangian: $L(x, y, v) = f(x) + g(y) + \frac{\rho}{2} \|Ax + By - c + v\|_2^2 - \frac{\rho}{2} \|v\|_2^2$

Alternating direction method of multipliers or ADMM (Glowinski and Marroco, 1975; Gabay and Mercier, 1976):

Alternating direction method of multipliers

For convex $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $g : \mathbb{R}^m \rightarrow \mathbb{R}$, matrices $A \in \mathbb{R}^{p \times n}$, $B \in \mathbb{R}^{p \times m}$, and $c \in \mathbb{R}^p$, consider

$$\min_{x \in \mathbb{R}^n, y \in \mathbb{R}^m} f(x) + g(y) \quad \text{s.t.} \quad Ax + By = c$$

Define augmented Lagrangian: $L(x, y, v) = f(x) + g(y) + \frac{\rho}{2} \|Ax + By - c + v\|_2^2 - \frac{\rho}{2} \|v\|_2^2$

Alternating direction method of multipliers or **ADMM** (Glowinski and Marroco, 1975; Gabay and Mercier, 1976):



Stephen Boyd,
Stanford

Alternating direction method of multipliers

For convex $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $g : \mathbb{R}^m \rightarrow \mathbb{R}$, matrices $A \in \mathbb{R}^{p \times n}$, $B \in \mathbb{R}^{p \times m}$, and $c \in \mathbb{R}^p$, consider

$$\min_{x \in \mathbb{R}^n, y \in \mathbb{R}^m} f(x) + g(y) \quad \text{s.t.} \quad Ax + By = c$$

Define augmented Lagrangian: $L(x, y, v) = f(x) + g(y) + \frac{\rho}{2} \|Ax + By - c + v\|_2^2 - \frac{\rho}{2} \|v\|_2^2$

Alternating direction method of multipliers or **ADMM** (Glowinski and Marroco, 1975; Gabay and Mercier, 1976): initialize $y^{(0)}$, $v^{(0)}$, repeat

$$x^{(k)} = \arg \min_{x \in \mathbb{R}^n} L(x, y^{(k-1)}, v^{(k-1)})$$

$$y^{(k)} = \arg \min_{y \in \mathbb{R}^m} L(x^{(k)}, y, v^{(k-1)})$$

$$v^{(k)} = v^{(k-1)} + Ax^{(k)} + By^{(k)} - c$$



Stephen Boyd,
Stanford

Alternating direction method of multipliers (cont.)

Some notes:

- An operator splitting technique; equivalent to **Douglas-Rachford algorithm** (Douglas and Rachford, 1956) via duality argument

Alternating direction method of multipliers (cont.)

Some notes:

- An operator splitting technique; equivalent to **Douglas-Rachford algorithm** (Douglas and Rachford, 1956) via duality argument
- Early theory by Gabay, Eckstein, Bertsekas

Alternating direction method of multipliers (cont.)

Some notes:

- An operator splitting technique; equivalent to **Douglas-Rachford algorithm** (Douglas and Rachford, 1956) via duality argument
- Early theory by Gabay, Eckstein, Bertsekas
- Like CD, it has gained immense popularity recently in machine learning and statistics, sparked by Boyd et al. (2011)

Alternating direction method of multipliers (cont.)

Some notes:

- An operator splitting technique; equivalent to **Douglas-Rachford algorithm** (Douglas and Rachford, 1956) via duality argument
- Early theory by Gabay, Eckstein, Bertsekas
- Like CD, it has gained immense popularity recently in machine learning and statistics, sparked by Boyd et al. (2011)
- Lots of interesting theory and extensions still being developed

Equivalence of Dykstra and ADMM

Short history:

- Boyd et al. (2011): ADMM for 2-set intersection is equivalent to Dykstra's algorithm

Equivalence of Dykstra and ADMM

Short history:

- Boyd et al. (2011): ADMM for 2-set intersection is equivalent to Dykstra's algorithm
- Bauschke and Koch (2013): no it isn't

Equivalence of Dykstra and ADMM

Short history:

- Boyd et al. (2011): ADMM for 2-set intersection is equivalent to Dykstra's algorithm
- Bauschke and Koch (2013): no it isn't

Consider best approximation problem with $d = 2$ sets, rewritten as:

$$\min_{u_1 \in \mathbb{R}^n, u_2 \in \mathbb{R}^n} \|y - u_1\|_2^2 + I_{C_1}(u_1) + I_{C_2}(u_2) \quad \text{s.t.} \quad u_1 = u_2$$

Equivalence of Dykstra and ADMM

Short history:

- Boyd et al. (2011): ADMM for 2-set intersection is equivalent to Dykstra's algorithm
- Bauschke and Koch (2013): no it isn't

Consider best approximation problem with $d = 2$ sets, rewritten as:

$$\min_{u_1 \in \mathbb{R}^n, u_2 \in \mathbb{R}^n} \|y - u_1\|_2^2 + I_{C_1}(u_1) + I_{C_2}(u_2) \quad \text{s.t.} \quad u_1 = u_2$$

ADMM iterations:

$$\begin{aligned}u_1^{(k)} &= P_{C_1} \left(\frac{y}{1 + \rho} + \frac{\rho(u_2^{(k-1)} - z^{(k-1)})}{1 + \rho} \right) \\u_2^{(k)} &= P_{C_2}(u_1^{(k)} + z^{(k-1)}) \\z^{(k)} &= z^{(k-1)} + u_1^{(k)} - u_2^{(k)}\end{aligned}$$

Equivalence of Dykstra and ADMM (cont.)

When C_1 is a **linear subspace** and $\rho = 1$, easy inductive proof shows ADMM iterations are:

$$u_1^{(k)} = P_{C_1}(u_2^{(k-1)})$$

$$u_2^{(k)} = P_{C_2}(u_1^{(k)} + z^{(k-1)})$$

$$z^{(k)} = z^{(k-1)} + u_1^{(k)} - u_2^{(k)}$$

Equivalence of Dykstra and ADMM (cont.)

When C_1 is a **linear subspace** and $\rho = 1$, easy inductive proof shows ADMM iterations are:

$$\begin{aligned}u_1^{(k)} &= P_{C_1}(u_2^{(k-1)}) \\u_2^{(k)} &= P_{C_2}(u_1^{(k)} + z^{(k-1)}) \\z^{(k)} &= z^{(k-1)} + u_1^{(k)} - u_2^{(k)}\end{aligned}$$

which is precisely **equivalent** to Dykstra's algorithm.

Equivalence of Dykstra and ADMM (cont.)

When C_1 is a **linear subspace** and $\rho = 1$, easy inductive proof shows ADMM iterations are:

$$\begin{aligned}u_1^{(k)} &= P_{C_1}(u_2^{(k-1)}) \\u_2^{(k)} &= P_{C_2}(u_1^{(k)} + z^{(k-1)}) \\z^{(k)} &= z^{(k-1)} + u_1^{(k)} - u_2^{(k)}\end{aligned}$$

which is precisely **equivalent** to Dykstra's algorithm. The case $d = 2$ with one set being a linear subspace is actually pretty important!

Equivalence of Dykstra and ADMM (cont.)

When C_1 is a **linear subspace** and $\rho = 1$, easy inductive proof shows ADMM iterations are:

$$\begin{aligned}u_1^{(k)} &= P_{C_1}(u_2^{(k-1)}) \\u_2^{(k)} &= P_{C_2}(u_1^{(k)} + z^{(k-1)}) \\z^{(k)} &= z^{(k-1)} + u_1^{(k)} - u_2^{(k)}\end{aligned}$$

which is precisely **equivalent** to Dykstra's algorithm. The case $d = 2$ with one set being a linear subspace is actually pretty important!

(Even for general d , and no constraints on $C_1, \dots, C_d$, we can view Dykstra's algorithm as a limiting case of “inertial” ADMM, under a particular scaling for $\rho_1, \dots, \rho_d$)

Coordinate descent for the lasso

The lasso

The **lasso** problem (Tibshirani, 1996; Chen et al., 1998):

$$\min_{w \in \mathbb{R}^p} \frac{1}{2} \|y - \Phi w\|_2^2 + \lambda \sum_{i=1}^p |w_i|$$

The lasso

The **lasso** problem (Tibshirani, 1996; Chen et al., 1998):

$$\min_{w \in \mathbb{R}^p} \frac{1}{2} \|y - \Phi w\|_2^2 + \lambda \sum_{i=1}^p |w_i|$$

i.e., $h_i(w_i) = \max_{d \in [-\lambda, \lambda]} dw_i$, for $i = 1, \dots, p$.

