Streaming sparse recovery: $\ell_1$ filtering

- Solving an optimization program like

$$\min_x \tau \|x\|_{\ell_1} + \frac{1}{2} \|\Phi x - y\|_2^2$$

can be costly.

- We want to *update* the solution when
  1. the underlying signal changes slightly, or
  2. we add measurements.
Duality and optimality conditions

Most of the work is done by deriving optimality conditions (a version of KKT) for the solution.

We can show that a vector $x^*$ supported on $\Gamma$ is the unique solution to

$$\min_x \quad \tau \|x\|_1 + \frac{1}{2} \|\Phi x - y\|_2^2$$

if

$$\Phi^T\Gamma (y - \Phi x^*) = \tau \text{sgn}(x^*_\Gamma) \quad \text{on } \Gamma$$

$$\|\Phi^T_{\Gamma^c} (y - \Phi x^*)\|_\infty \leq \tau \quad \text{on } \Gamma^c$$

(Show this on the board ...
Variable $\tau$

Given the support $\Gamma$, the non-zero components of the solution $x^*$ obey

$$x^*_\Gamma = (\Phi_\Gamma^T \Phi_\Gamma)^{-1} \Phi_\Gamma^T y - \tau (\Phi_\Gamma^T \Phi_\Gamma)^{-1} \text{sgn}(x^*_\Gamma)$$

If we were to nudge $\tau$ just a little, the solution would move like

$$\partial x = \begin{cases} (\Phi_\Gamma^T \Phi_\Gamma)^{-1} \text{sgn}(x^*_\Gamma) & \text{on } \Gamma \\ 0 & \text{on } \Gamma^c \end{cases}$$

This direction holds until a component disappears, or a new dual constraint becomes active.

$\Rightarrow$ as we change $\tau$, the path of solutions is piecewise linear
Time-varying sparse signals

- Initial measurements. Observe
  \[ y_0 = \Phi x_0 + e_0 \]

- Initial reconstruction. Solve
  \[
  \min_{x} \tau \|x\|_1 + \frac{1}{2} \|\Phi x - y_0\|_2^2
  \]

- A new set of measurements arrives:
  \[ y_1 = \Phi x_1 + e_1 \]

- Reconstruct again using $\ell_1$-min:
  \[
  \min_{x} \tau \|x\|_1 + \frac{1}{2} \|\Phi x - y_1\|_2^2
  \]

- We can gradually move from the first solution to the second solution using *homotopy*
  \[
  \min \tau \|x\|_1 + \frac{1}{2} \|\Phi x - (1 - \epsilon)y_0 - \epsilon y_1\|_2^2
  \]

  Take $\epsilon$ from $0 \rightarrow 1$
Update direction

\[
\min \tau \|x\|_{\ell_1} + \frac{1}{2} \|\Phi x - (1 - \epsilon)y_{old} - \epsilon y_{new}\|_2^2
\]

- Path from old solution to new solution is *piecewise linear*
- Optimality conditions for fixed \(\epsilon\):
  \[
  \Phi_{\Gamma}^T(\Phi x - (1 - \epsilon)y_{old} - \epsilon y_{new}) = -\tau \text{sign } x_{\Gamma}
  \]
  \[
  \|\Phi_{\Gamma c}^T(\Phi x - (1 - \epsilon)y_{old} - \epsilon y_{new})\|_\infty < \tau
  \]

\(\Gamma = \text{active support}\)
- Update direction:
  \[
  \partial x = \begin{cases} 
  -(\Phi_{\Gamma}^T \Phi_{\Gamma})^{-1} \Phi_{\Gamma}^T(y_{old} - y_{new}) & \text{on } \Gamma \\
  0 & \text{off } \Gamma
  \end{cases}
  \]
Experiments

Sparse signal example, with update. $n=1024$, $m=512$, $T=m/5$, $k \in [0, T/20]$
Experiments

Piecewise constant signal [adapted from WaveLab]

