Sparse Representations in Power Systems

Jack Cooper

University of Bremen

August 6, 2009

Outline

- Background
  - Power Signals
  - Clemson Work
- Theory
- Algorithms
- Operators/Dictionaries
- Application
- Conclusions

Signal Properties

- Characteristics of a Power Signal (Xu)

Signal Properties

- Characteristics of a Power Signal (Xu)
  - Low frequency harmonics - caused by power electronic devices, arc furnaces, transformers, rotational machines, and aggregate loads.

Signal Properties

- Characteristics of a Power Signal (Xu)
  - Low frequency harmonics - caused by power electronic devices, arc furnaces, transformers, rotational machines, and aggregate loads.
  - Swells and Sags of the RMS Voltage.

Signal Properties

- Characteristics of a Power Signal (Xu)
  - Low frequency harmonics - caused by power electronic devices, arc furnaces, transformers, rotational machines, and aggregate loads.
  - Swells and Sags of the RMS Voltage.
  - Transients - caused by lightnings, equipment faults, switching operations, and more.

Transients

- Transient Disturbances

Transients

- Transient Disturbances
  - Superimposed to the fundamental frequency.

Transients

- Transient Disturbances
  - Superimposed to the fundamental frequency.
  - Modeled by damped sinusoids to appear as short lived impulses.

Transients

- Transient Disturbances
  - Superimposed to the fundamental frequency.
  - Modeled by damped sinusoids to appear as short lived impulses.
  - Capacitor switching transients
    - damped sinusoids with frequency around 1000-2000 Hz
    - damping factor 450-500
    - phase angle $-\pi/2$ to $\pi/2$.

Example Signal with 60Hz Component

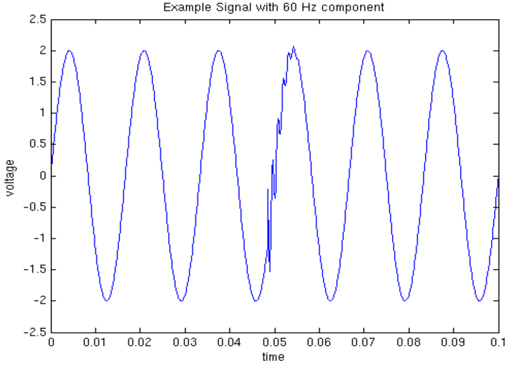

Figure: The test signal with the 60 Hz component, without noise sampled at 1024 points from 0 to .1 seconds.

Sparse Framework

- Feature extraction - detect and identify transient disturbance

Sparse Framework

- Feature extraction - detect and identify transient disturbance
- Finding sparse solutions allows for a more crisp identification, more robust in noise

Sparse Framework

- Feature extraction - detect and identify transient disturbance
- Finding sparse solutions allows for a more crisp identification, more robust in noise

$$\min_{\alpha} \|\alpha\|_0 \text{ subject to } x = \Phi\alpha \tag{1}$$

Sparse Framework

- Feature extraction - detect and identify transient disturbance
- Finding sparse solutions allows for a more crisp identification, more robust in noise

$$\min_{\alpha} \|\alpha\|_0 \text{ subject to } x = \Phi\alpha \tag{1}$$

NP-Hard, instead

$$\min_{\alpha} \|\alpha\|_1 \text{ subject to } x = \Phi\alpha \tag{2}$$

Or in noise,

$$\min_{\alpha} \lambda \|\alpha\|_1 + \frac{1}{2} \|x - \Phi\alpha\|_2^2 \tag{3}$$

Clemson Research

- Bruckstein, Donoho, Elad - From Sparse Solutions of Systems of Equations to Sparse Modeling of Signals and Images
- Chen, Donoho, Saunders - Atomic Decomposition by Basis Pursuit
- Donoho, Elad - On the Stability of Basis Pursuit in the Presence of Noise
- Donoho, Elad, Temlyakov - Stable Recovery of Sparse Overcomplete Representations in the Precense of Noise
- Tropp - Greed is Good: Algorithmic Results for Sparse Approximation

Clemson Research

- Two different (overcomplete) dictionaries - Wavelet Packet and Damped Sinusoid

Clemson Research

- Two different (overcomplete) dictionaries - Wavelet Packet and Damped Sinusoid
- Two different algorithms - Matching Pursuit and Interior Point Method

Clemson Research

- Two different (overcomplete) dictionaries - Wavelet Packet and Damped Sinusoid
- Two different algorithms - Matching Pursuit and Interior Point Method
- Mutual Coherence analysis

$$\mu(\Phi) = \max_{1\le k,j\le m, k\ne j} \frac{\left|\phi_k^T \phi_j\right|}{\left\|\phi_k\right\|_2 \left\|\phi_j\right\|_2}$$

Used for criterea on finding sparest solution possible without noise, stability in noise, etc. Often insufficient.

Clemson Research

- Two different (overcomplete) dictionaries - Wavelet Packet and Damped Sinusoid
- Two different algorithms - Matching Pursuit and Interior Point Method
- Mutual Coherence analysis

$$\mu(\Phi) = \max_{1 \le k,j \le m, k \ne j} \frac{\left|\phi_k^T \phi_j\right|}{\left\|\phi_k\right\|_2 \left\|\phi_j\right\|_2}$$

  Used for criterea on finding sparest solution possible without noise, stability in noise, etc. Often insufficient.
- Time Frequency Plane Comparision

Lingering Questions

- What about the infinite dimesional setting?
- What other dictionaries might be better?
- What other algorithms are there?