The lasso

The **lasso** problem (Tibshirani, 1996; Chen et al., 1998):

$$\min_{w \in \mathbb{R}^p} \frac{1}{2} \|y - \Phi w\|_2^2 + \lambda \sum_{i=1}^p |w_i|$$

i.e., $h_i(w_i) = \max_{d \in [-\lambda, \lambda]} dw_i$, for $i = 1, \dots, p$. Dual problem:

$$\min_{u \in \mathbb{R}^n} \|y - u\|_2^2 \quad \text{s.t.} \quad u \in \bigcap_{i=1}^p \{v \in \mathbb{R}^n : \Phi_i^T v \in [-\lambda, \lambda]\}$$

The lasso

The **lasso** problem (Tibshirani, 1996; Chen et al., 1998):

$$\min_{w \in \mathbb{R}^p} \frac{1}{2} \|y - \Phi w\|_2^2 + \lambda \sum_{i=1}^p |w_i|$$

i.e., $h_i(w_i) = \max_{d \in [-\lambda, \lambda]} dw_i$, for $i = 1, \dots, p$. Dual problem:

$$\min_{u \in \mathbb{R}^n} \|y - u\|_2^2 \quad \text{s.t.} \quad u \in \bigcap_{i=1}^p \{v \in \mathbb{R}^n : \Phi_i^T v \in [-\lambda, \lambda]\}$$

Coordinate descent for lasso (Friedman et al., 2007; many others):

$$w_i^{(k)} = S_{\lambda / \|\Phi_i\|_2^2} \left(\frac{\Phi_i^T (y - \sum_{j < i} \Phi_j w_j^{(k)} - \sum_{j > i} \Phi_j w_j^{(k-1)})}{\|\Phi_i\|_2^2} \right),$$

$i = 1, \dots, p$

The lasso

The **lasso** problem (Tibshirani, 1996; Chen et al., 1998):

$$\min_{w \in \mathbb{R}^p} \frac{1}{2} \|y - \Phi w\|_2^2 + \lambda \sum_{i=1}^p |w_i|$$

i.e., $h_i(w_i) = \max_{d \in [-\lambda, \lambda]} dw_i$, for $i = 1, \dots, p$. Dual problem:

$$\min_{u \in \mathbb{R}^n} \|y - u\|_2^2 \quad \text{s.t.} \quad u \in \bigcap_{i=1}^p \{v \in \mathbb{R}^n : \Phi_i^T v \in [-\lambda, \lambda]\}$$

Coordinate descent for lasso (Friedman et al., 2007; many others):

$$w_i^{(k)} = S_{\lambda/\|\Phi_i\|_2^2} \left(\frac{\Phi_i^T (y - \sum_{j < i} \Phi_j w_j^{(k)} - \sum_{j > i} \Phi_j w_j^{(k-1)})}{\|\Phi_i\|_2^2} \right),$$

$i = 1, \dots, p$

Equivalent to Dykstra's (Hildreth's) algorithm on the dual!

Convergence rates

Implications of this equivalence are interesting from both sides.

Convergence rates

Implications of this equivalence are interesting from both sides. E.g., convergence results from Dykstra/Hildreth carry over to CD

Convergence rates

Implications of this equivalence are interesting from both sides. E.g., convergence results from Dykstra/Hildreth carry over to CD

Theorem (Adaptation of Iusem and De Pierro, 1990). Assume Φ is in general position. CD for the lasso has an **asymptotically linear** convergence rate, i.e., for large enough k ,

$$\frac{\|w^{(k+1)} - \hat{w}\|_{\Sigma}}{\|w^{(k)} - \hat{w}\|_{\Sigma}} \leq \left(\frac{a^2}{a^2 + \lambda_{\min}(\Phi_A^T \Phi_A) / \max_{i \in A} \|\Phi_i\|_2^2} \right)^{1/2}$$

where $\Sigma = \Phi^T \Phi$, $\|x\|_{\Sigma}^2 = x^T \Sigma x$ for $x \in \mathbb{R}^p$, $A = \text{supp}(\hat{w})$ is the active set of $\hat{w}$, and $a = |A|$ is its size

Convergence rates (cont.)

Theorem (Adaptation of Deutsch and Hundal, 1994). Assume Φ is in general position. CD for the lasso has an **asymptotically linear** convergence rate, i.e., for large enough k ,

$$\frac{\|w^{(k+1)} - \hat{w}\|_{\Sigma}}{\|w^{(k)} - \hat{w}\|_{\Sigma}} \leq \left(1 - \prod_{j=1}^{a-1} \frac{\|P_{\{i_{j+1}, \dots, i_a\}}^{\perp} \Phi_{i_j}\|_2^2}{\|\Phi_{i_j}\|_2^2} \right)^{1/2}$$

where $A = \{i_1, \dots, i_a\}$, $i_1 < \dots < i_a$, and $P_{\{i_{j+1}, \dots, i_a\}}^{\perp}$ is projection onto orthocomplement of span of $\Phi_{\{i_{j+1}, \dots, i_a\}}$

Convergence rates (cont.)

Theorem (Adaptation of Deutsch and Hundal, 1994). Assume Φ is in general position. CD for the lasso has an **asymptotically linear** convergence rate, i.e., for large enough k ,

$$\frac{\|w^{(k+1)} - \hat{w}\|_{\Sigma}}{\|w^{(k)} - \hat{w}\|_{\Sigma}} \leq \left(1 - \prod_{j=1}^{a-1} \frac{\|P_{\{i_{j+1}, \dots, i_a\}}^{\perp} \Phi_{i_j}\|_2^2}{\|\Phi_{i_j}\|_2^2} \right)^{1/2}$$

where $A = \{i_1, \dots, i_a\}$, $i_1 < \dots < i_a$, and $P_{\{i_{j+1}, \dots, i_a\}}^{\perp}$ is projection onto orthocomplement of span of $\Phi_{\{i_{j+1}, \dots, i_a\}}$

This bound is typically tighter than that from the previous theorem.

Convergence rates (cont.)

Theorem (Adaptation of Deutsch and Hundal, 1994). Assume Φ is in general position. CD for the lasso has an **asymptotically linear** convergence rate, i.e., for large enough k ,

$$\frac{\|w^{(k+1)} - \hat{w}\|_{\Sigma}}{\|w^{(k)} - \hat{w}\|_{\Sigma}} \leq \left(1 - \prod_{j=1}^{a-1} \frac{\|P_{\{i_{j+1}, \dots, i_a\}}^{\perp} \Phi_{i_j}\|_2^2}{\|\Phi_{i_j}\|_2^2}\right)^{1/2}$$

where $A = \{i_1, \dots, i_a\}$, $i_1 < \dots < i_a$, and $P_{\{i_{j+1}, \dots, i_a\}}^{\perp}$ is projection onto orthocomplement of span of $\Phi_{\{i_{j+1}, \dots, i_a\}}$

This bound is typically tighter than that from the previous theorem. E.g., for orthogonal Φ , this bound is zero, whereas the previous one is $\sqrt{a^2/(a^2 + 1)}$

Convergence rates (cont.)

Some remarks:

- Both bounds degrade for higher correlations between columns of Φ

Convergence rates (cont.)

Some remarks:

- Both bounds degrade for higher correlations between columns of Φ
- The second bound shows that **order matters!**

Convergence rates (cont.)

Some remarks:

- Both bounds degrade for higher correlations between columns of Φ
- The second bound shows that **order matters!**
- (Asymptotically) linear convergence without strong convexity ...

Convergence rates (cont.)

Some remarks:

- Both bounds degrade for higher correlations between columns of Φ
- The second bound shows that **order matters!**
- (Asymptotically) linear convergence without strong convexity ... not true of modern finite-time analyses of CD

Convergence rates (cont.)

Some remarks:

- Both bounds degrade for higher correlations between columns of Φ
- The second bound shows that **order matters!**
- (Asymptotically) linear convergence without strong convexity ... not true of modern finite-time analyses of CD
- Asymptotics kick in when CD identifies active set ...

Convergence rates (cont.)

Some remarks:

- Both bounds degrade for higher correlations between columns of Φ
- The second bound shows that **order matters!**
- (Asymptotically) linear convergence without strong convexity ... not true of modern finite-time analyses of CD
- Asymptotics kick in when CD identifies active set ... evidence for the advantage of **warm starts?**

Parallel coordinate descent

Parallel Dykstra's algorithm

A **parallel version** of Dykstra's algorithm (Iusem and Pierro, 1987; Gaffke and Mathar; 1989) simply follows from product space trick,

Parallel Dykstra's algorithm

A **parallel version** of Dykstra's algorithm (Iusem and Pierro, 1987; Gaffke and Mathar, 1989) simply follows from product space trick, i.e., rewrite d -set best approximation problem as

$$\min_{u=(u_1,\dots,u_d)\in\mathbb{R}^{nd}} \sum_{i=1}^d \gamma_i \|y - u_i\|_2^2 \quad \text{s.t.} \quad u \in C_0 \cap (C_1 \times \dots \times C_d)$$

where $C_0 = \{(u_1, \dots, u_d) \in \mathbb{R}^{nd} : u_1 = \dots = u_d\}$, $\gamma_1, \dots, \gamma_d > 0$ are weights with $\sum_{i=1}^d \gamma_i = 1$.