Zoom in for Haar wavelet coefficients
Experiments
## Experiments

<table>
<thead>
<tr>
<th>Signal type</th>
<th>DynamicX* (nProdAtA, CPU)</th>
<th>LASSO homotopy (nProdAtA, CPU)</th>
<th>GPSR-BB (nProdAtA, CPU)</th>
<th>FPC_AS (nProdAtA, CPU)</th>
</tr>
</thead>
<tbody>
<tr>
<td>N = 1024, M = 512, T = m/5, k ~ T/20, Values = +/- 1</td>
<td>(23.72, 0.132)</td>
<td>(235, 0.924)</td>
<td>(104.5, 0.18)</td>
<td>(148.65, 0.177)</td>
</tr>
<tr>
<td>Blocks</td>
<td>(2.7, 0.028)</td>
<td>(76.8, 0.490)</td>
<td>(17, 0.133)</td>
<td>(53.5, 0.196)</td>
</tr>
<tr>
<td>Pcw. Poly.</td>
<td>(13.83, 0.151)</td>
<td>(150.2, 1.096)</td>
<td>(26.05, 0.212)</td>
<td>(66.89, 0.25)</td>
</tr>
<tr>
<td>House slices</td>
<td>(26.2, 0.011)</td>
<td>(53.4, 0.019)</td>
<td>(92.24, 0.012)</td>
<td>(90.9, 0.036)</td>
</tr>
</tbody>
</table>

\[ \tau = 0.01 \| A^T y \|_\infty \]

nProdAtA: roughly the avg. no. of matrix vector products with A and A^T
CPU: average cputime to solve
Adding a measurement: Recursive least-squares

- Classical least-squares:
  solve a system of linear eqns \( y = Ax + e \)
  min energy solution \( \min_x \|Ax - y\|^2_2 \)
  analytical solution \( \hat{x} = (A^T A)^{-1} A^T y \)

- Suppose we add new measurements \( w = B^T x \)

\[
\begin{bmatrix}
A \\
B
\end{bmatrix}
\begin{bmatrix}
x \\
w
\end{bmatrix}
= 
\begin{bmatrix}
y
\end{bmatrix}
\]

\[
\hat{x}_0 = (A^T A + B^T B)^{-1} (A^T y + B^T w)
\]

- Recursive Least-Squares (RLS): easy low-rank update

\[
\hat{x}_1 = \hat{x}_0 + (I + B(A^T A)^{-1} B^T)^{-1} (A^T A)^{-1} B^T (w - B\hat{x}_0)
\]
Adding a measurement: Dynamic $\ell_1$

- We want the analog of RLS for the LASSO. Adding one measurement

\[
\begin{bmatrix}
    y \\
    w
\end{bmatrix} = \begin{bmatrix}
    \Phi \\
    b
\end{bmatrix} x + \begin{bmatrix}
    e \\
    d
\end{bmatrix} \rightarrow \min_x \tau \|x\|_{\ell_1} + \frac{1}{2} \|\Phi x - y\|_2^2 + \frac{1}{2} \|bx - w\|_2^2
\]

- Challenges:
  - not as smooth as least-squares update
  - solution can change drastically with just one new measurement
  - need to move slowly, use a homotopy method

(see also work by Garrigues et al. 08)
Dynamic $\ell_1$ update

- Work in the new measurement slowly

$$\min \tau \|x\|_{\ell_1} + \frac{1}{2} \left( \|\Phi x - y\|_2^2 + \epsilon \|bx - w\|_2^2 \right)$$

Again, the solution path is piecewise linear in $\epsilon$

- Optimality conditions

$$\Phi_{\Gamma}^T(\Phi x - y) + \epsilon b_{\Gamma}^T(bx - w) = -\tau \text{ sign } x_{\Gamma}$$

$$\|\Phi_{\Gamma_c}^T(\Phi x - y) + \epsilon b_{\Gamma_c}^T(bx - w)\|_\infty < \tau$$

- From critical point $x^{\epsilon_0}$, update direction is

$$\partial x = \begin{cases} 
(\Phi_{\Gamma}^T \Phi_{\Gamma} + \epsilon_0 b_{\Gamma}^T b_{\Gamma})^{-1} b_{\Gamma}^T (w - bx^{\epsilon_0}) & \text{on } \Gamma \\
0 & \text{off } \Gamma 
\end{cases}$$
Number of steps per update

Measurements $m = 150$
Signal length $n = 256$
Summary of $\ell_1$ filtering

- Instead of solving new programs from scratch, work the new data in slowly using homotopy continuation
- Proper homotopy formulation allows us to (easily) use optimality conditions to “hop” along the path of solutions
- Each “hop” costs $O(mn)$ — a few matrix-vector multiplies
- Small number of “hops” if the solutions are closely related