Probabilistic Matching Pursuit for Compressive Sensing

Atul Divekar, Okan Ersoy,
School of Electrical and Computer Engineering,
465 Northwestern Ave.,
Purdue University,
West Lafayette, IN 47907-2035

May 6, 2010

Contents

Abstract

Compressive sensing investigates the recovery of a signal that can be sparsely represented in an orthonormal basis or overcomplete dictionary given a small number of linear combinations of the signal. We present a novel matching pursuit algorithm that uses the measurements to probabilistically select a subset of bases that is likely to contain the true bases constituting the signal. The algorithm is successful in recovering the original signal in cases where deterministic matching pursuit algorithms fail. We also show that exact recovery is possible when the number of nonzero coefficients is upto one less than the number of measurements. This overturns a previously held assumption in compressive sensing research.

Chapter 1

Introduction

Consider an underdetermined system $y = \Phi c$ where $\Phi \in R^{MXN}$ with $M < N$, c is a N-dimensional signal and y is a length M set of measurements equal to linear combinations of Φ. Suppose that c has S nonzero elements, and we wish to recover c from y. One possible technique is to consider every subset Φ_I of $|I| = S$ columns drawn from Φ and test whether it fits y by least squares leaving no residue. However this requires testing of $C(N, S)$ subsets, which is infeasible for even moderate values of N and S.

Recent papers [1,2] show that if c has S nonzero elements with $S \leq \frac{M}{2}$ and the matrix Φ satisfies some additional conditions, then c can be recovered either exactly or with a small approximation error. For example, it is shown in [2] that if matrix Φ satisfies a *Restricted Isometry Property*(RIP), then l_1 minimization can recover the vector c. Explicitly, the matrix Φ satisfies the RIP with parameters (m, δ) for $\delta \in (0, 1)$ if

$$(1 - \delta)||c||_2^2 \leq ||\Phi_I c||_2^2 \leq (1 + \delta)||c||_2^2 \tag{1.1}$$

for every size m subset I of columns of Φ. If Φ satisfies the RIP with $m = 2S$ and $\delta < \sqrt{2} - 1$, then c can be recovered perfectly by solving

$$min\, ||c||_1 \text{ such that } y = \Phi c \tag{1.2}$$

If c is not exactly sparse, but the components decay rapidly in magnitude, then c can be approximately recovered with a distortion that is bounded by

$$||c^* - c||_{l_2} \leq C_0 s^{-\frac{1}{2}} ||c - c_s||_{l_1} \tag{1.3}$$

The linear program in (1.2) is a convex optimization problem that can be solved by interior point methods. However it is difficult to prove that a matrix Φ satisfies the RIP, and for large signals the convex optimization can still be computationally slow.

A second approach to this problem involves greedy algorithms such as Orthogonal Matching Pursuit (OMP) [3] and its variants [4, 5]. In these algorithms, the projection $z = \Phi^T y$ of the data is used to identify a single or a few bases that is/are believed to be in the true signal, and then the component of the data y that is spanned by all the bases selected so far

is removed, leaving behind a residue r that is orthogonal to the bases selected. The residue is then used to identify more bases using $z = \Phi^T r$.

The OMP algorithm [3] provides a weak result: If c is S-sparse and $y = \Phi c$ is known with Φ a $M * N$ sampling matrix consisting of zero mean normal random variables with equal variances, OMP recovers c in S iterations except with probability N^{-1}. Failure cases are discussed in [6]. The CoSaMP algorithm [5] provides exact recovery for an S-sparse signal by Matching Pursuit provided that the RIP constants $\delta_{2S} < \delta_{4S} \leq 0.1$. This implies a relatively small range of eigenvalues $(1 - \delta_{2S}, 1 + \delta_{2S})$ allowed for each $2S$ column subset of Φ, and verifying that Φ satisfies the RIP is also computationally difficult. In general, deterministic Matching Pursuit(MP) algorithms suffer from an important weakness: it is possible to construct signals $y = \Phi c$ for which the MP algorithm makes a wrong choice for a basis believed to be in the original signal, removes this basis from the samples, and then is led astray in making future choices.