Parallel Dykstra's algorithm

A **parallel version** of Dykstra's algorithm (Iusem and Pierro, 1987; Gaffke and Mathar, 1989) simply follows from product space trick, i.e., rewrite d -set best approximation problem as

$$\min_{u=(u_1,\dots,u_d)\in\mathbb{R}^{nd}} \sum_{i=1}^d \gamma_i \|y - u_i\|_2^2 \quad \text{s.t.} \quad u \in C_0 \cap (C_1 \times \dots \times C_d)$$

where $C_0 = \{(u_1, \dots, u_d) \in \mathbb{R}^{nd} : u_1 = \dots = u_d\}$, $\gamma_1, \dots, \gamma_d > 0$ are weights with $\sum_{i=1}^d \gamma_i = 1$. Applying Dykstra to the above:

$$\left. \begin{aligned} u_0^{(k)} &= \sum_{i=1}^d \gamma_i u_i^{(k-1)} \\ u_i^{(k)} &= P_{C_i}(u_0^{(k)} + z_i^{(k-1)}) \\ z_i^{(k)} &= u_0^{(k)} + z_i^{(k-1)} - u_i^{(k)} \end{aligned} \right\} \quad i = 1, \dots, d$$

Parallel-Dykstra-CD

Passing parallel Dykstra's algorithm through the connection to CD gives what we call **parallel-Dykstra-CD**:

$$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} \frac{1}{2} \left\| y - \Phi w^{(k-1)} + \frac{\Phi_i w_i^{(k-1)}}{\gamma_i} - \frac{\Phi_i w_i}{\gamma_i} \right\|_2^2 + h_i \left(\frac{w_i}{\gamma_i} \right),$$

$i = 1, \dots, d$

Parallel-Dykstra-CD

Passing parallel Dykstra's algorithm through the connection to CD gives what we call **parallel-Dykstra-CD**:

$$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} \frac{1}{2} \left\| y - \Phi w^{(k-1)} + \frac{\Phi_i w_i^{(k-1)}}{\gamma_i} - \frac{\Phi_i w_i}{\gamma_i} \right\|_2^2 + h_i \left(\frac{w_i}{\gamma_i} \right),$$

$i = 1, \dots, d$

Some remarks:

Parallel-Dykstra-CD

Passing parallel Dykstra's algorithm through the connection to CD gives what we call **parallel-Dykstra-CD**:

$$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} \frac{1}{2} \left\| y - \Phi w^{(k-1)} + \frac{\Phi_i w_i^{(k-1)}}{\gamma_i} - \frac{\Phi_i w_i}{\gamma_i} \right\|_2^2 + h_i \left(\frac{w_i}{\gamma_i} \right),$$

$i = 1, \dots, d$

Some remarks:

- When $\gamma_i = 1$, the i th update is the full “Jacobi” parallelization
...

Parallel-Dykstra-CD

Passing parallel Dykstra's algorithm through the connection to CD gives what we call **parallel-Dykstra-CD**:

$$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} \frac{1}{2} \left\| y - \Phi w^{(k-1)} + \frac{\Phi_i w_i^{(k-1)}}{\gamma_i} - \frac{\Phi_i w_i}{\gamma_i} \right\|_2^2 + h_i \left(\frac{w_i}{\gamma_i} \right),$$

$i = 1, \dots, d$

Some remarks:

- When $\gamma_i = 1$, the i th update is the full “Jacobi” parallelization ... but recall we must constrain $\sum_{i=1}^d \gamma_i = 1$!

Parallel-Dykstra-CD

Passing parallel Dykstra's algorithm through the connection to CD gives what we call **parallel-Dykstra-CD**:

$$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} \frac{1}{2} \left\| y - \Phi w^{(k-1)} + \frac{\Phi_i w_i^{(k-1)}}{\gamma_i} - \frac{\Phi_i w_i}{\gamma_i} \right\|_2^2 + h_i \left(\frac{w_i}{\gamma_i} \right),$$

$i = 1, \dots, d$

Some remarks:

- When $\gamma_i = 1$, the i th update is the full “Jacobi” parallelization ... but recall we must constrain $\sum_{i=1}^d \gamma_i = 1$!
- Interpret it as a kind of **weighted averaging** of d Jacobi updates

Parallel-Dykstra-CD

Passing parallel Dykstra's algorithm through the connection to CD gives what we call **parallel-Dykstra-CD**:

$$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} \frac{1}{2} \left\| y - \Phi w^{(k-1)} + \frac{\Phi_i w_i^{(k-1)}}{\gamma_i} - \frac{\Phi_i w_i}{\gamma_i} \right\|_2^2 + h_i \left(\frac{w_i}{\gamma_i} \right),$$

$i = 1, \dots, d$

Some remarks:

- When $\gamma_i = 1$, the i th update is the full “Jacobi” parallelization ... but recall we must constrain $\sum_{i=1}^d \gamma_i = 1$!
- Interpret it as a kind of **weighted averaging** of d Jacobi updates
- Converges under no assumptions

Parallel-Dykstra-CD

Passing parallel Dykstra's algorithm through the connection to CD gives what we call **parallel-Dykstra-CD**:

$$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} \frac{1}{2} \left\| y - \Phi w^{(k-1)} + \frac{\Phi_i w_i^{(k-1)}}{\gamma_i} - \frac{\Phi_i w_i}{\gamma_i} \right\|_2^2 + h_i \left(\frac{w_i}{\gamma_i} \right),$$

$i = 1, \dots, d$

Some remarks:

- When $\gamma_i = 1$, the i th update is the full “Jacobi” parallelization ... but recall we must constrain $\sum_{i=1}^d \gamma_i = 1$!
- Interpret it as a kind of **weighted averaging** of d Jacobi updates
- Converges under no assumptions
- For the lasso problem, parallel-Dykstra-CD also has asymptotic linear convergence (adapted from Iusem and De Pierro, 1990)

Parallel-ADMM-CD

Consider ADMM for the 2-set problem:

$$\min_{u=(u_1,\dots,u_d)\in\mathbb{R}^{nd}} \sum_{i=1}^d \gamma_i \|y - u_i\|_2^2 \quad \text{s.t.} \quad u \in C_0 \cap (C_1 \times \dots \times C_d)$$

Parallel-ADMM-CD

Consider ADMM for the 2-set problem:

$$\min_{u=(u_1,\dots,u_d)\in\mathbb{R}^{nd}} \sum_{i=1}^d \gamma_i \|y - u_i\|_2^2 \quad \text{s.t.} \quad u \in C_0 \cap (C_1 \times \dots \times C_d)$$

Passing steps through the same connection (between projection and penalized minimization) gives what we call **parallel-ADMM-CD**:

$$u_0^{(k)} = \frac{y - \Phi w^{(k-1)}}{1 + \sum_{i=1}^d \rho_i} + \frac{(\sum_{i=1}^d \rho_i) u_0^{(k-1)}}{1 + \sum_{i=1}^d \rho_i} + \frac{\Phi(w^{(k-2)} - w^{(k-1)})}{1 + \sum_{i=1}^d \rho_i}$$
$$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} \frac{1}{2} \left\| u_0^{(k)} + \frac{\Phi_i w_i^{(k-1)}}{\rho_i} - \frac{\Phi_i w_i}{\rho_i} \right\|_2^2 + h_i \left(\frac{w_i}{\rho_i} \right),$$
$$i = 1, \dots, d$$

Parallel-ADMM-CD

Consider ADMM for the 2-set problem:

$$\min_{u=(u_1,\dots,u_d)\in\mathbb{R}^{nd}} \sum_{i=1}^d \gamma_i \|y - u_i\|_2^2 \quad \text{s.t.} \quad u \in C_0 \cap (C_1 \times \dots \times C_d)$$

Passing steps through the same connection (between projection and penalized minimization) gives what we call **parallel-ADMM-CD**:

$$\begin{aligned} u_0^{(k)} &= \frac{y - \Phi w^{(k-1)}}{1 + \sum_{i=1}^d \rho_i} + \frac{(\sum_{i=1}^d \rho_i) u_0^{(k-1)}}{1 + \sum_{i=1}^d \rho_i} + \frac{\Phi(w^{(k-2)} - w^{(k-1)})}{1 + \sum_{i=1}^d \rho_i} \\ w_i^{(k)} &= \arg \min_{w_i \in \mathbb{R}^{p_i}} \frac{1}{2} \left\| u_0^{(k)} + \frac{\Phi_i w_i^{(k-1)}}{\rho_i} - \frac{\Phi_i w_i}{\rho_i} \right\|_2^2 + h_i \left(\frac{w_i}{\rho_i} \right), \\ &\hspace{25em} i = 1, \dots, d \end{aligned}$$

Here $\rho_1, \dots, \rho_d > 0$ are arbitrary augmented Lagrangian parameters

Parallel-ADMM-CD (cont.)

