An example, with a variety of algorithms

\[
\min \| y - Ax \| + \lambda |x|_1
\]

This problem turns up often

- Sparse reconstruction

  - \( y \) is a signal
  - \( A \) is a dictionary
  - \( x \) is a sparse encoding
L reg linear regression

model \( y_i \) by \( \beta_i^T x + x_6 \),

the bias

solve

\[
\sum_i \left(y_i - (u_i x + x_6) \right)^2 + \lambda |x|
\]

(it is not usual to regularize the bias.
- you could estimate it separately)

\[
\|y - (Ax + x_6)\|^2 + \lambda |x|
\]

Now estimate \( x_6 \), separately, so

get \( \hat{y} \)

\[
\|\hat{y} - Ax\|^2 + \lambda |x|
\]
In both cases, the attraction is that the $L_1$ norm encourages sparsity (many zeros in $x$). One way to see this is notice that the penalty for small $x_i$ is large compared to $L_2$ — it's worth making small values zero.

**Compressed Sensing:**

Assume we have $m$ linear measurements of an unknown signal $\mathbf{x}$

\[ y_i = \phi_i \cdot \mathbf{x} + \text{noise} \]
Suppose we know that \( z \) is compressible, or has a sparse rep in a transform domain (with a dictionary) \( W \)

then we can recover \( z \) by solving

\[
\| \Phi z - y \|^2 + \lambda \| Wz \|_1
\]

Now assume that \( W \) is invertible
We can solve

\[
\min \|Ax - y\|^2 + \lambda \|x\|_1
\]

where \( Wx = z \)

\[A W = \Phi\]

\( A \) is the dictionary
This problem is very different from

\[ \| y - Ax \|^2 + \lambda \| x \|^2 \] 2-norm.

2-norm problem:

\[(A^TA + \lambda I)x = A^Ty\]

\[\Rightarrow \text{linear!}\]

\[1\text{ norm:} \]

not linear

2 norm:

\[\lim_{\lambda \to 0} x = (A^TA)^{-1}A^Ty\]  
\[= (AA)^{-1}A^Ty\]

\[+ \Rightarrow \text{Moore-Penrose pseudo inverse}\]
2-norm: \( \lambda \to \infty \) gives \( x \to 0 \)

1-norm: \( x \to \infty \) gives

\[ x = 0 \text{ for values of } \lambda > \lambda_{\text{max}} \]

where \( \lambda_{\text{max}} = \| 2A^T y \|_{\infty} \)

\( \| u \|_{\infty} = \max_i |u_i| \) (inf norm)

2-norm

\( x(\lambda) \) is a curve

(rational, algebraic, some
mildly interesting geometry)

1-norm

\( x(\lambda) \) is piecewise linear
Some transformations of this problem

\[
\min \| Ax - y \|^2 + \lambda \| x \|_1
\]

can be turned into a quadratic program.

\[
\min \| Ax - y \|^2 + \lambda \sum_i t_i
\]

s.t.

\[
-x_i - t_i \leq 0
\]

\[
x_i - t_i \leq 0
\]

Notice this might be worrying — the

objective is

\[
x^T A^T A x - 2y^T A x + y^T y
\]

This could be sense
Consider
\[
\min \left[ f(x) + \lambda g(x) \right]
\]
and

\[
\min f(x)
\]

\( s + g(x) \leq \mu \).

Lagrangian for (2):

\[
f(x) + \lambda (g(x) - \mu)
\]

KKT

\[
\nabla f + \lambda \nabla g
\]

\( \lambda > 0 \)

\( g\lambda = 0 \)

\( g(x) - \mu = 0 \)

exclude the different values of \( \mu \) -> values of \( \lambda \geq 0 \)
So for any $\lambda \in \mathbb{R}$, I can choose a $\mu \in \mathbb{R}$ so that I get the same solution.

To proceed, we need a richer notion of gradient, to deal with the absolute value.

---

**Example**

Convex fn of one par

Over here, there is a cone of tangents. I can construct unique tangent lines.
Subgradients

\[ f(x) \]

\[ x \quad \text{derivative discontinuity} \]

Consider the graph of a convex function \((x, f(x))\).

- At a differentiable point, there is a tangent plane.

Equation is easy.

Surface \( g(x) = x_n - f(x_{i:n}) = 0 \)

Normal \( \nabla q = (- \nabla f, 1) \)

(This isn't unit - doesn't matter)
so the tangent plane at 

\((u, f(u))\)

is 

\[-\nabla f \cdot x_{i:n} + x_n - (\nabla f \cdot u + f(u)) = 0\]

Now think about a non-differentiable point.

- there is a cone of normals

\[\frac{d}{dx} \sin x\]

any line in this seesaw is tangent

so any vector in this range is tangent

family of normals - 2D cone
Any normal in this filled cone is normal.

All this means that at such points, our function has a family of derivatives — known as subgradients. Because a tangent plane (= normal) yields a derivative

\[ TP \quad \text{Normal} \quad \text{SubGrad} \]

\[ -p \cdot x + x_n - \alpha = 0 \quad (-p, 1) \quad p. \]
This yields an easy construction for subgradient of abs

\[ \partial |x| = \begin{cases} 
-1 & x < 0 \\
1 & x > 0 \\
[-1, 1] & x = 0 
\end{cases} \]

Subgradient because any derivation of derivative condition was geometric, it works for subgradients

So for \( f(x) \) to be a min, must have

\[ 0 \in \partial f \]

\( \uparrow \) because subgradients produce intervals, or worse
So, for our problem

\[ \mathbf{0} \in \mathcal{D}[\|Ax-y\|^2 + \lambda \|x\|_1] \]

\[ = 2A^T(Ax-y) + \lambda \nabla \|x\|_1 \]

\[ = 2A^T(Ax-y) + \lambda \nabla s \]

\[ \text{where } s_i = \begin{cases} 1 & x_i > 0 \\ -1 & x_i < 0 \\ [-1,1] & x_i = 0 \end{cases} \]

\[ s \text{ is the sign vector.} \]

Notice if you know the sign vector

for a solution, then the solution is easy to get.
Write \( J_c(s) = \{ i \mid s_i \neq [-1, 1] \} \)

\[ J_c(s) = \{ \text{All indices} \} - J(s) \]

Assume we know \( s \).