Infinite Dimensional Setting

- Daubechies, Defrise, De Mol - An Iterative Thresholding Algorithm for Linear Inverse Problems with a Sparsity Constraint

Infinite Dimensional Setting

- Daubechies, Defrise, De Mol - An Iterative Thresholding Algorithm for Linear Inverse Problems with a Sparsity Constraint
- Using an operator K, instead of a Dictionary Φ

Infinite Dimensional Setting

- Daubechies, Defrise, De Mol - An Iterative Thresholding Algorithm for Linear Inverse Problems with a Sparsity Constraint
- Using an operator K, instead of a Dictionary Φ
- Considering problem as ill-posed, unbounded or badly conditioned generalized inverse in need or regularization

Infinite Dimensional Setting

- Daubechies, Defrise, De Mol - An Iterative Thresholding Algorithm for Linear Inverse Problems with a Sparsity Constraint
- Using an operator K, instead of a Dictionary Φ
- Considering problem as ill-posed, unbounded or badly conditioned generalized inverse in need or regularization
- Frequent restriction to injective K instead of Mutual Coherence.

Infinite Dimensional Setting

- Daubechies, Defrise, De Mol - An Iterative Thresholding Algorithm for Linear Inverse Problems with a Sparsity Constraint
- Using an operator K, instead of a Dictionary Φ
- Considering problem as ill-posed, unbounded or badly conditioned generalized inverse in need or regularization
- Frequent restriction to injective K instead of Mutual Coherence.
  - excludes overcomplete basis.

Infinite Dimensional Setting

- Daubechies, Defrise, De Mol - An Iterative Thresholding Algorithm for Linear Inverse Problems with a Sparsity Constraint
- Using an operator K, instead of a Dictionary Φ
- Considering problem as ill-posed, unbounded or badly conditioned generalized inverse in need or regularization
- Frequent restriction to injective K instead of Mutual Coherence.
  - excludes overcomplete basis.
- Choose sparse solution from overcomplete basis versus use sparsity constraint as regularization of ill posed problem with injective K

Infinite Dimensional Setting

- Daubechies, Defrise, De Mol - An Iterative Thresholding Algorithm for Linear Inverse Problems with a Sparsity Constraint
- Using an operator K, instead of a Dictionary Φ
- Considering problem as ill-posed, unbounded or badly conditioned generalized inverse in need or regularization
- Frequent restriction to injective K instead of Mutual Coherence.
  - excludes overcomplete basis.
- Choose sparse solution from overcomplete basis versus use sparsity constraint as regularization of ill posed problem with injective K
- Difference in how the l^1 penalty function is brought in: regularization compared to the recasting of

$$\min_{\alpha} ||\alpha||_1 \text{ subject to } |S - D\alpha| \leq \delta$$

to

$$\min_{\alpha} \lambda \|\alpha\|_1 + \frac{1}{2} \|x - \Phi\alpha\|_2^2 \tag{4}$$

Regularization

- To what extent is minimizing the functional

$$\Phi(f) = \frac{1}{2}||Kf - g||_2^2 + \alpha||f||_1 \tag{5}$$

a successful regularization?

Daubechies

Theorem (Daubechies, Defrise, De Mol)

For injective K, a bounded linear operator with $||K|| < 1$ *and* $\alpha > 0$*, let* f^* *be the minimizer of* Φ*. Then for any observed image* g*, if* $\alpha(\epsilon)$ *satisfies*

$$\lim_{\epsilon \longrightarrow 0} \alpha(\epsilon) = 0 \text{ and } \lim_{\epsilon \longrightarrow 0} \frac{\epsilon^2}{\alpha(\epsilon)} = 0$$

Then,

$$\lim_{\epsilon \longrightarrow 0} [\sup_{||g - K f_0|| \leq \epsilon} ||f^* - f_0||] = 0$$

where f_0 *is the "true" solution.*

Daubechies

Theorem (Daubechies, Defrise, De Mol)

For injective K, a bounded linear operator with $||K|| < 1$ and $\alpha > 0$, let f^* be the minimizer of Φ. Then for any observed image g, if $\alpha(\epsilon)$ satisfies

$$\lim_{\epsilon \to 0} \alpha(\epsilon) = 0 \text{ and } \lim_{\epsilon \to 0} \frac{\epsilon^2}{\alpha(\epsilon)} = 0$$

Then,

$$\lim_{\epsilon \to 0} [\sup_{||g - K f_0|| \leq \epsilon} ||f^* - f_0||] = 0$$

where f_0 is the "true" solution.

This is presented in the literature with a range for p. p=1 is used for this context.

Donoho, Elad

Theorem (Donoho, Elad)

With noise level ϵ, a dictionary D of size NxL, and mutual coherence $\mu(D)$, then if the true solution α_0 is sparse enough, satisfying

$$||\alpha_0||_0 < \frac{1+\mu(D)}{2\mu(D)+\sqrt{N}(\epsilon+\delta)/T}$$

then the solution $\hat{\alpha}$ of

$$\min_{\alpha} ||\alpha||_1 \text{ subject to } |S - D\alpha| \leq \delta$$

with $\delta \geq \epsilon$ exhibits stability

$$||\hat{\alpha} - \alpha_0||_1 < T.$$

Donoho, Elad

Theorem (Donoho, Elad)

With noise level ϵ, a dictionary D of size NxL, and mutual coherence $\mu(D)$, then if the true solution α_0 is sparse enough, satisfying

$$||\alpha_0||_0 < \frac{1+\mu(D)}{2\mu(D)+\sqrt{N}(\epsilon+\delta)/T}$$

then the solution $\hat{\alpha}$ of

$$\min_{\alpha} ||\alpha||_1 \text{ subject to } |S - D\alpha| \leq \delta$$

with $\delta \geq \epsilon$ exhibits stability

$$||\hat{\alpha} - \alpha_0||_1 < T.$$

The dictionary can be overcomplete.