The literature also contains compressive sensing recovery applications where the recovery works very well, even though the Φ matrix contains highly correlated columns which do not satisfy any reasonable bound on the RIP constants for even small values of S. An example is the face recognition work in [7] where a dictionary contains highly similar faces and recognition is successfully carried out by l_1 minimization. In this work the class of faces that contains most of the resultant weights, is returned as the identifying solution. Indeed, Restricted Isometry is a sufficient, but not necessary, condition for compressive sensing recovery.

We present below an algorithm, Probabilistic Matching Pursuit, that selects size K subsets of the columns of Φ by examining the probability that a base might be a component of the true signal x. Each subset Φ_J, where J is a size K index set, is selected to have a significant probability of containing the S columns that truly make up signal x. If this is the case, the residue $y - \Phi_J(\Phi_J^T \Phi_J)^{-1} \Phi_J^T y$ has zero energy, providing a simple test for whether the true components have been found. Our algorithm may be regarded as exploring a previously overlooked middle ground in compressive sensing: Recall that directly checking all $C(N, S)$ subsets of size S columns of Φ is infeasible. At the other extreme, OMP and its variants sequentially find a single set of bases that is proposed as the solution. Our algorithm tests a reasonable(i.e. non-exponential in problem size) number of likely solutions, much smaller than $C(N, S)$, but more than the single choice of OMP.

The maximum number of bases selected in our algorithm is K. The larger K is, the greater the probability that all the true bases will be present. We assume that every set of $K' \leq M$ bases in Φ are linearly independent. Then for the residue test described above to work, we need $K \leq K' - 1$, so that if the chosen set J of size K does not include a base Φ_i that is present in the sample vector y, $\Phi_{J \cup \{i\}}$ is still linearly independent, and the least squares fit will leave a non-zero residue. In the sequel, let $K_m = K' - 1$.

We note that a probabilistic approach to matching pursuit was previously suggested in [8] in the context of denoising. There the denoised signal is found as a weighed sum of expansions, with the weights for each expansion depending on the correlation between the noisy samples and the bases involved in the expansion. Although it is named similarly to the

```
// Input: M by N matrix Φ, sample vector y = Φc
// Output: A K' − 1 sparse approximation u of c
for n = 1 to MaxIter do
    J ← ∅, R ← {1..N}, r ← y
    k ← 1
    while k ≤ K_m and ||r|| > 0 do
        z ← Φ_R^T r.
        For each i ∈ R, assign p_i.
        Select base j from R according to p_i.
        J ← J ∪ {j}, R ← R − {j}
        ĉ_J ← (Φ_J^T Φ_J)^{-1} Φ_J^T y.
        r ← y − Φ_J ĉ_J
        k ← k + 1
    end while
    if ||r|| = 0 then
        u ← 0
        u_i ← ĉ_i for i ∈ J
        return u
    end if
end for
return No solution found
```

Figure 1.1: Probabilistic Matching Pursuit

algorithm we propose here, our algorithm was developed independently and is completely different from that work. It finds the single set of bases actually present in the signal, as opposed to a weighed sum. The noise in our case results from the cross correlation between true signal components and false ones.

In Chapter 2 we describe the Probabilistic Matching Pursuit algorithm and analyze the number of iterations needed to recover the true components in Chapter 3. In Section 4 we show that this algorithm can recover the true bases even if $2S > M$ and Chapter 5 describe our experimental results.