Some remarks:

- Note $u_0^{(k)}$ is a convex combination of residual $y - \Phi w^{(k)}$ and momentum term

Parallel-ADMM-CD (cont.)

Some remarks:

- Note $u_0^{(k)}$ is a convex combination of residual $y - \Phi w^{(k)}$ and momentum term
- Converges under no assumptions

Parallel-ADMM-CD (cont.)

Some remarks:

- Note $u_0^{(k)}$ is a convex combination of residual $y - \Phi w^{(k)}$ and momentum term
- Converges under no assumptions
- When $\sum_{i=1}^d \rho_i = 1$, reduces to parallel-Dykstra-CD

Parallel-ADMM-CD (cont.)

Some remarks:

- Note $u_0^{(k)}$ is a convex combination of residual $y - \Phi w^{(k)}$ and momentum term
- Converges under no assumptions
- When $\sum_{i=1}^d \rho_i = 1$, **reduces to** parallel-Dykstra-CD ... which is just a restatement of equivalence ADMM and Dykstra for 2-set problem, with one a linear subspace

Parallel-ADMM-CD (cont.)

Some remarks:

- Note $u_0^{(k)}$ is a convex combination of residual $y - \Phi w^{(k)}$ and momentum term
- Converges under no assumptions
- When $\sum_{i=1}^d \rho_i = 1$, **reduces to** parallel-Dykstra-CD ... which is just a restatement of equivalence ADMM and Dykstra for 2-set problem, with one a linear subspace
- But no constraints on $\rho_1, \dots, \rho_d > 0$, so parallel-ADMM-CD is **strictly more general**

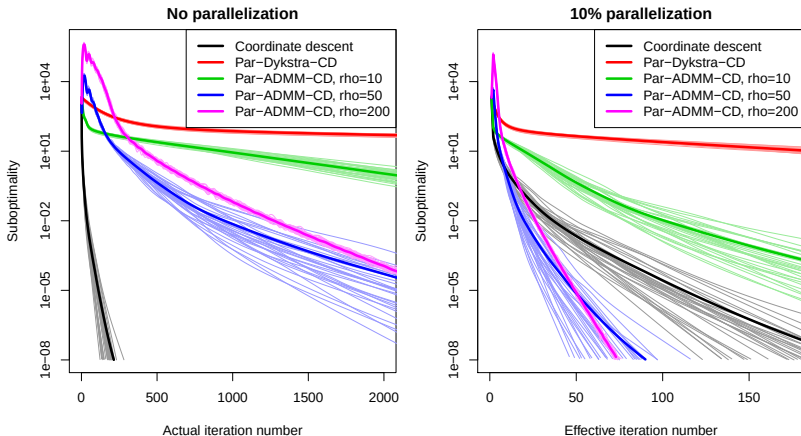
Parallel-ADMM-CD (cont.)

Some remarks:

- Note $u_0^{(k)}$ is a convex combination of residual $y - \Phi w^{(k)}$ and momentum term
- Converges under no assumptions
- When $\sum_{i=1}^d \rho_i = 1$, **reduces to** parallel-Dykstra-CD ... which is just a restatement of equivalence ADMM and Dykstra for 2-set problem, with one a linear subspace
- But no constraints on $\rho_1, \dots, \rho_d > 0$, so parallel-ADMM-CD is **strictly more general**
- Comparing parallel-ADMM-CD to parallel CD algorithms in the current literature: latter are all stochastic (instead of cyclic)

Lasso parallel CD example

Experimental setup: for $n = 200$ and $p = 500$, we aggregate results over 30 random instances of lasso problems



Extension to nonquadratic loss

Coordinate descent for general loss

Regularized estimation problem: given convex $f : \mathbb{R}^n \rightarrow \mathbb{R}$, solve

$$\min_{w \in \mathbb{R}^p} f(\Phi w) + \sum_{i=1}^d h_i(w_i)$$

Coordinate descent for general loss

Regularized estimation problem: given convex $f : \mathbb{R}^n \rightarrow \mathbb{R}$, solve

$$\min_{w \in \mathbb{R}^p} f(\Phi w) + \sum_{i=1}^d h_i(w_i)$$

Coordinate descent:

$$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} f \left(\sum_{j < i} \Phi_j w_j^{(k)} + \sum_{j > i} \Phi_j w_j^{(k-1)} + \Phi_i w_i \right) + h_i(w_i),$$
$$i = 1, \dots, d$$

Coordinate descent for general loss

Regularized estimation problem: given convex $f : \mathbb{R}^n \rightarrow \mathbb{R}$, solve

$$\min_{w \in \mathbb{R}^p} f(\Phi w) + \sum_{i=1}^d h_i(w_i)$$

Coordinate descent:

$$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} f \left(\sum_{j < i} \Phi_j w_j^{(k)} + \sum_{j > i} \Phi_j w_j^{(k-1)} + \Phi_i w_i \right) + h_i(w_i),$$
$$i = 1, \dots, d$$

Usually not computable in closed-form, so rarely used for general f

Coordinate descent for general loss

Regularized estimation problem: given convex $f : \mathbb{R}^n \rightarrow \mathbb{R}$, solve

$$\min_{w \in \mathbb{R}^p} f(\Phi w) + \sum_{i=1}^d h_i(w_i)$$

Coordinate descent:

$$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} f \left(\sum_{j < i} \Phi_j w_j^{(k)} + \sum_{j > i} \Phi_j w_j^{(k-1)} + \Phi_i w_i \right) + h_i(w_i),$$
$$i = 1, \dots, d$$

Usually not computable in closed-form, so rarely used for general f

(Common approach is to use proximal Newton, and then CD for the inner loop, where the loss is quadratic)

Dykstra's algorithm for Bregman projection

Best Bregman-approximation problem: given differentiable, strictly convex $g : \mathbb{R}^n \rightarrow \mathbb{R}$, solve

$$\min_{u \in \mathbb{R}^n} D_g(u, b) \quad \text{s.t.} \quad u \in C_1 \cap \cdots \cap C_d$$

where $D_g(u, b) = g(u) - g(b) - \langle \nabla g(b), u - b \rangle$ denotes **Bregman divergence** with respect to g .

Dykstra's algorithm for Bregman projection

Best Bregman-approximation problem: given differentiable, strictly convex $g : \mathbb{R}^n \rightarrow \mathbb{R}$, solve

$$\min_{u \in \mathbb{R}^n} D_g(u, b) \quad \text{s.t.} \quad u \in C_1 \cap \cdots \cap C_d$$

where $D_g(u, b) = g(u) - g(b) - \langle \nabla g(b), u - b \rangle$ denotes **Bregman divergence** with respect to g . **Dykstra's algorithm**:

$$\left. \begin{aligned} u_0^{(k)} &= u_d^{(k-1)} \\ u_i^{(k)} &= (P_{C_i}^g \circ \nabla g^*) \left(\nabla g(u_{i-1}^{(k)}) + z_i^{(k-1)} \right) \\ z_i^{(k)} &= \nabla g(u_{i-1}^{(k)}) + z_i^{(k-1)} - \nabla g(u_i^{(k)}) \end{aligned} \right\} \quad i = 1, \dots, d$$

where $P_C^g(x) = \arg \min_{c \in C} D_g(c, x)$ denotes Bregman projection, and g^* denotes the conjugate of g

General Dykstra-CD equivalence

Suppose as before that, for $i = 1, \dots, d$,

$$h_i(v) = \max_{d \in D_i} \langle d, v \rangle, \quad C_i = (\Phi_i^T)^{-1}(D_i) = \{v \in \mathbb{R}^n : \Phi_i^T v \in D_i\}$$

General Dykstra-CD equivalence

Suppose as before that, for $i = 1, \dots, d$,

$$h_i(v) = \max_{d \in D_i} \langle d, v \rangle, \quad C_i = (\Phi_i^T)^{-1}(D_i) = \{v \in \mathbb{R}^n : \Phi_i^T v \in D_i\}$$

and now also that $g(v) = f^*(-v)$, $b = -\nabla f(0)$

General Dykstra-CD equivalence

Suppose as before that, for $i = 1, \dots, d$,

$$h_i(v) = \max_{d \in D_i} \langle d, v \rangle, \quad C_i = (\Phi_i^T)^{-1}(D_i) = \{v \in \mathbb{R}^n : \Phi_i^T v \in D_i\}$$

and now also that $g(v) = f^*(-v)$, $b = -\nabla f(0)$

Then:

General Dykstra-CD equivalence

Suppose as before that, for $i = 1, \dots, d$,

$$h_i(v) = \max_{d \in D_i} \langle d, v \rangle, \quad C_i = (\Phi_i^T)^{-1}(D_i) = \{v \in \mathbb{R}^n : \Phi_i^T v \in D_i\}$$

and now also that $g(v) = f^*(-v)$, $b = -\nabla f(0)$

Then:

- Regularized estimation problem and best Bregman-approx are **duals**.