Donoho, Elad, Temlyakov

Theorem (Donoho, Elad, Temlyakov)

With noise level ϵ, a dictionary D of size NxL, and mutual coherence $\mu(D)$, then if the true solution α_0 exists with

$$M = ||\alpha_0||_0 < \frac{(\frac{1}{\mu(D)} + 1)}{4}$$

then the solution $\hat{\alpha}$ of

$$\min_{\alpha} ||\alpha||_1 \text{ subject to } |S - D\alpha| \leq \delta$$

with $\delta \geq \epsilon$ exhibits stability

$$||\hat{\alpha} - \alpha_0||_2^2 \leq \frac{(\epsilon + \delta)^2}{1 - \mu(D)(4M - 1)}.$$

Donoho, Elad, Temlyakov

Theorem (Donoho, Elad, Temlyakov)

With noise level ϵ, a dictionary D of size NxL, and mutual coherence $\mu(D)$, then if the true solution α_0 exists with

$$M = ||\alpha_0||_0 < \frac{(\frac{1}{\mu(D)} + 1)}{4}$$

then the solution $\hat{\alpha}$ of

$$\min_{\alpha} ||\alpha||_1 \text{ subject to } |S - D\alpha| \leq \delta$$

with $\delta \geq \epsilon$ exhibits stability

$$||\hat{\alpha} - \alpha_0||_2^2 \leq \frac{(\epsilon + \delta)^2}{1 - \mu(D)(4M - 1)}.$$

Other conditions from Tropp

Iterative Soft Thresholding - Daubechies, Defrise, De Mol

- Paper in infinite dimensional setting

Iterative Soft Thresholding - Daubechies, Defrise, De Mol

- Paper in infinite dimensional setting
- One parameter choice is to minimize l^1 penalty functional

$$\Phi_\alpha(f) = ||Kf - g||_2^2 + \alpha||f||_1$$

Iterative Soft Thresholding - Daubechies, Defrise, De Mol

- Paper in infinite dimensional setting
- One parameter choice is to minimize l^1 penalty functional

$$\Phi_\alpha(f) = ||Kf - g||_2^2 + \alpha||f||_1$$

- Minimize this by minimizing a surrogate functional

$$\Phi_\alpha^{SUR}(f;a) = ||Kf - g||^2 + \alpha||f||_1 - ||Kf - Ka||^2 + ||f - a||^2$$

Iterative Soft Thresholding - Daubechies, Defrise, De Mol

- Paper in infinite dimensional setting
- One parameter choice is to minimize l^1 penalty functional

$$\Phi_\alpha(f) = ||Kf - g||_2^2 + \alpha||f||_1$$

- Minimize this by minimizing a surrogate functional

$$\Phi_\alpha^{SUR}(f; a) = ||Kf - g||^2 + \alpha||f||_1 - ||Kf - Ka||^2 + ||f - a||^2$$

- Iterate to with $f^n = \arg\min(\Phi_\alpha^{SUR}(f; f^{n-1}))$

Iterative Soft Thresholding - Daubechies, Defrise, De Mol

- Paper in infinite dimensional setting
- One parameter choice is to minimize l^1 penalty functional

$$\Phi_\alpha(f) = ||Kf - g||_2^2 + \alpha||f||_1$$

- Minimize this by minimizing a surrogate functional

$$\Phi_\alpha^{SUR}(f;a) = ||Kf - g||^2 + \alpha||f||_1 - ||Kf - Ka||^2 + ||f - a||^2$$

- Iterate to with $f^n = \arg\min(\Phi_\alpha^{SUR}(f;f^{n-1}))$
- This minimizer satisfies $f = S_\alpha\left(a + [K^*(g - Ka)]\right)$ where

$$S_\alpha(x) = \begin{cases} x - \frac{\omega}{2} & \text{if } x \geq \frac{\omega}{2} \\ 0 & \text{if } |x| < \frac{\omega}{2} \\ x + \frac{\omega}{2} & \text{if} x \leq -\frac{\omega}{2} \end{cases}$$

Iterative Soft Thresholding - Daubechies, Defrise, De Mol

- Paper in infinite dimensional setting
- One parameter choice is to minimize l^1 penalty functional

$$\Phi_\alpha(f) = ||Kf - g||_2^2 + \alpha||f||_1$$

- Minimize this by minimizing a surrogate functional

$$\Phi_\alpha^{SUR}(f;a) = ||Kf - g||^2 + \alpha||f||_1 - ||Kf - Ka||^2 + ||f - a||^2$$

- Iterate to with $f^n = \arg\min(\Phi_\alpha^{SUR}(f;f^{n-1}))$
- This minimizer satisfies $f = S_\alpha\left(a + [K^*(g - Ka)]\right)$ where

$$S_\alpha(x) = \begin{cases} x - \frac{\omega}{2} & \text{if } x \geq \frac{\omega}{2} \\ 0 & \text{if } |x| < \frac{\omega}{2} \\ x + \frac{\omega}{2} & \text{if} x \leq -\frac{\omega}{2} \end{cases}$$

- Iterate using f^{n-1} for a.

Iterative Soft Thresholding

- When these f^n's converge to the surrogate minimizer, it also matches the minimizer of Φ!

Iterative Soft Thresholding

- When these f^n's converge to the surrogate minimizer, it also matches the minimizer of Φ!
- if $||K|| < 1$ and injective then these iterations converge strongly to the minimizer of Φ, regardless of the initial choice.