Then \( x \overset{J_c(s)}{=} 0 \)

so we have

\[
A_J^T(A_J x - y_J) + \lambda s_J = 0
\]

which is a straightforward linear system (we assume that \( A_J^T A_J \) has full rank for any \( J \) we deal with — fairly reasonable.)
Now, assume

\[ \lambda_1 < \lambda_2, \quad s(\lambda_1) = s(\lambda_2) = 0 \]

1) \[ \frac{x^t(\lambda_1)}{J(\lambda)} = \frac{x^t(\lambda_2)}{J(\lambda)} = 0 \quad \text{(obvious)} \]

2) \[ s(t\lambda_1 + (1-t)\lambda_2) = 0 \quad \text{(obvious)} \]

3) \[
\begin{aligned}
\mathbf{A}^T & \left( \mathbf{A} \left[ t \frac{x^t(\lambda_1)}{J(\lambda)} + (1-t) \frac{x^t(\lambda_2)}{J(\lambda)} \right] - \mathbf{y} \right) \\
& + \left( t\lambda_1 + (1-t)\lambda_2 \right) \sigma = 0 \\
& \text{(easy)}
\end{aligned}
\]

But this means

\[ x(\lambda) \text{ is piecewise linear} \]
So it is natural to try and construct this path. — but there is some bad news.

write \( d \) for dimension of \( x \).

then, clearly, \( \# \) of verts in path is \( \leq 3^d \)

actually, the upper bound is \( \frac{(3^d+1)}{2} \), and it can be attained (see papers).
Algorithms

1) Making pursuit

\[
\begin{align*}
  r_i &= y_i; \quad x_0 = 0 \\
  - \text{choose } \mathbf{x}_j \text{ of } \mathbf{A} \text{ that has largest value of } r_i^T \mathbf{A}_j \\
  - x_{i+1} &= x_i + (r_i^T \mathbf{A}_j) \cdot \frac{r_i}{(\mathbf{A}_j^T \mathbf{A}_j)} \cdot e_j \\
  - \tilde{r}_{i+1} &= \tilde{r}_i - (r_i^T \mathbf{A}_j) \cdot \frac{\mathbf{A}_j}{(\mathbf{A}_j^T \mathbf{A}_j)} \\
\end{align*}
\]

Notice

- residual always gets smaller
- \(|x|_1 \) gets bigger
Orthogonal matching pursuit

- like matching pursuit, BUT
  - readjust all non-zero coeffs each time you insert a column to get best fit in that space.
  - better estimate in k steps, but each step takes more work.

Homotopy algorithms
- a whole class of algorithm, constructing approx or exact $x(\lambda)$. 
Notice

1) we know \( x(0) \)

(because \( A^T (Ax - y) = 0 \), linear alg)

2) for sufficiently large \( \lambda = \lambda_{\text{max}} \)

\[ x(\lambda_{\text{max}}) = 0 \]

- and we can compute \( \lambda_{\text{max}} \)

\[ 0 \in A^T (Ax - y) + \lambda_{\text{max}} \begin{bmatrix} -1; 1 \\ \vdots \\ -1; 1 \end{bmatrix} \]

and \( x = 0 \)

So we must have

\[ 0 \in -A^T y + \lambda_{\text{max}} \begin{bmatrix} -1; 1 \\ \vdots \\ -1; 1 \end{bmatrix} \]

So \( \lambda_{\text{max}} = \|A^T y\|_\infty \)

\( \| \) largest abs. value.
So we can do

\[ \lambda_0 = 0 \quad \& \quad A^T A x_0 = A^T y \]

- Tangent to path:

\[ (A^T A) \frac{d x_i}{d \lambda} + \sigma(\lambda) = 0 \]

- Search along tangent for \( i \) that gives first sign change
- Update \( x \)
- Stop when \( \lambda = \lambda_{\text{max}} \)

**Advantage**

- Complete path

**Disadvantages:**

1) What if 2 knots coincide (rare)
2) Too many knots (approximations are available)
Another view of the PL path.

Problem

\[ \min \|Ax - y\|^2 + \lambda \sum ti \]

\[ \text{st} \quad -t_i \leq x_i \leq t_i \]

Notice this is a family of QPs

-linear constraints, but the polytope doesn't change when \( \lambda \) changes

- By inspection, at soln \( x_i = \{ t_i \} \)

- By inspection, these are 1-faces or 0-faces

\[ \Rightarrow \text{soln is always on 1-face or 0-face} \]
But look at SVM (linear)

\[ \lambda \frac{w^T w}{2} + \frac{1}{N} \sum_i \xi_i \]

st

\[ y_i(w^T x_i + b) \geq 1 - \xi_i \]

\[ \xi_i \geq 0 \]

1) as \( \lambda \) changes, polytope does not change.

2) at soln, we always have

\[ \xi_i = \max(0, 1 - y_i (w^T x_i + b)) \]

So a soln is always on a 0-face or 1-face

3) \( \Rightarrow \) homotopy path is Ph. !

(and fairly straightforward to construct)