Chapter 2

Probabilistic Matching Pursuit

The algorithm is listed in Figure 1.1. It generates several (upto MaxIter) subsets of bases and tests whether each contains all the bases present in the original signal. If all the bases have been selected, the least squares residue is zero and the algorithm terminates. Let R be the set of bases not yet selected for inclusion in the basis set J. To generate each subset of bases, we maintain a residual r at each iteration and use it to generate $z = \Phi_R^T r$. Let $A = \Phi_R^T \Phi_R - I$. Then we have $c_R = z_R - Ac_R$, or

$$c_i = z_i - \sum_{j \neq i; i,j \in R} a_{ij} c_j \tag{2.1}$$

Here $a_{ij} = \Phi_i^T \Phi_j$. At each iteration we consider only c_i from R. To each basis in R we assign a probability p_i, and select a base j from the bases in R according to probabilities p_i, that is, $j \leftarrow i$ with probability p_i, $\forall i \in R$. As in the standard OMP algorithm, we find the least square estimates for coefficients in J and then the residue r after removing the components corresponding to J. This process is repeated a maximum of K_m times. If the residue is found to be zero (assuming no rounding error or sensor noise), the true bases must be present in J, since we have assumed that every set of K' bases is linearly independent.

Since we have selected bases probabilistically, it is possible that some base Φ_i that is present in the true sample y may be excluded from J. In this case the least squares residue has significant energy. We repeat the inner loop upto MaxIter times in the hope of choosing all the correct bases atleast once.

The probabilities can be assigned in several ways. One approach stems from considering an estimate for each c_i from Equation(2.1). We model each c_i as a normal random variable with mean z_i and variance $\sigma_\eta^2 = \sigma_c^2 \sigma_a^2 (S - e)$. We then find p_i as the probability that c_i has the largest magnitude among all the coefficients. We have

$$p_i = P(|\hat{c}_i| > |\hat{c}_j|, \forall j \in R \setminus \{i\}) \tag{2.2}$$

$$= \int_{\hat{c}_i = -\infty}^{\infty} p_{\hat{c}_i}(\hat{c}_i) \prod_{j \neq i, j \in R} \int_{-|\hat{c}_i|}^{|\hat{c}_i|} p_{\hat{c}_j}(\hat{c}_j) d\hat{c}_j d\hat{c}_i \tag{2.3}$$

However we found that calculating p_i in this way is slow and in fact unnecessary.

Instead, we find the largest h magnitudes from the set $\{|z_i|, i \in R\}$. Let the index set corresponding to these h coefficients be L. Then we set

$$p_i = \begin{cases} \frac{1-\epsilon}{h} & \text{for } i \in L \\ \frac{\epsilon}{|R|-h} & \text{for } i \notin L \end{cases} \tag{2.4}$$

Here ϵ is a small positive number independent of the problem size. We used $\epsilon = 0.01$ and $h = 8$. The intuition is that as long as all the bases actually present in the signal are selected with some constant probability, they will certainly be discovered in a finite number of iterations. In our analysis we use this simplified probability assignment.

Chapter 3

Analysis of Complexity

We analyze the number of iterations needed by the algorithm to find the true nonzero components of the signal. We assume that each column of Φ is unit norm, and let $a_{i,j} = \Phi_i^T \Phi_j$. Let t_i, $1 \leq i \leq S$, be the indices of the nonzero locations of the signal c. We assume that the coefficient magnitudes are in decreasing order, i.e. $|c_{t_1}| > |c_{t_2}| > ..|c_{t_S}|$. Also, let I be the index set of all zero coefficients, so that $c_j = 0$, $\forall j \in I$.

Let p be the probability that all S bases are recovered in $K_m = K' - 1$ iterations of the inner loop. We wish to estimate p. Let i_j denote the index of the jth true (i.e. nonzero) coefficient recovered, for $1 \leq j \leq S$, and let k_j be the number of selections made by the inner loop after i_{j-1} is found upto and including the selection of i_j. Let $i_j(k_j)$ denote the event that k_j selections are made by the inner loop after i_{j-1} is found upto and including the selection of i_j.

Then we have

$$p = \sum_{(i_1, i_2..i_S)} \sum_{\sum_{i=1}^{S} k_i \leq K_m} P(i_1(k_1), i_2(k_2), ..., i_S(k_S)) \tag{3.1}$$

The first sum is over all possible orderings in which the true coefficients are found, and the second is over all possible $(k_1, k_2, ..k_S)$ sets with each $k_i > 0$. Let p_j be the probability of selecting a true coefficient after $j-1$ true coefficients have been selected. Let q_j be the probability of selecting a zero coefficient between the selection of the $(j-1)^{st}$ and j^{th} true coefficients.Since $N \gg S$, q_j is approximately constant over this range, and $q_j = 1 - p_j$.