Iterative Hard Thresholding - Bredies, Lorenz

- Similar to soft thresholding but with a function

$$H(x) = \begin{cases} 0 \text{ for } |x| \leq 1 \\ \frac{||f||^2}{2\alpha} \text{ for } |x| > 1 \end{cases}$$

Iterative Hard Thresholding - Bredies, Lorenz

- Similar to soft thresholding but with a function

$$H(x) = \begin{cases} 0 \text{ for } |x| \leq 1 \\ \frac{||f||^2}{2\alpha} \text{ for } |x| > 1 \end{cases}$$

- Another function used in the algorithm:

$$\varphi(x) = \begin{cases} |x| \text{ for } |x| \leq \frac{||f||^2}{2\alpha} \\ \frac{\alpha}{||f||^2}(x^2 + (\frac{||f||^2}{2\alpha})^2 \text{ for } |x| > \frac{||f||^2}{2\alpha} \end{cases}$$

Iterative Hard Thresholding - Bredies, Lorenz

- Initialize $u^0 = 0$

Iterative Hard Thresholding - Bredies, Lorenz

- Initialize $u^0 = 0$
- Direction Determination:

$$v^n = H\left(\frac{-\left(K^*\left(Ku^n - f\right)\right)}{\alpha}\right).$$

Iterative Hard Thresholding - Bredies, Lorenz

- Initialize $u^0 = 0$
- Direction Determination:

$$v^n = H\left(\frac{-\left(K^*\left(Ku^n - f\right)\right)}{\alpha}\right).$$

- Step Size:

$$s_n = \min\{1, \frac{\alpha(\varphi(u^n) - \varphi(v^n)) + (K^*(Ku^n - f))(u^n - v^n)}{||K(v^n - u^n)||^2}\}$$

where the expression makes sense and $s_n = 1$ otherwise.

Iterative Hard Thresholding - Bredies, Lorenz

- Initialize $u^0 = 0$
- Direction Determination:

$$v^n = H\left(\frac{-\left(K^*\left(Ku^n - f\right)\right)}{\alpha}\right).$$

- Step Size:

$$s_n = \min\{1, \frac{\alpha(\varphi(u^n) - \varphi(v^n)) + (K^*(Ku^n - f))(u^n - v^n)}{||K(v^n - u^n)||^2}\}$$

  where the expression makes sense and $s_n = 1$ otherwise.
- Iterate: $u^{n+1} = u^n + s_n(v^n - u^n)$

Iterative Hard Thresholding - Bredies, Lorenz

- Initialize $u^0 = 0$
- Direction Determination:

$$v^n = H\left(\frac{-\left(K^*\left(Ku^n - f\right)\right)}{\alpha}\right).$$

- Step Size:

$$s_n = \min\{1, \frac{\alpha(\varphi(u^n) - \varphi(v^n)) + (K^*(Ku^n - f))(u^n - v^n)}{||K(v^n - u^n)||^2}\}$$

  where the expression makes sense and $s_n = 1$ otherwise.
- Iterate: $u^{n+1} = u^n + s_n(v^n - u^n)$
- Converges like $n^{-\frac{1}{2}}$ for the l^1 case

Semismooth Newton Method - Griesse, Lorenz

- Solve l^1 penalty minimization by finding the solution to

$$u - S_\alpha(u - \gamma K^*(Ku - f)) = 0$$

where

$$S_\alpha(u_k) = \max\{0, |u_k| - \alpha\}\mathsf{sgn}(u_k)$$

for some $\gamma > 0$.

Semismooth Newton Method - Griesse, Lorenz

- Solve l^1 penalty minimization by finding the solution to

$$u - S_\alpha(u - \gamma K^*(Ku - f)) = 0$$

where

$$S_\alpha(u_k) = \max\{0, |u_k| - \alpha\}\mathrm{sgn}(u_k)$$

for some $\gamma > 0$.

- Active Set Algorithm
  1. Initialize Active Sets, set sign vector

Semismooth Newton Method - Griesse, Lorenz

- Solve l^1 penalty minimization by finding the solution to

$$u - S_\alpha(u - \gamma K^*(Ku - f)) = 0$$

where

$$S_\alpha(u_k) = \max\{0, |u_k| - \alpha\}\mathrm{sgn}(u_k)$$

for some $\gamma > 0$.

- Active Set Algorithm
  1. Initialize Active Sets, set sign vector
  2. Set $u^n_{I_n} = 0$ and calculate $u^n_{A_n}$ by solving

$$K^* K_{A_n,A_n} u^n_{A_n} = (K^* f + (sgn)^n \alpha)_{A_n}$$

Semismooth Newton Method - Griesse, Lorenz

- Solve l^1 penalty minimization by finding the solution to

$$u - S_\alpha(u - \gamma K^*(Ku - f)) = 0$$

where

$$S_\alpha(u_k) = \max\{0, |u_k| - \alpha\}\text{sgn}(u_k)$$

for some $\gamma > 0$.

- Active Set Algorithm
  1. Initialize Active Sets, set sign vector
  2. Set $u^n_{I_n} = 0$ and calculate $u^n_{A_n}$ by solving

$$K^* K_{A_n, A_n} u^n_{A_n} = (K^* f + (sgn)^n \alpha)_{A_n}$$

  3. Update Active Set

$$A_{n+1} = \{k \in \mathbb{N} | u^n - \gamma K^*(Ku^n - f)|_k > \gamma\alpha\}$$

Semismooth Newton Method - Griesse, Lorenz

- Solve l^1 penalty minimization by finding the solution to

$$u - S_\alpha(u - \gamma K^*(Ku - f)) = 0$$

where

$$S_\alpha(u_k) = \max\{0, |u_k| - \alpha\}\text{sgn}(u_k)$$

for some $\gamma > 0$.

- Active Set Algorithm
  1. Initialize Active Sets, set sign vector
  2. Set $u^n_{I_n} = 0$ and calculate $u^n_{A_n}$ by solving

$$K^* K_{A_n, A_n} u^n_{A_n} = (K^* f + (sgn)^n \alpha)_{A_n}$$

  3. Update Active Set

$$A_{n+1} = \{k \in \mathbb{N} | u^n - \gamma K^*(Ku^n - f)|_k > \gamma\alpha\}$$

