Matching Pursuit

Relatively simple algorithm. Iteratively choose columns from Φ, keeping track of the approximation and the residual. At each step, choose the vector which is most closely correlated with the residual.

Algorithm:

Given signal $f \in \mathbb{R}^n$, dictionary Φ $(n \times p)$

① Set $\underbrace{f_0}_{\text{approximation}} = 0$, $\underbrace{R_0}_{\text{residual}} = f$, $k = 0$

② Select γ_k such that $|\langle R_k, \phi_{\gamma_k} \rangle|$ is maximized

$$\gamma_k = \arg\max_{1 \le m \le p} |\langle R_k, \phi_m \rangle|$$

③ Set

$$f_{k+1} = f_k + \langle R_k, \phi_{\gamma_k} \rangle \phi_{\gamma_k}$$

$$R_{k+1} = R_k - \langle R_k, \phi_{\gamma_k} \rangle \phi_{\gamma_k}$$

④ Repeat 2-3 until $\|f - f_k\|_2 \le \underbrace{\epsilon}_{\text{user specified}}$ and/or we have the desired # of terms.

Note that at each step

$$R_{k+1} \perp \phi_{\gamma_k}$$

since

$$\langle R_{k+1}, \phi_{\gamma_k} \rangle = \langle R_k, \phi_{\gamma_k} \rangle - \langle R_k, \phi_{\gamma_k} \rangle \overbrace{\cancel{\langle \phi_{\gamma_k}, \phi_{\gamma_k} \rangle}}^{1}$$

$$= 0$$

Thus we can write

$$\|f\|_2^2 = |\langle f, \phi_{\gamma_0}\rangle|^2 + \|R_1\|_2^2$$

$$= |\langle f, \phi_{\gamma_0}\rangle|^2 + |\langle R_1, \phi_{\gamma_1}\rangle|^2 + \|R_2\|_2^2$$

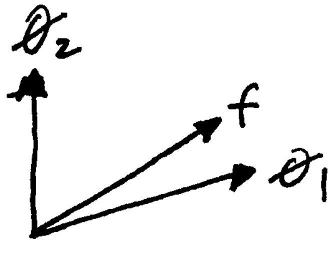
$$\vdots$$

$$= \sum_{k=0}^{M-1} |\langle R_m, \phi_{\gamma_k}\rangle|^2 + \|R_M\|_2^2$$

At each step, we are selecting the ϕ_{γ_k} which makes the residual error R_{k+1} as small as possible.

<u>Note:</u> The <u>same</u> column vector can be selected <u>multiple</u> times

<u>Ex:</u>



But the error $\|R_k\|_2 \to 0$ as $k \to \infty$.

In fact:

<u>Thm</u>: There exists $2 > 0$ s.t. $\forall\ K \geq 0$

$$\|R_K\|_2 \leq 2^{-2K} \cdot \|f\|_2$$

(see Mallat, Theorem 9.10)

In general, MP takes an ∞ number of steps to converge, but the error $\to 0$ exponentially.

<u>Orthogonal Matching Pursuit</u>

Instead of keeping track of the selected vectors

$$\phi_{\gamma_0}, \phi_{\gamma_1}, \ldots, \phi_{\gamma_K}$$

OMP keep track of a projection onto an <u>orthogonal</u> <u>subspace</u> spanned by vectors we have selected so far.

Each step adds a new dimension

$\Rightarrow$ OMP converges in $\leq n$ steps.

However, OMP is more expensive, since we have to run Gram-Schmidt at every iteration.

Algorithm (OMP):

Given signal $f \in \mathbb{R}^n$, $n \times p$ dictionary Φ

① Set $f_0 = 0$, $R_0 = 0$

② Select γ_k as before

$$\gamma_k = \arg\max_{1 \le m \le p} |\langle R_k, \varphi_m \rangle|$$

③ Set

$$u_k = \varphi_{\gamma_k} - \sum_{m=0}^{k-1} \frac{\langle \varphi_{\gamma_k}, u_m \rangle}{\|u_m\|_2^2} \cdot u_m$$

$$f_{k+1} = f_k + \langle R_k, u_k \rangle u_k$$

$$R_{k+1} = R_k - \langle R_k, u_k \rangle u_k$$

Note that at each k

$$f = \underbrace{P_{V_k} f}_{\substack{\text{projection of} \\ f \text{ onto span}\{u_0, \dots, u_k\}}} + \underbrace{R_k}_{\text{residual}} \qquad R_k \perp P_{V_k} f$$

This avoids redundancies in the selection of the γ_k.