General Dykstra-CD equivalence

Suppose as before that, for $i = 1, \dots, d$,

$$h_i(v) = \max_{d \in D_i} \langle d, v \rangle, \quad C_i = (\Phi_i^T)^{-1}(D_i) = \{v \in \mathbb{R}^n : \Phi_i^T v \in D_i\}$$

and now also that $g(v) = f^*(-v)$, $b = -\nabla f(0)$

Then:

- Regularized estimation problem and best Bregman-approx are **duals**. Solutions $\hat{w}, \hat{u}$ related by $\hat{u} = -\nabla f(\Phi \hat{w})$

General Dykstra-CD equivalence

Suppose as before that, for $i = 1, \dots, d$,

$$h_i(v) = \max_{d \in D_i} \langle d, v \rangle, \quad C_i = (\Phi_i^T)^{-1}(D_i) = \{v \in \mathbb{R}^n : \Phi_i^T v \in D_i\}$$

and now also that $g(v) = f^*(-v)$, $b = -\nabla f(0)$

Then:

- Regularized estimation problem and best Bregman-approx are **duals**. Solutions $\hat{w}, \hat{u}$ related by $\hat{u} = -\nabla f(\Phi \hat{w})$
- Coordinate descent and Dykstra's algorithm are still **equivalent**,

General Dykstra-CD equivalence

Suppose as before that, for $i = 1, \dots, d$,

$$h_i(v) = \max_{d \in D_i} \langle d, v \rangle, \quad C_i = (\Phi_i^T)^{-1}(D_i) = \{v \in \mathbb{R}^n : \Phi_i^T v \in D_i\}$$

and now also that $g(v) = f^*(-v)$, $b = -\nabla f(0)$

Then:

- Regularized estimation problem and best Bregman-approx are **duals**. Solutions $\hat{w}, \hat{u}$ related by $\hat{u} = -\nabla f(\Phi \hat{w})$
- Coordinate descent and Dykstra's algorithm are still **equivalent**, in that at all iterations

$$z_i^{(k)} = \Phi_i w_i^{(k)}, \quad u_i^{(k)} = -\nabla f \left(\sum_{j \leq i} \Phi_j w_j^{(k)} + \sum_{j > i} \Phi_j w_j^{(k-1)} \right),$$

$i = 1, \dots, d$

General parallel CD algorithms

Use the product space trick to turn best Bregman-approx into 2-set problem:

$$\min_{u \in \mathbb{R}^{nd}} D_{\tilde{g}}(u, \tilde{b}) \quad \text{s.t.} \quad u \in C_0 \cap (C_1 \times \cdots \times C_d)$$

where $C_0 = \{(u_1, \dots, u_d) \in \mathbb{R}^{nd} : u_1 = \cdots = u_d\}$, $\gamma_1, \dots, \gamma_d > 0$ are weights with $\sum_{i=1}^d \gamma_i = 1$, as before,

General parallel CD algorithms

Use the product space trick to turn best Bregman-approx into 2-set problem:

$$\min_{u \in \mathbb{R}^{nd}} D_{\tilde{g}}(u, \tilde{b}) \quad \text{s.t.} \quad u \in C_0 \cap (C_1 \times \cdots \times C_d)$$

where $C_0 = \{(u_1, \dots, u_d) \in \mathbb{R}^{nd} : u_1 = \cdots = u_d\}$, $\gamma_1, \dots, \gamma_d > 0$ are weights with $\sum_{i=1}^d \gamma_i = 1$, as before, and now $\tilde{g}(u_1, \dots, u_d) = \sum_{i=1}^d \gamma_i g(u_i)$, $\tilde{b} = (b, \dots, b) \in \mathbb{R}^{nd}$

General parallel CD algorithms

Use the product space trick to turn best Bregman-approx into 2-set problem:

$$\min_{u \in \mathbb{R}^{nd}} D_{\tilde{g}}(u, \tilde{b}) \quad \text{s.t.} \quad u \in C_0 \cap (C_1 \times \cdots \times C_d)$$

where $C_0 = \{(u_1, \dots, u_d) \in \mathbb{R}^{nd} : u_1 = \cdots = u_d\}$, $\gamma_1, \dots, \gamma_d > 0$ are weights with $\sum_{i=1}^d \gamma_i = 1$, as before, and now $\tilde{g}(u_1, \dots, u_d) = \sum_{i=1}^d \gamma_i g(u_i)$, $\tilde{b} = (b, \dots, b) \in \mathbb{R}^{nd}$

Apply Dykstra's algorithm, reformulate, to give **parallel-Dykstra-CD**:

$$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} f \left(\Phi w^{(k)} - \frac{\Phi_i w_i^{(k)}}{\gamma_i} + \frac{\Phi_i w_i}{\gamma_i} \right) + h_i \left(\frac{w_i}{\gamma_i} \right),$$

$i = 1, \dots, d$

General parallel CD algorithms (cont.)

Or instead, apply ADMM, reformulate, to give **parallel-ADMM-CD**:

$$u_0^{(k)} = -\nabla f \left(\left(\sum_{i=1}^d \rho_i \right) (u_0^{(k)} - u_0^{(k-1)}) - \Phi(w^{(k-2)} - 2w^{(k-1)}) \right)$$
$$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} \frac{1}{2} \left\| u_0^{(k)} + \frac{\Phi_i w_i^{(k-1)}}{\rho_i} - \frac{\Phi_i w_i}{\rho_i} \right\|_2^2 + h_i \left(\frac{w_i}{\rho_i} \right),$$
$$i = 1, \dots, d$$

for augmented Lagrangian parameters $\rho_1, \dots, \rho_d > 0$

General parallel CD algorithms (cont.)

Or instead, apply ADMM, reformulate, to give **parallel-ADMM-CD**:

$$u_0^{(k)} = -\nabla f \left(\left(\sum_{i=1}^d \rho_i \right) (u_0^{(k)} - u_0^{(k-1)}) - \Phi(w^{(k-2)} - 2w^{(k-1)}) \right)$$
$$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} \frac{1}{2} \left\| u_0^{(k)} + \frac{\Phi_i w_i^{(k-1)}}{\rho_i} - \frac{\Phi_i w_i}{\rho_i} \right\|_2^2 + h_i \left(\frac{w_i}{\rho_i} \right),$$
$$i = 1, \dots, d$$

for augmented Lagrangian parameters $\rho_1, \dots, \rho_d > 0$

Some remarks:

General parallel CD algorithms (cont.)

Or instead, apply ADMM, reformulate, to give **parallel-ADMM-CD**:

$$u_0^{(k)} = -\nabla f \left(\left(\sum_{i=1}^d \rho_i \right) (u_0^{(k)} - u_0^{(k-1)}) - \Phi(w^{(k-2)} - 2w^{(k-1)}) \right)$$
$$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} \frac{1}{2} \left\| u_0^{(k)} + \frac{\Phi_i w_i^{(k-1)}}{\rho_i} - \frac{\Phi_i w_i}{\rho_i} \right\|_2^2 + h_i \left(\frac{w_i}{\rho_i} \right),$$
$$i = 1, \dots, d$$

for augmented Lagrangian parameters $\rho_1, \dots, \rho_d > 0$

Some remarks:

- Parallel-Dykstra-CD performs d f -minimizations (penalized) in a cycle, and parallel-ADMM-CD performs 1 (unpenalized)

General parallel CD algorithms (cont.)

Or instead, apply ADMM, reformulate, to give **parallel-ADMM-CD**:

$$u_0^{(k)} = -\nabla f \left(\left(\sum_{i=1}^d \rho_i \right) (u_0^{(k)} - u_0^{(k-1)}) - \Phi(w^{(k-2)} - 2w^{(k-1)}) \right)$$
$$w_i^{(k)} = \arg \min_{w_i \in \mathbb{R}^{p_i}} \frac{1}{2} \left\| u_0^{(k)} + \frac{\Phi_i w_i^{(k-1)}}{\rho_i} - \frac{\Phi_i w_i}{\rho_i} \right\|_2^2 + h_i \left(\frac{w_i}{\rho_i} \right),$$
$$i = 1, \dots, d$$

for augmented Lagrangian parameters $\rho_1, \dots, \rho_d > 0$

Some remarks:

- Parallel-Dykstra-CD performs d f -minimizations (penalized) in a cycle, and parallel-ADMM-CD performs 1 (unpenalized)
- These two are **not equivalent** (for $\sum_{i=1}^d \rho_i = 1$, or any config.)

Logistic lasso

As an example, consider the **logistic lasso** problem:

$$\min_{w \in \mathbb{R}^p} \sum_{i=1}^n \left(-y_i \phi_i^T w + \log(1 + \exp(\phi_i^T w)) \right) + \lambda \sum_{i=1}^p |w_i|$$

where $\phi_i \in \mathbb{R}^n$, $i = 1, \dots, n$ are rows of Φ .