  4. Update sign vector

Semismooth Newton Method - Griesse, Lorenz

- Solve l^1 penalty minimization by finding the solution to

$$u - S_\alpha(u - \gamma K^*(Ku - f)) = 0$$

  where

$$S_\alpha(u_k) = \max\{0, |u_k| - \alpha\}\text{sgn}(u_k)$$

  for some $\gamma > 0$.
- Active Set Algorithm
  1. Initialize Active Sets, set sign vector
  2. Set $u^n_{I_n} = 0$ and calculate $u^n_{A_n}$ by solving

$$K^* K_{A_n, A_n} u^n_{A_n} = (K^* f + (sgn)^n \alpha)_{A_n}$$

  3. Update Active Set

$$A_{n+1} = \{k \in \mathbb{N} | u^n - \gamma K^*(Ku^n - f)|_k > \gamma\alpha\}$$

  4. Update sign vector
  5. End if active set does not change, iterate if it does

Semismooth Newton Method

- Converges locally superlinear
- If it converges, then it converges to the global solution

Feature Sign Search - Lee, Battle, Raina, Ng

- Paper written in finite dimensional setting

Feature Sign Search - Lee, Battle, Raina, Ng

- Paper written in finite dimensional setting
- Active Set algorithm

Feature Sign Search - Lee, Battle, Raina, Ng

- Paper written in finite dimensional setting
- Active Set algorithm
-
    1. Initialize Active Set, sign vector

Feature Sign Search - Lee, Battle, Raina, Ng

- Paper written in finite dimensional setting
- Active Set algorithm
-
  1. Initialize Active Set, sign vector
  2. Find $i = \arg\max_i \left|\frac{\partial ||y - Ax||^2}{\partial x_i}\right|$

Feature Sign Search - Lee, Battle, Raina, Ng

- Paper written in finite dimensional setting
- Active Set algorithm
-
    1. Initialize Active Set, sign vector
    2. Find $i = \arg\max_i |\frac{\partial ||y-Ax||^2}{\partial x_i}|$
    3. Activate x_1 if it improves the objective ie if

$$|\frac{\partial ||y-Ax||^2}{\partial x_i}| > \alpha$$

then add i to the active set and adjust the sign vector accordingly

Feature Sign Search - Lee, Battle, Raina, Ng

- Paper written in finite dimensional setting
- Active Set algorithm
-
  1. Initialize Active Set, sign vector
  2. Find $i = \arg\max_i |\frac{\partial ||y - Ax||^2}{\partial x_i}|$
  3. Activate x_1 if it improves the objective ie if

     $$|\frac{\partial ||y - Ax||^2}{\partial x_i}| > \alpha$$

     then add i to the active set and adjust the sign vector accordingly
  4. Feature Sign Step

     Compute the solution to the unconstrainted QP over just the active set, A

     $$x_{new} = (A^T A)^{-1}(A^T y - \frac{\alpha \mathrm{sgn}(x)}{2})$$

Feature Sign Search - Lee, Battle, Raina, Ng

- Paper written in finite dimensional setting
- Active Set algorithm
-
  1. Initialize Active Set, sign vector
  2. Find $i = \arg\max_i |\frac{\partial ||y-Ax||^2}{\partial x_i}|$
  3. Activate x_1 if it improves the objective ie if
  $$|\frac{\partial ||y-Ax||^2}{\partial x_i}| > \alpha$$
  then add i to the active set and adjust the sign vector accordingly
  4. Feature Sign Step
  Compute the solution to the unconstrainted QP over just the active set, A
  $$x_{new} = (A^T A)^{-1}(A^T y - \frac{\alpha \text{sgn}(x)}{2})$$
  Using a line search from x to x_{new} choose the x that minimizes the objective.

Feature Sign Search - Lee, Battle, Raina, Ng

- Paper written in finite dimensional setting
- Active Set algorithm
-
  1. Initialize Active Set, sign vector
  2. Find $i = \arg\max_i |\frac{\partial ||y-Ax||^2}{\partial x_i}|$
  3. Activate x_1 if it improves the objective ie if
  $$|\frac{\partial ||y-Ax||^2}{\partial x_i}| > \alpha$$
  then add i to the active set and adjust the sign vector accordingly
  4. Feature Sign Step
  Compute the solution to the unconstrainted QP over just the active set, A
  $$x_{new} = (A^T A)^{-1}(A^T y - \frac{\alpha \text{sgn}(x)}{2})$$
  Using a line search from x to x_{new} choose the x that minimizes the objective. Update the active set and the sign vector

Feature Sign Search

- Check optimality

  For nonzero coefficients: $\frac{\partial ||y - Ax||^2}{\partial x_i} + \alpha \text{sgn}(x_i) = 0$

  If not go back to Step 4 (no new activation)

Feature Sign Search

- Check optimality
  For nonzero coefficients: $\frac{\partial ||y-Ax||^2}{\partial x_i} + \alpha \text{sgn}(x_i) = 0$
  If not go back to Step 4 (no new activation)
  For zero coefficients: $|\frac{\partial ||y-Ax||^2}{\partial x_i}| \leq \alpha$ If not go back to Step 3

Feature Sign Search

- Check optimality

  For nonzero coefficients: $\frac{\partial ||y-Ax||^2}{\partial x_i} + \alpha \text{sgn}(x_i) = 0$

  If not go back to Step 4 (no new activation)

  For zero coefficients: $|\frac{\partial ||y-Ax||^2}{\partial x_i}| \leq \alpha$ If not go back to Step 3
- Finds the Global Minimizer

Feature Sign Search

- Check optimality
  For nonzero coefficients: $\frac{\partial ||y-Ax||^2}{\partial x_i} + \alpha \text{sgn}(x_i) = 0$
  If not go back to Step 4 (no new activation)
  For zero coefficients: $|\frac{\partial ||y-Ax||^2}{\partial x_i}| \leq \alpha$ If not go back to Step 3
- Finds the Global Minimizer
- Considers overcomplete dictionaries