Basis Pursuit

We have a dictionary Φ ($n \times p$), and we want to find a *sparse* decomposition α

$$f = \Phi\alpha$$

for a given signal f.

Since $\Phi = \begin{array}{c|c} & \overset{p}{\longrightarrow} \\ n\downarrow & \begin{bmatrix} | & | & | & | & | & | & | & \end{bmatrix} \end{array}$ is rectangular, there are *many* ways to decompose f. Out of all the valid decompositions

$$\{\beta : \Phi\beta = f\} \subset \mathbb{R}^p$$

We want the one that uses the smallest number of terms, i.e. we'd like to solve

$$\min \#\{\gamma : \beta(\gamma) \neq 0\} \quad \text{subject to } \Phi\beta = f$$

We'll call the number of non-zero terms in a vector the "ℓ_0 norm" (though it is not a norm at all)

$$\min \|\beta\|_{\ell_0} \quad \text{s.t. } \Phi\beta = f$$

The problem is easy enough to state, but is basically impossible to solve in general (it is a combinatorial optimization problem).

<u>Basis Pursuit</u> (BP) replaces the ℓ_0 norm with ℓ_1, which also tends to give sparse solutions:

$$(\text{BP}) \qquad \min \|\beta\|_{\ell_1} \quad \text{s.t.} \quad \Phi\beta = f$$

This program is convex, and can be solved in a reasonable amount of time.

<u>BP as a linear program</u>

We can recast BP as a linear program using a simple trick. We are interested in

$$\min_{\beta} \sum_{\gamma=1}^{p} |\beta(\gamma)| \quad \text{s.t.} \quad \Phi\beta = f$$

← nonlinear functional

← linear constraints

The functional is piecewise linear.

Introduce an auxilary variable $u \in \mathbb{R}^p$, and consider

$$\min_{u,\beta} \sum_{\gamma=1}^{p} u(\gamma) \quad \text{s.t.} \quad -u(\gamma) \leq \beta(\gamma) \leq u(\gamma) \quad \forall \gamma = 1, \ldots, p$$
$$\Phi\beta = f$$

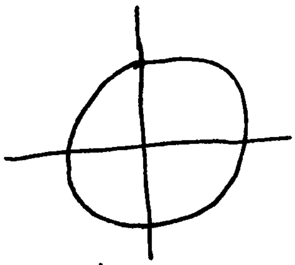
This is a linear program (linear functional and linear constraints)

It is not hard to see that the solutions of the linear program and of BP will be the same.

$\Rightarrow$ $\boxed{\text{Basis Pursuit = Linear Programming}}$

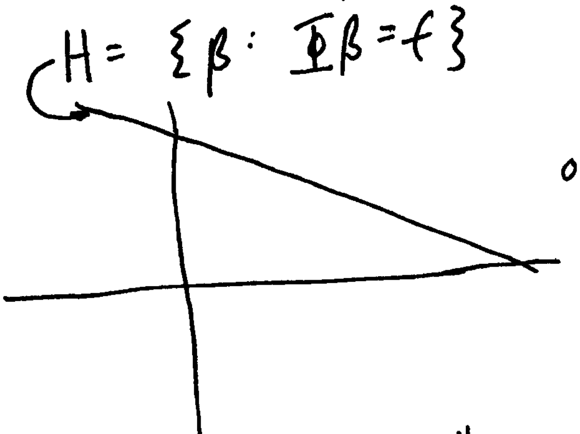
The ℓ_1-norm and sparsity

The ℓ_1-norm intrinsically favors sparse signals.
To see this, consider all the points on the unit sphere in $\mathbb{R}^2$

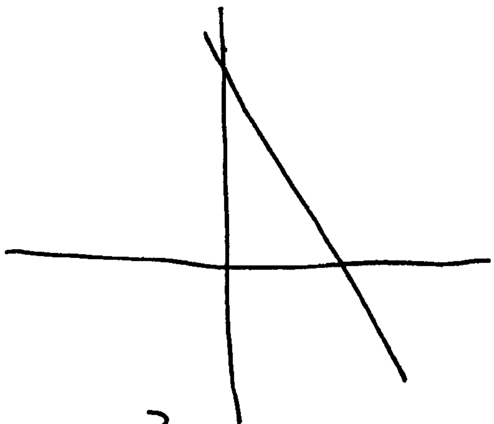


which have smallest ℓ_1-norm?