Logistic lasso

As an example, consider the **logistic lasso** problem:

$$\min_{w \in \mathbb{R}^p} \sum_{i=1}^n \left(-y_i \phi_i^T w + \log(1 + \exp(\phi_i^T w)) \right) + \lambda \sum_{i=1}^p |w_i|$$

where $\phi_i \in \mathbb{R}^n$, $i = 1, \dots, n$ are rows of Φ . Parallel-ADMM-CD:

$$u_{0i}^{(k)} = y_i - \sigma(\rho u_{0i}^{(k)} - c_i^{(k)}), \quad i = 1, \dots, n$$

$$w_i^{(k)} = S_{\lambda \rho_i / \|\Phi_i\|_2^2} \left(\frac{\rho_i \Phi_i^T (u_0^{(k)} + \Phi_i w_i^{(k-1)}) / \rho_i}{\|\Phi_i\|_2^2} \right), \quad i = 1, \dots, p$$

where we let $\rho = \sum_{i=1}^p \rho_i$, $c_i^{(k)} = \rho u_{0i}^{(k-1)} + \phi_i^T (w^{(k-2)} - 2w^{(k-1)})$, $i = 1, \dots, n$, and $\sigma(t) = 1/(1 + e^{-t})$

Logistic lasso

As an example, consider the **logistic lasso** problem:

$$\min_{w \in \mathbb{R}^p} \sum_{i=1}^n \left(-y_i \phi_i^T w + \log(1 + \exp(\phi_i^T w)) \right) + \lambda \sum_{i=1}^p |w_i|$$

where $\phi_i \in \mathbb{R}^n$, $i = 1, \dots, n$ are rows of Φ . Parallel-ADMM-CD:

$$u_{0i}^{(k)} = y_i - \sigma(\rho u_{0i}^{(k)} - c_i^{(k)}), \quad i = 1, \dots, n$$

$$w_i^{(k)} = S_{\lambda \rho_i / \|\Phi_i\|_2^2} \left(\frac{\rho_i \Phi_i^T (u_0^{(k)} + \Phi_i w_i^{(k-1)}) / \rho_i}{\|\Phi_i\|_2^2} \right), \quad i = 1, \dots, p$$

where we let $\rho = \sum_{i=1}^p \rho_i$, $c_i^{(k)} = \rho u_{0i}^{(k-1)} + \phi_i^T (w^{(k-2)} - 2w^{(k-1)})$, $i = 1, \dots, n$, and $\sigma(t) = 1/(1 + e^{-t})$

Here, u_0 -update: n univariate minimizations (e.g., can use bisection search), w -update: p soft-thresholds.

Logistic lasso

As an example, consider the **logistic lasso** problem:

$$\min_{w \in \mathbb{R}^p} \sum_{i=1}^n \left(-y_i \phi_i^T w + \log(1 + \exp(\phi_i^T w)) \right) + \lambda \sum_{i=1}^p |w_i|$$

where $\phi_i \in \mathbb{R}^n$, $i = 1, \dots, n$ are rows of Φ . Parallel-ADMM-CD:

$$u_{0i}^{(k)} = y_i - \sigma(\rho u_{0i}^{(k)} - c_i^{(k)}), \quad i = 1, \dots, n$$

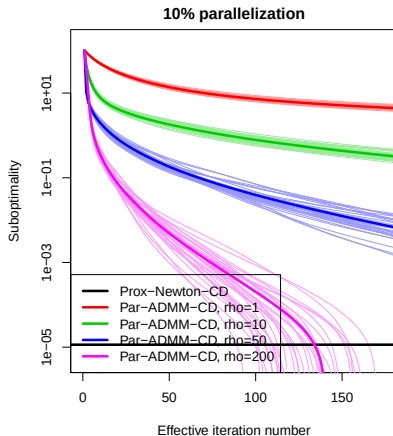
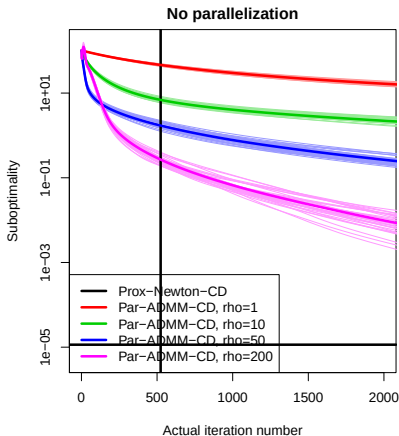
$$w_i^{(k)} = S_{\lambda \rho_i / \|\Phi_i\|_2^2} \left(\frac{\rho_i \Phi_i^T (u_0^{(k)} + \Phi_i w_i^{(k-1)}) / \rho_i}{\|\Phi_i\|_2^2} \right), \quad i = 1, \dots, p$$

where we let $\rho = \sum_{i=1}^p \rho_i$, $c_i^{(k)} = \rho u_{0i}^{(k-1)} + \phi_i^T (w^{(k-2)} - 2w^{(k-1)})$, $i = 1, \dots, n$, and $\sigma(t) = 1/(1 + e^{-t})$

Here, u_0 -update: n univariate minimizations (e.g., can use bisection search), w -update: p soft-thresholds. **Both are parallelizable!**

Logistic lasso parallel CD example

Experimental setup: for $n = 200$ and $p = 500$, we aggregate results over 30 random instances of logistic lasso problems



Back to additive models

Back to additive trend filtering

Recall **additive trend filtering**:

$$\text{Primal:} \quad \min_{\theta_1, \dots, \theta_d \in \mathbb{R}^n} \frac{1}{2} \|y - \sum_{j=1}^d \theta_j\|_2^2 + \lambda \sum_{j=1}^d \|\Delta_j^{(k+1)} \theta_j\|_1$$

$$\text{Dual:} \quad \min_{u \in \mathbb{R}^n} \|y - u\|_2^2 \quad \text{s.t.} \quad u \in C_1 \cap \dots \cap C_d$$

where $C_j = \{(\Delta_j^{(k+1)})^T v_j : \|v_j\|_\infty \leq \lambda\}$, $j = 1, \dots, d$

Back to additive trend filtering

Recall **additive trend filtering**:

$$\text{Primal: } \min_{\theta_1, \dots, \theta_d \in \mathbb{R}^n} \frac{1}{2} \|y - \sum_{j=1}^d \theta_j\|_2^2 + \lambda \sum_{j=1}^d \|\Delta_j^{(k+1)} \theta_j\|_1$$

$$\text{Dual: } \min_{u \in \mathbb{R}^n} \|y - u\|_2^2 \quad \text{s.t.} \quad u \in C_1 \cap \dots \cap C_d$$

where $C_j = \{(\Delta_j^{(k+1)})^T v_j : \|v_j\|_\infty \leq \lambda\}$, $j = 1, \dots, d$

Backfitting or block CD: center y , and repeat

$$\theta_j^{(t)} = \text{TF}_{k,\lambda} \left(y - \sum_{\ell < j} \theta_\ell^{(t)} - \sum_{\ell > j} \theta_\ell^{(t-1)}, X_j \right), \quad j = 1, \dots, d$$

where $\text{TF}_{k,\lambda}(z, x)$ is k th order univariate trend filtering with tuning parameter λ , response z , and inputs x

Parallel backfitting for trend filtering

Parallel-ADMM-backfitting: center y , and repeat

$$u_0^{(t)} = \frac{y - \sum_{j=1}^d \theta_j^{(t-1)}}{1 + \rho} + \frac{\rho u_0^{(t-1)}}{1 + \rho} + \frac{\sum_{j=1}^d (\theta_j^{(t-2)} - \theta_j^{(t-1)})}{1 + \rho}$$
$$\theta_j^{(t)} = \rho_j \cdot \text{TF}_{k,\lambda}(u_0^{(t)} + \theta_j^{(t-1)}/\rho_j, X_j), \quad j = 1, \dots, d$$

where we let $\rho = \sum_{j=1}^p \rho_j$

Parallel backfitting for trend filtering

Parallel-ADMM-backfitting: center y , and repeat

$$u_0^{(t)} = \frac{y - \sum_{j=1}^d \theta_j^{(t-1)}}{1 + \rho} + \frac{\rho u_0^{(t-1)}}{1 + \rho} + \frac{\sum_{j=1}^d (\theta_j^{(t-2)} - \theta_j^{(t-1)})}{1 + \rho}$$
$$\theta_j^{(t)} = \rho_j \cdot \text{TF}_{k,\lambda}(u_0^{(t)} + \theta_j^{(t-1)}/\rho_j, X_j), \quad j = 1, \dots, d$$

where we let $\rho = \sum_{j=1}^p \rho_j$

Some remarks:

Parallel backfitting for trend filtering

Parallel-ADMM-backfitting: center y , and repeat

$$u_0^{(t)} = \frac{y - \sum_{j=1}^d \theta_j^{(t-1)}}{1 + \rho} + \frac{\rho u_0^{(t-1)}}{1 + \rho} + \frac{\sum_{j=1}^d (\theta_j^{(t-2)} - \theta_j^{(t-1)})}{1 + \rho}$$
$$\theta_j^{(t)} = \rho_j \cdot \text{TF}_{k,\lambda}(u_0^{(t)} + \theta_j^{(t-1)}/\rho_j, X_j), \quad j = 1, \dots, d$$

where we let $\rho = \sum_{j=1}^p \rho_j$

Some remarks:

- Converges under no assumptions

Parallel backfitting for trend filtering

Parallel-ADMM-backfitting: center y , and repeat

$$u_0^{(t)} = \frac{y - \sum_{j=1}^d \theta_j^{(t-1)}}{1 + \rho} + \frac{\rho u_0^{(t-1)}}{1 + \rho} + \frac{\sum_{j=1}^d (\theta_j^{(t-2)} - \theta_j^{(t-1)})}{1 + \rho}$$
$$\theta_j^{(t)} = \rho_j \cdot \text{TF}_{k,\lambda}(u_0^{(t)} + \theta_j^{(t-1)}/\rho_j, X_j), \quad j = 1, \dots, d$$

where we let $\rho = \sum_{j=1}^p \rho_j$

Some remarks:

- Converges under no assumptions
- When $\rho = 1$, reduces to **parallel-Dykstra-backfitting**, in which case we have $u_0^{(t)} = y - \sum_{j=1}^d \theta_j^{(t-1)}$, the residual

Parallel backfitting for trend filtering

Parallel-ADMM-backfitting: center y , and repeat

$$u_0^{(t)} = \frac{y - \sum_{j=1}^d \theta_j^{(t-1)}}{1 + \rho} + \frac{\rho u_0^{(t-1)}}{1 + \rho} + \frac{\sum_{j=1}^d (\theta_j^{(t-2)} - \theta_j^{(t-1)})}{1 + \rho}$$
$$\theta_j^{(t)} = \rho_j \cdot \text{TF}_{k,\lambda}(u_0^{(t)} + \theta_j^{(t-1)}/\rho_j, X_j), \quad j = 1, \dots, d$$

where we let $\rho = \sum_{j=1}^p \rho_j$

Some remarks:

- Converges under no assumptions
- When $\rho = 1$, reduces to **parallel-Dykstra-backfitting**, in which case we have $u_0^{(t)} = y - \sum_{j=1}^d \theta_j^{(t-1)}$, the residual
- If we replace $\text{TF}_{k,\lambda}(\cdot)$ operator by smoothing spline, P-spline, wavelet smoothing, then still converges!

Summary and future work

Summary:

- Dykstra's algorithm and coordinate descent are **equivalent** (act on equivalent dual problems)

Summary and future work

Summary:

- Dykstra's algorithm and coordinate descent are **equivalent** (act on equivalent dual problems)
- This connection provides new convergence results for lasso CD, as well as new **parallel-Dykstra-CD algorithm**

Summary and future work

Summary:

- Dykstra's algorithm and coordinate descent are **equivalent** (act on equivalent dual problems)
- This connection provides new convergence results for lasso CD, as well as new **parallel-Dykstra-CD algorithm**
- ADMM equivalent to Dykstra for product space reformulation and $\rho = 1$. For $\rho > 1$, gives new **parallel-ADMM-CD algorithm**

Summary and future work

Summary:

- Dykstra's algorithm and coordinate descent are **equivalent** (act on equivalent dual problems)
- This connection provides new convergence results for lasso CD, as well as new **parallel-Dykstra-CD algorithm**
- ADMM equivalent to Dykstra for product space reformulation and $\rho = 1$. For $\rho > 1$, gives new **parallel-ADMM-CD** algorithm
- This is all true outside of quadratic loss, except the parallel CD algorithms from Dykstra and ADMM are distinct

Summary and future work

Summary:

- Dykstra's algorithm and coordinate descent are **equivalent** (act on equivalent dual problems)
- This connection provides new convergence results for lasso CD, as well as new **parallel-Dykstra-CD algorithm**
- ADMM equivalent to Dykstra for product space reformulation and $\rho = 1$. For $\rho > 1$, gives new **parallel-ADMM-CD** algorithm
- This is all true outside of quadratic loss, except the parallel CD algorithms from Dykstra and ADMM are distinct

Future work:

Summary and future work

Summary:

- Dykstra's algorithm and coordinate descent are **equivalent** (act on equivalent dual problems)
- This connection provides new convergence results for lasso CD, as well as new **parallel-Dykstra-CD algorithm**
- ADMM equivalent to Dykstra for product space reformulation and $\rho = 1$. For $\rho > 1$, gives new **parallel-ADMM-CD** algorithm
- This is all true outside of quadratic loss, except the parallel CD algorithms from Dykstra and ADMM are distinct

Future work:

- Extend beyond seminorms (projection becomes prox)?

Summary and future work

Summary:

- Dykstra's algorithm and coordinate descent are **equivalent** (act on equivalent dual problems)
- This connection provides new convergence results for lasso CD, as well as new **parallel-Dykstra-CD algorithm**
- ADMM equivalent to Dykstra for product space reformulation and $\rho = 1$. For $\rho > 1$, gives new **parallel-ADMM-CD** algorithm
- This is all true outside of quadratic loss, except the parallel CD algorithms from Dykstra and ADMM are distinct

Future work:

- Extend beyond seminorms (projection becomes prox)?
- CD in Hilbert spaces, via general Dykstra results?

Summary and future work

Summary:

- Dykstra's algorithm and coordinate descent are **equivalent** (act on equivalent dual problems)
- This connection provides new convergence results for lasso CD, as well as new **parallel-Dykstra-CD algorithm**
- ADMM equivalent to Dykstra for product space reformulation and $\rho = 1$. For $\rho > 1$, gives new **parallel-ADMM-CD** algorithm
- This is all true outside of quadratic loss, except the parallel CD algorithms from Dykstra and ADMM are distinct

Future work:

- Extend beyond seminorms (projection becomes prox)?
- CD in Hilbert spaces, via general Dykstra results?
- Asynchronous parallel CD algorithms, via ADMM?

References and acknowledgements

- R. Tibshirani. “Dykstra’s Algorithm, ADMM, and Coordinate Descent: Connections, Insights, and Extensions”. NIPS, 2017.
- V. Sadhanala and R. Tibshirani. “Additive Models with Trend Filtering”. arXiv, 2017.



Veeranjaneyulu Sadhanala

<http://www.stat.cmu.edu/~ryantibs/talks/dykstra-cd-2018.pdf>

Thank you for listening

Bonus time

Alternating conditional expectations

Given random variables $X_1, \dots, X_p, Y$, consider the problem

$$\begin{aligned} \min_{f, g_1, \dots, g_p} \quad & \mathbb{E} \left[\left(f(Y) - \sum_{i=1}^p g_i(X_i) \right)^2 \right] \\ \text{s.t.} \quad & \mathbb{E}[f(Y)] = \mathbb{E}[g_1(X_1)] = \dots = \mathbb{E}[g_p(X_p)] = 0, \\ & \mathbb{E}[f^2(Y)] = 1, \mathbb{E}[g_i^2(X_i)] < \infty, \quad i = 1, \dots, p \end{aligned}$$

Alternating conditional expectations

Given random variables $X_1, \dots, X_p, Y$, consider the problem

$$\begin{aligned} \min_{f, g_1, \dots, g_p} \quad & \mathbb{E} \left[\left(f(Y) - \sum_{i=1}^p g_i(X_i) \right)^2 \right] \\ \text{s.t.} \quad & \mathbb{E}[f(Y)] = \mathbb{E}[g_1(X_1)] = \dots = \mathbb{E}[g_p(X_p)] = 0, \\ & \mathbb{E}[f^2(Y)] = 1, \mathbb{E}[g_i^2(X_i)] < \infty, \quad i = 1, \dots, p \end{aligned}$$

Breiman and Friedman (1985): under regularity conditions, optimal transformations $f^*, g_1^*, \dots, g_p^*$ exist

Alternating conditional expectations

Given random variables $X_1, \dots, X_p, Y$, consider the problem

$$\begin{aligned} \min_{f, g_1, \dots, g_p} \quad & \mathbb{E} \left[\left(f(Y) - \sum_{i=1}^p g_i(X_i) \right)^2 \right] \\ \text{s.t.} \quad & \mathbb{E}[f(Y)] = \mathbb{E}[g_1(X_1)] = \dots = \mathbb{E}[g_p(X_p)] = 0, \\ & \mathbb{E}[f^2(Y)] = 1, \mathbb{E}[g_i^2(X_i)] < \infty, \quad i = 1, \dots, p \end{aligned}$$