Elastic Net

- Extend the l^1 penalty to include an l^2 regularization also

$$\Phi_\alpha(f) = ||Kf - g||_2^2 + \alpha||f||_1 + \frac{\beta}{2}||f||_2^2$$

Elastic Net

- Extend the l^1 penalty to include an l^2 regularization also

$$\Phi_\alpha(f) = ||Kf - g||_2^2 + \alpha||f||_1 + \frac{\beta}{2}||f||_2^2$$

- Same algorithms, different functional

Elastic Net

- Extend the l^1 penalty to include an l^2 regularization also

$$\Phi_\alpha(f) = ||Kf - g||_2^2 + \alpha||f||_1 + \frac{\beta}{2}||f||_2^2$$

- Same algorithms, different functional
- Helps ill-conditioning in Active Set algorithms

Elastic Net

- Extend the l^1 penalty to include an l^2 regularization also

$$\Phi_\alpha(f) = ||Kf - g||_2^2 + \alpha||f||_1 + \frac{\beta}{2}||f||_2^2$$

- Same algorithms, different functional
- Helps ill-conditioning in Active Set algorithms
- Has unique minimizer even for noninjective K

Basis Learning - Lee, Battle, Raina, Ng

- Optimize basis vectors for sparse reconstruction by fixing the observed and the input.

Basis Learning - Lee, Battle, Raina, Ng

- Optimize basis vectors for sparse reconstruction by fixing the observed and the input.
- Use a sparse coding algorithm to represent training signals in basis then optimize using Lagrange Dual:

$$\text{trace}(X^T X - XS^T(SS^T + \Lambda)^{-1}(XS^T)^T - c\Lambda)$$

Basis Learning - Lee, Battle, Raina, Ng

- Optimize basis vectors for sparse reconstruction by fixing the observed and the input.
- Use a sparse coding algorithm to represent training signals in basis then optimize using Lagrange Dual:

$$\text{trace}(X^T X - XS^T(SS^T + \Lambda)^{-1}(XS^T)^T - c\Lambda)$$

- Newton search to maximize this dual

Basis Learning - Lee, Battle, Raina, Ng

- Optimize basis vectors for sparse reconstruction by fixing the observed and the input.
- Use a sparse coding algorithm to represent training signals in basis then optimize using Lagrange Dual:

$$\text{trace}(X^T X - XS^T(SS^T + \Lambda)^{-1}(XS^T)^T - c\Lambda)$$

- Newton search to maximize this dual
- Optimal basis is of the form

$$B^T = (SS^T + \Lambda)^{-1}(XS^T)^T$$

Basis Learning - Lee, Battle, Raina, Ng

- Optimize basis vectors for sparse reconstruction by fixing the observed and the input.
- Use a sparse coding algorithm to represent training signals in basis then optimize using Lagrange Dual:

$$\text{trace}(X^T X - XS^T(SS^T + \Lambda)^{-1}(XS^T)^T - c\Lambda)$$

- Newton search to maximize this dual
- Optimal basis is of the form

$$B^T = (SS^T + \Lambda)^{-1}(XS^T)^T$$

- Iterate this, shrinking B based on zero columns.

Test Signal 1

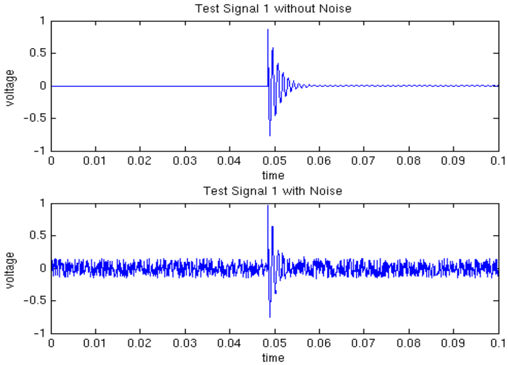

Figure: Test Signal 1 sampled at 1024 points from 0 to .1

Test Signal 2

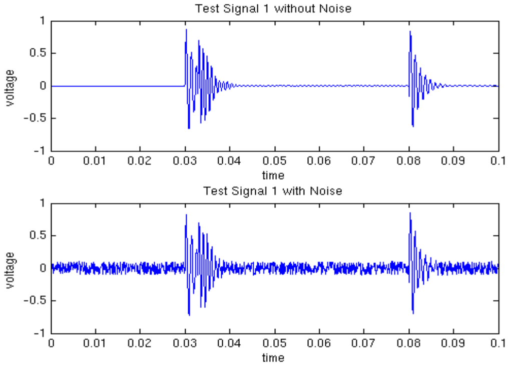

Figure: Test Signal 2 sampled at 1024 points from 0 to .1

How to choose alpha?

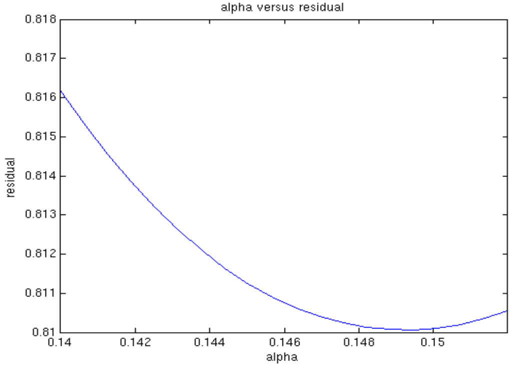

Figure: Alpha values versus the residual of the reconstruction and the noiseless signal. The minimizer is chosen.

Wavelet Dictionary

- Mother wavelet chosen to be Daubechies 4 wavelets

Wavelet Dictionary

- Mother wavelet chosen to be Daubechies 4 wavelets
- Fit well to short impulses

Wavelet Dictionary

- Mother wavelet chosen to be Daubechies 4 wavelets
- Fit well to short impulses
- Level 4 depth

Wavelet Dictionary

- Mother wavelet chosen to be Daubechies 4 wavelets
- Fit well to short impulses
- Level 4 depth
- Previously enumerated all waveforms of a wavelet packet - prohibitavely big, now can use function handle

Wavelet Dictionary Results

- The optimal alpha value is selected at .1467
- The number of nonzeros is 113.