Let $T = \sum_{i=1}^{S} K_i$. We have $p = P(T \leq K_m)$. Each k_j is a geometric random variable with mean $\frac{1}{p_j}$ and variance $\frac{1-p_j}{p_j^2}$. We can compute the distribution of T by a S-fold convolution of the distributions of k_i, or approximate T as a normal random variable with mean $\mu_T = \sum_{j=1}^{S} p_j$ and variance $\sigma_T^2 = \sum_{j=1}^{S} \frac{1-p_j}{p_j^2}$. Then $p = G(\frac{K_m - \mu_T}{\sigma_T})$, where G is the cumulative distribution function for a normal distribution.

Consider the situation between the selection of the $(j-1)^{st}$ and j^{th} true coefficients. Let the $S-j+1$ unselected bases which are components of the signal have indices $n_1, n_2...n_{S-j+1}$, in decreasing order of magnitude of the original coefficient values. Let $E(n_i, j)$ denote the event that $|z_{n_i}|$ is among the h largest $|z|$ values and S_{n_i} the event that n_i is selected by the

algorithm.

We have

$$p_j = \sum_{i=1}^{S-j+1} P(S_{n_i}) \tag{3.2}$$

where

$$\begin{aligned} P(S_{n_i}) &= P(S_{n_i}|E(n_i,j))P(E(n_i,j)) \\ &+ P(S_{n_i}|\overline{E(n_i,j)})P(\overline{E(n_i,j)}) \end{aligned} \tag{3.3}$$

Then

$$p_j \approx \sum_{i=1}^{S-j+1} \frac{(1-\epsilon)}{h} P(E(n_i,j)) \tag{3.4}$$

$$= \frac{(1-\epsilon)}{h} \sum_{i=1}^{S-j+1} P(E(n_i,j)) \tag{3.5}$$

In practical compressive sensing problems, the coefficients have a rapidly decaying magnitude. We model the S nonzero coefficients as having a Laplacian distribution $p_{c_i}(c) = \frac{1}{2\beta} e^{\frac{-|c|}{\beta}}$. In the algorithm the bases corresponding to the top h magnitudes are each selected with probability $\frac{1-\epsilon}{h}$. We assume that at intermediate stages of the algorithm, the distribution of the unselected bases remains Laplacian with the same β. Let $r = \frac{\beta}{\sigma_\eta(j)}$ where $\sigma_\eta^2(j)$ is the noise variance just before the jth true base is selected. We find by simulation that

$$\sum_{i=1}^{S-j+1} P(E(n_i,j)) \approx h(1-(1-\frac{S-j+1}{N})e^{-\delta r_j^2}) \tag{3.6}$$

Here $\delta \approx 4$ for $h = 8$. We can use this approximation to predict the probability of recovery.

3.1 Sources of noise in z

We discuss the different factors that produce noise in the projections z_i. Let J represent the index set of false bases(with $c_i = 0$) selected until a particular iteration of the algorithm, let I be the true bases also selected by then, and let I' be the true bases not yet selected by then. The residue at this stage is

$$r = y - \Phi_I \tilde{c_I} - \Phi_J \tilde{c_J} \tag{3.7}$$

where $\tilde{c_I}$ and $\tilde{c_J}$ are the least square estimates of the coefficients corresponding to Φ_I and Φ_J. Then the projection $z_i = \Phi_i^T r$ for each $i \in R$ is

$$z_i = c_i + \Phi_i^T \Phi_I (c_I - \tilde{c_I}) + \Phi_i^T \Phi_{I'} c_{I'} - \Phi_i^T \Phi_J \tilde{c_J} \tag{3.8}$$

The false bases Φ_J have two effects: they can distort the least square estimates so that $(c_I - \tilde{c_I})$ is not negligible, and they add additional distortion $-\Phi_i^T \Phi_J \tilde{c_J}$ to each z_i. This last term can reduce the magnitude of z_i corresponding to $c_i \neq 0$ and pull it out of the largest h $|z|$ values, and also increase the magnitude of z_i corresponding to $c_i = 0$.