Now consider all the points on the hyperplane



or



which point on H has smallest ℓ_1-norm?

10

Efficiency of Basis Pursuit

In fact, we can show that given an $n \times p$ matrix Φ and a vector $y \in \mathbb{R}^n$, the solution to

$$\min_{\beta \in \mathbb{R}^p} \|\beta\|_1 \quad \text{subject to} \quad \Phi\beta = y$$

never has more than n non-zero terms. So in a very concrete way, BP is selecting a *basis* (set of n linearly independent vectors) from the dictionary Φ.

We prove this as follows. Let α be a feasible vector ($\Phi\alpha = y$) supported on $\Gamma \subset \{1, 2, \ldots, p\}$ — that is, Γ contains the indices of the non-zero locations in α. We will use s as the size of Γ:

$$s = |\Gamma| = \#\text{non-zero terms in } x.$$

We will show that if $s > n$, then there necessarily exists a vector with $s - 1$ non-zeros terms that is also feasible.

Let Φ_Γ be the $n \times s$ matrix containing the s columns of Φ indexed by Γ. Let $z \in \mathbb{R}^s$ be a vector containing the signs of x on Γ

$$z = \{\text{sgn}(x[\gamma]),\ \gamma \in \Gamma\}$$

and let $z' \in \mathbb{R}^p$ be the extension of z

$$z'[\gamma] = \begin{cases} \text{sgn}(x[\gamma]) & \gamma \in \Gamma \\ 0 & \gamma \notin \Gamma \end{cases}$$

Suppose $s > n$. Then Φ_Γ has more columns than rows, and hence has a non-trivial null space. Let $h \in \text{Null}(\Phi_\Gamma)$ be any vector in this null space, and let h' be the extended version of h:

$$h'[\gamma] = \begin{cases} h[\gamma] & \gamma \in \Gamma \\ 0 & \gamma \notin \Gamma \end{cases}.$$

It should be clear that $h' \in \text{Null}(\Phi)$. We may assume without loss of generality that

$$\langle h, z \rangle \leq 0,$$

since otherwise we could just use $-h$ (since of course if $h \in \text{Null}(\Phi_\Gamma)$, then $-h \in \text{Null}(\Phi_\Gamma)$). For ϵ small enough, we will have

$$\text{sgn}(x + \epsilon h) = \text{sgn}(x) = z,$$

and so

$$\begin{aligned}
\|x + \epsilon h'\|_1 &= \sum_{\gamma \in \Gamma} \text{sgn}(x[\gamma] + \epsilon h[\gamma])(x[\gamma] + h[\gamma]) \\
&= \sum_{\gamma \in \Gamma} z[\gamma] x[\gamma] + \epsilon \sum_{\gamma \in \Gamma} z[\gamma] h[\gamma] \\
&= \|x\|_1 + \epsilon \langle z, h \rangle \\
&\leq \|x\|_1
\end{aligned}$$

since $\langle z, h \rangle \leq 0$.

We can move in the direction h lowering the ℓ_1 norm until $\text{sgn}(x[\gamma] + \epsilon h[\gamma]) \neq \text{sgn}(x[\gamma])])$. This happens exactly when one of the elements of $x + \epsilon h$ is driven exactly to zero; call the corresponding stepsize ϵ_0. By construction

$$x_1 = x + \epsilon_0 h',$$

is feasible, has smaller ℓ_1 norm than x, and has one fewer non-zero element than x.

The entire argument above relied on finding a non-zero vector in the null space of Φ_Γ. We are guaranteed that such a vector exists if $s > n$, so the argument above implies that the solution (or at least one of the solutions if the solutions if there is more than one) to basis pursuit has at most n non-zero terms.