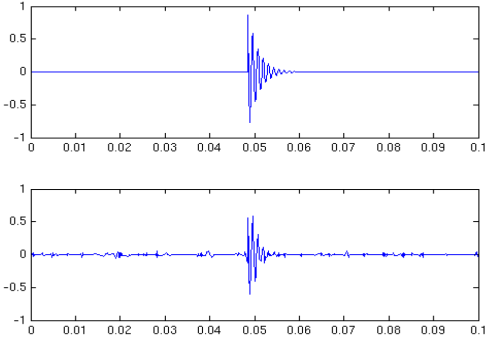

Figure: The reconstruction compared to the noiseless signal

Wavelet Dictionary Results

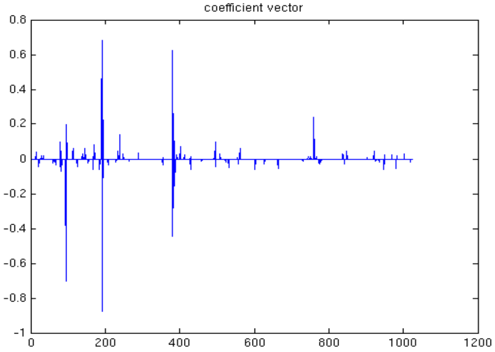

Figure: The coefficient vector

Wavelet Dictionary Results

Algorithm	Runtime (s)
Soft Thresholding	0.022067
Hard Thresholding	max Iter
SSN	∞
FSS	0.690612
Elastic Net SSN	0.370986
Elastic Net FSS	0.729112
GPSR	0.099766
Interior Point	15.021680

Table: Algorithm running times on Wavelet Operator, Test Signal 1

Wavelet Dictionary Results Signal 2

- The optimal alpha value is selected at .0747
- The number of nonzeros is 291

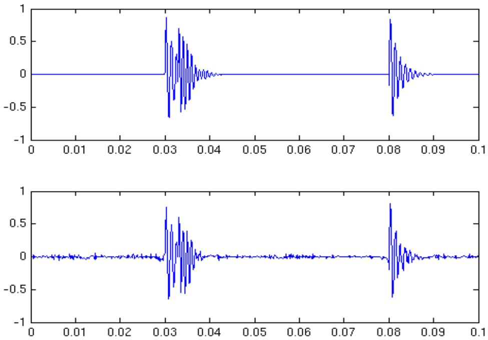

Figure: The reconstruction compared to the noiseless signal

Wavelet Dictionary Results Signal 2

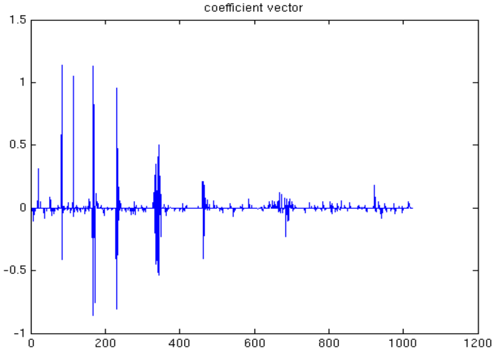

Figure: The coefficient vector

Wavelet Dictionary Results Signal 2

Algorithm	Runtime (s)
Soft Thresholding	0.021567
Hard Thresholding	max Iter
SSN	∞
FSS	2.305259
Elastic Net SSN	0.392921
Elastic Net FSS	2.449922
GPSR	0.162870
Interior Point	14.193927

Table: Algorithm running times on Wavelet Operator, Test Signal 2

Learned Dictionary

- Trained from 1600 signals of single damped sinusoids

Learned Dictionary

- Trained from 1600 signals of single damped sinusoids
- Frequencies, damping factors over range associated with capacitor switches

Learned Dictionary

- Trained from 1600 signals of single damped sinusoids
- Frequencies, damping factors over range associated with capacitor switches
- Starting points range over 0 to .1

Learned Dictionary

- Trained from 1600 signals of single damped sinusoids
- Frequencies, damping factors over range associated with capacitor switches
- Starting points range over 0 to .1
- Sized pared down to 354 from learning, much easier to deal with

Learned Dictionary Results

- The optimal alpha is .0495.
- The number of nonzeros is 13.

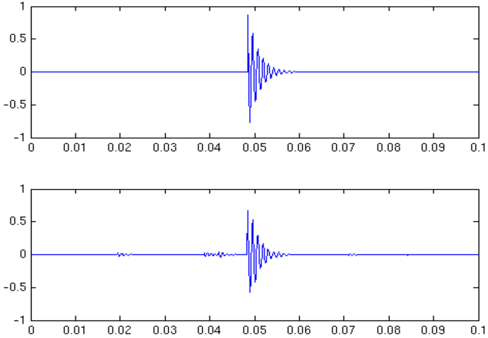

Figure: The reconstruction compared to the noiseless signal

Learned Dictionary Results

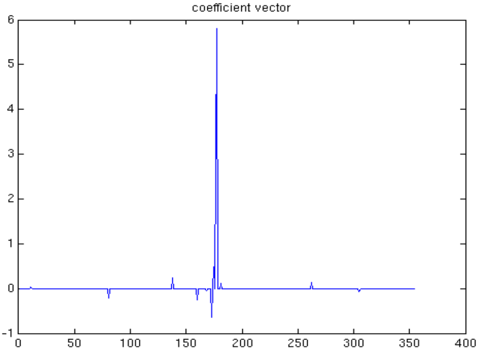

Figure: The coefficient vector

Learned Dictionary Results

Algorithm	Runtime (s)
Soft Thresholding	0.633864
Hard Thresholding	max iter
SSN	∞
FSS	0.078537
Elastic Net SSN	0.089418
Elastic Net FSS	0.078365
GPSR	0.099766
Interior Point	7.790935

Table: Algorithm running times on Learned Basis Dictionary, Test Signal 1

Learned Dictionary Results Signal 2

- The optimal alpha is alpha=5.6000e-06
- The number of nonzeros is 249.