Breiman and Friedman (1985): under regularity conditions, optimal transformations $f^*, g_1^*, \dots, g_p^*$ exist and satisfy

$$\begin{aligned} f^*(y) &= \mathbb{E} \left[\sum_{i=1}^p g_i^*(X_i) \mid Y = y \right] / \left\| \mathbb{E} \left[\sum_{i=1}^p g_i^*(X_i) \mid Y \right] \right\| \\ g_i^*(x) &= \mathbb{E} \left[f^*(Y) - \sum_{j \neq i} g_j^*(X_j) \mid X_i = x \right], \quad i = 1, \dots, p \end{aligned}$$

Alternating conditional expectations

Given random variables $X_1, \dots, X_p, Y$, consider the problem

$$\begin{aligned} \min_{f, g_1, \dots, g_p} \quad & \mathbb{E} \left[\left(f(Y) - \sum_{i=1}^p g_i(X_i) \right)^2 \right] \\ \text{s.t.} \quad & \mathbb{E}[f(Y)] = \mathbb{E}[g_1(X_1)] = \dots = \mathbb{E}[g_p(X_p)] = 0, \\ & \mathbb{E}[f^2(Y)] = 1, \mathbb{E}[g_i^2(X_i)] < \infty, \quad i = 1, \dots, p \end{aligned}$$

Breiman and Friedman (1985): under regularity conditions, optimal transformations $f^*, g_1^*, \dots, g_p^*$ exist and satisfy

$$\begin{aligned} f^*(y) &= \mathbb{E} \left[\sum_{i=1}^p g_i^*(X_i) \mid Y = y \right] / \left\| \mathbb{E} \left[\sum_{i=1}^p g_i^*(X_i) \mid Y \right] \right\| \\ g_i^*(x) &= \mathbb{E} \left[f^*(Y) - \sum_{j \neq i} g_j^*(X_j) \mid X_i = x \right], \quad i = 1, \dots, p \end{aligned}$$

Leads to the **alternating conditional expectations** or **ACE** algorithm

Alternating conditional expectations (cont.)

Some definitions:

- For each $i = 1, \dots, p$, denote by H_i the Hilbert space of all measurable functions g_i s.t. $\mathbb{E}[g_i(X_i)] = 0$, $\mathbb{E}[g_i^2(X_i)] < \infty$, endowed with the usual inner product and norm:

$$\langle g_i, h_i \rangle = \mathbb{E}[g_i(X_i)h_i(X_i)], \quad \|g_i\| = \langle g_i, g_i \rangle = \mathbb{E}[g_i^2(X_i)]$$

Alternating conditional expectations (cont.)

Some definitions:

- For each $i = 1, \dots, p$, denote by H_i the Hilbert space of all measurable functions g_i s.t. $\mathbb{E}[g_i(X_i)] = 0$, $\mathbb{E}[g_i^2(X_i)] < \infty$, endowed with the usual inner product and norm:

$$\langle g_i, h_i \rangle = \mathbb{E}[g_i(X_i)h_i(X_i)], \quad \|g_i\| = \langle g_i, g_i \rangle = \mathbb{E}[g_i^2(X_i)]$$

- Define H_0 similarly, but corresponding to Y . Also define B_0 to be the unit ball in H_0

Alternating conditional expectations (cont.)

Some definitions:

- For each $i = 1, \dots, p$, denote by H_i the Hilbert space of all measurable functions g_i s.t. $\mathbb{E}[g_i(X_i)] = 0$, $\mathbb{E}[g_i^2(X_i)] < \infty$, endowed with the usual inner product and norm:

$$\langle g_i, h_i \rangle = \mathbb{E}[g_i(X_i)h_i(X_i)], \quad \|g_i\| = \langle g_i, g_i \rangle = \mathbb{E}[g_i^2(X_i)]$$

- Define H_0 similarly, but corresponding to Y . Also define B_0 to be the unit ball in H_0
- Define H to be the Hilbert space of all functions of the form

$$f(y) + \sum_{i=1}^p g_i(x_i), \quad f \in H_0, \quad g_i \in H_i, \quad i = 1, \dots, p$$

Alternating conditional expectations (cont.)

Some definitions:

- For each $i = 1, \dots, p$, denote by H_i the Hilbert space of all measurable functions g_i s.t. $\mathbb{E}[g_i(X_i)] = 0$, $\mathbb{E}[g_i^2(X_i)] < \infty$, endowed with the usual inner product and norm:

$$\langle g_i, h_i \rangle = \mathbb{E}[g_i(X_i)h_i(X_i)], \quad \|g_i\| = \langle g_i, g_i \rangle = \mathbb{E}[g_i^2(X_i)]$$

- Define H_0 similarly, but corresponding to Y . Also define B_0 to be the unit ball in H_0
- Define H to be the Hilbert space of all functions of the form

$$f(y) + \sum_{i=1}^p g_i(x_i), \quad f \in H_0, \quad g_i \in H_i, \quad i = 1, \dots, p$$

Note that $H_0, H_1, \dots, H_p$ are closed linear subspaces in H

ACE as Dykstra's algorithm?

ACE algorithm (single-loop):

$$f^{(k)}(y) = \mathbb{E} \left[\sum_{i=1}^p g_i^{(k-1)}(X_i) \mid Y = y \right] / \left\| \mathbb{E} \left[\sum_{i=1}^p g_i^{(k-1)}(X_i) \mid Y \right] \right\|$$
$$g_i^{(k)}(x) = \mathbb{E} \left[f^{(k)}(Y) - \sum_{j < i} g_j^{(k)}(X_j) - \sum_{j > i} g_j^{(k-1)}(X_j) \mid X_i = x \right],$$
$$i = 1, \dots, p$$

ACE as Dykstra's algorithm?

ACE algorithm (single-loop):

$$f^{(k)}(y) = \mathbb{E} \left[\sum_{i=1}^p g_i^{(k-1)}(X_i) \mid Y = y \right] / \left\| \mathbb{E} \left[\sum_{i=1}^p g_i^{(k-1)}(X_i) \mid Y \right] \right\|$$
$$g_i^{(k)}(x) = \mathbb{E} \left[f^{(k)}(Y) - \sum_{j < i} g_j^{(k)}(X_j) - \sum_{j > i} g_j^{(k-1)}(X_j) \mid X_i = x \right],$$
$$i = 1, \dots, p$$

This is “almost” of Dykstra form:

$$u_i^{(k)} = P_{C_i}(u_{i-1}^{(k)} + z_i^{(k-1)})$$
$$z_i^{(k)} = u_{i-1}^{(k)} + z_i^{(k-1)} - u_i^{(k)}$$

ACE as Dykstra's algorithm?

ACE algorithm (single-loop):

$$f^{(k)}(y) = \mathbb{E} \left[\sum_{i=1}^p g_i^{(k-1)}(X_i) \mid Y = y \right] / \left\| \mathbb{E} \left[\sum_{i=1}^p g_i^{(k-1)}(X_i) \mid Y \right] \right\|$$
$$g_i^{(k)}(x) = \mathbb{E} \left[f^{(k)}(Y) - \sum_{j < i} g_j^{(k)}(X_j) - \sum_{j > i} g_j^{(k-1)}(X_j) \mid X_i = x \right],$$
$$i = 1, \dots, p$$

This is “almost” of Dykstra form:

$$u_i^{(k)} = P_{C_i}(u_{i-1}^{(k)} + z_i^{(k-1)})$$
$$z_i^{(k)} = u_{i-1}^{(k)} + z_i^{(k-1)} - u_i^{(k)}$$

where we let $u = f - \sum_{i=1}^p g_i$, $z_0 = -f$, $z_i = g_i$, $i = 1, \dots, p$, and $C_i = H_i^\perp$, $i = 0, \dots, p$ (orthocomplements in H).

ACE as Dykstra's algorithm?

ACE algorithm (single-loop):

$$f^{(k)}(y) = \mathbb{E} \left[\sum_{i=1}^p g_i^{(k-1)}(X_i) \mid Y = y \right] / \left\| \mathbb{E} \left[\sum_{i=1}^p g_i^{(k-1)}(X_i) \mid Y \right] \right\|$$
$$g_i^{(k)}(x) = \mathbb{E} \left[f^{(k)}(Y) - \sum_{j < i} g_j^{(k)}(X_j) - \sum_{j > i} g_j^{(k-1)}(X_j) \mid X_i = x \right],$$
$$i = 1, \dots, p$$

This is “almost” of Dykstra form:

$$u_i^{(k)} = P_{C_i}(u_{i-1}^{(k)} + z_i^{(k-1)})$$
$$z_i^{(k)} = u_{i-1}^{(k)} + z_i^{(k-1)} - u_i^{(k)}$$

where we let $u = f - \sum_{i=1}^p g_i$, $z_0 = -f$, $z_i = g_i$, $i = 1, \dots, p$, and $C_i = H_i^\perp$, $i = 0, \dots, p$ (orthocomplements in H). Trouble is **scaling step** ... otherwise Dykstra theory would apply directly