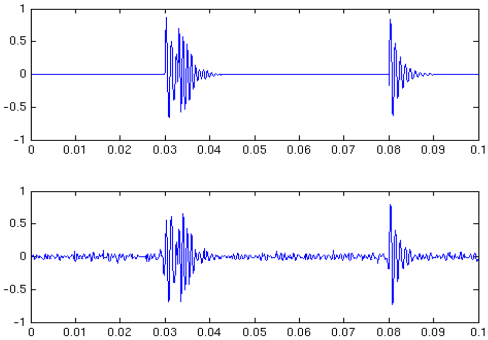

Figure: The reconstruction compared to the noiseless signal

Learned Dictionary Results Signal 2

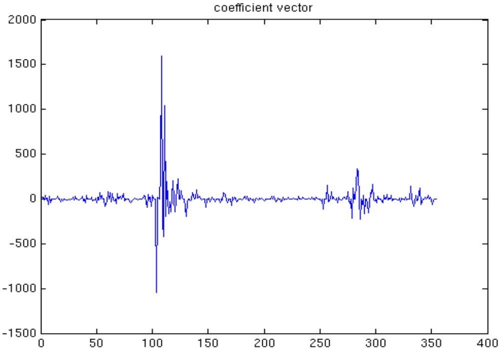

Figure: The coefficient vector

Learned Dictionary Results Signal 2

Algorithm	Runtime (s)
Soft Thresholding	max Iter
Hard Thresholding	max Iter
SSN	∞
FSS	7.975966
Elastic Net SSN	0.348048
Elastic Net FSS	5.441613
GPSR	max Iter
Interior Point	73.504971

Table: Algorithm running times on Learned B, Test Signal 2

This time the elastic net beta parameter had an effect resulting a solution of 322 nonzeros, but improved running time. beta taken as 1e-6.

Conclusions - Questions - Thoughts

- Sparsity Constraint research is approached from many angles. It is hard to say any setting is better than any other setting, or any algorithm is the best. The answer is always, it depends.

Conclusions - Questions - Thoughts

- Sparsity Constraint research is approached from many angles. It is hard to say any setting is better than any other setting, or any algorithm is the best. The answer is always, it depends.
- Overcomplete basis are particularly left out of a lot of the theoretical framework by the injective operator assumptions. Have to rely on empirical evidence.

Conclusions - Questions - Thoughts

- Sparsity Constraint research is approached from many angles. It is hard to say any setting is better than any other setting, or any algorithm is the best. The answer is always, it depends.
- Overcomplete basis are particularly left out of a lot of the theoretical framework by the injective operator assumptions. Have to rely on empirical evidence.
- How to find the best alpha value? How to find the best alpha and beta in Elastic Net?

Conclusions - Questions - Thoughts

- Sparsity Constraint research is approached from many angles. It is hard to say any setting is better than any other setting, or any algorithm is the best. The answer is always, it depends.
- Overcomplete basis are particularly left out of a lot of the theoretical framework by the injective operator assumptions. Have to rely on empirical evidence.
- How to find the best alpha value? How to find the best alpha and beta in Elastic Net?
- Elastic Net helps SSN converge on a solution - particularly in the messy last case

Conclusions - Questions - Thoughts

- Sparsity Constraint research is approached from many angles. It is hard to say any setting is better than any other setting, or any algorithm is the best. The answer is always, it depends.
- Overcomplete basis are particularly left out of a lot of the theoretical framework by the injective operator assumptions. Have to rely on empirical evidence.
- How to find the best alpha value? How to find the best alpha and beta in Elastic Net?
- Elastic Net helps SSN converge on a solution - particularly in the messy last case
- The learned dictionary is very sparse when close to the training signals, not so much otherwise

Conclusions - Questions - Thoughts

Conclusions - Questions - Thoughts

- The wavelet dictionary is more adaptable to different signals, but does not have as high of an underlying sparsity

Conclusions - Questions - Thoughts

- The wavelet dictionary is more adaptable to different signals, but does not have as high of an underlying sparsity
- When there are a small number of nonzero elements FSS is appropriate.

Conclusions - Questions - Thoughts

- The wavelet dictionary is more adaptable to different signals, but does not have as high of an underlying sparsity
- When there are a small number of nonzero elements FSS is appropriate.
- GPSR and Soft Thresholding preform very well with the wavelet operator.

Conclusions - Questions - Thoughts

- The wavelet dictionary is more adaptable to different signals, but does not have as high of an underlying sparsity
- When there are a small number of nonzero elements FSS is appropriate.
- GPSR and Soft Thresholding preform very well with the wavelet operator.
- Most algorithms are faster than the one I was using at Clemson.

Conclusions - Questions - Thoughts

- The wavelet dictionary is more adaptable to different signals, but does not have as high of an underlying sparsity
- When there are a small number of nonzero elements FSS is appropriate.
- GPSR and Soft Thresholding preform very well with the wavelet operator.
- Most algorithms are faster than the one I was using at Clemson.
- How would OMP preform in this case?

Conclusions - Questions - Thoughts

- The wavelet dictionary is more adaptable to different signals, but does not have as high of an underlying sparsity
- When there are a small number of nonzero elements FSS is appropriate.
- GPSR and Soft Thresholding preform very well with the wavelet operator.
- Most algorithms are faster than the one I was using at Clemson.
- How would OMP preform in this case?
- How to classify detected transients?

Solver Developers

- Iterative Thresholding Methods - Dirk Lorenz
- FSS - Stefan Schiffler
- SNN - Stefan Schiffler
- Basis Learning - Klaus Steinhorst
- GPSR - Mario Figueiredo, Robert Nowak, Stephen Wright
- Interior Point Method - SparseLab
- Wavelet Operator - Sparco

Thanks

Thanks to those who wrote the code I plundered, to Stefan for helping, and to the audience for listening.