Also, even if no false bases have been selected, $(c_I - \tilde{c_I})$ can be significant because of the correlation of the bases in Φ_I with those in $\Phi_{I'}$.

Chapter 4

Recovery when the number of nonzero components exceeds half the number of samples

A common belief in previous work [4, 5] is that perfect recovery is possible only if $M \geq 2S$. The argument made is the following: Suppose that $M < 2S$. Then each set of $2S$ columns from Φ is linearly dependent. Let $y = \Phi_1 c_1$, where c_1 is the length S coefficient vector that constitutes the true signal and Φ_1 be the corresponing S column subset of Φ. Let Φ_2 be another S column subset disjoint from Φ_1. Since $[\Phi_1 \Phi_2]$ is linearly dependent, there exists a length S vector c_2 such that $[\Phi_1 \Phi_2]c = 0$, where $c = [c_1^T c_2^T]^T$. Then $\Phi_1 c_1 + \Phi_2 c_2 = 0$, or $y = \Phi_1 c_1 = -\Phi_2 c_2$. Thus there exists a false solution c_2 that would make the residue $||r|| = 0$.

However, this is overly pessimistic. We show that the probability that such a false solution is found is negligible. Assume that every set of $K_m + 1$ bases is linearly independent, and that the algorithm selects upto K_m bases until $||r|| = 0$. At a particular stage of the algorithm, suppose that a subset of bases Φ_I from Φ_1 has been selected, and let $\Phi_{I'}$ represent the remaining bases from Φ_1. Let c_I and $c_{I'}$ represent the corresponding components of the true sparse vector c_1 that produces $y = \Phi_1 c_1$. Also suppose that a set Φ_J of bases disjoint from Φ_1 has also been selected. A false solution is found if there exists a vector v_1 such that $\Phi_{I'} c_{I'} = \Phi_J v_1$. This requires the columns of the matrix $A = [\Phi_{I'} \Phi_J]$ to be linearly dependent. If $|I'| + |J| \leq K_m + 1$, the columns of A are linearly independent and a false solution cannot exist.

Consider the situation where $|I'| + |J| \geq K_m + 2$. Let $N(A)$ denote the nullspace of A, and d_{N_A} the dimension of $N(A)$. Since the rank of A is $K_m + 1$, by the rank-nullity theorem, $d_{N_A} = |I'| + |J| - K_m - 1$.

Let $Z = \{v : \Phi_{I'} v \in \text{span}\{\Phi_J\}\}$ be the set of vectors v such that $\Phi_{I'} v$ is in the intersection of $\text{span}\{\Phi_{I'}\}$ and $\text{span}\{\Phi_J\}$. This is the subset of coefficients of $\Phi_{I'}$ for which a false solution can result because some linear combination of Φ_J can match $\Phi_{I'} v$. We have $\dim(\text{Z}) = \dim(\text{span}\{\Phi_{I'}\} \cap N(A)) = \min(|I'|, d_{N_A})$.

Since $|J| \leq K_m$, we have $d_{N_A} = |I'| + |J| - K_m - 1 \leq |I'| - 1$. Then $\dim(Z) < |I'|$, and

a false solution is limited to a subspace of $c_{I'}$ with dimension less than $|I'|$. The probability that $c_{I'}$ lies in this subspace is negligible. This implies that the probability of finding a false solution is negligible.

Note that this result does not depend on any particular method of selecting bases, either probabilistically or deterministically. As long as $|J| \leq K_m$, we have $\dim(Z) < |I'|$, and the set of $c_{I'}$ for which a false solution is obtained is small.

A simple intuitive example for perfect recovery is the following: Consider the constant magnitude coefficient case. Suppose that we have $z_k = \pm c + \eta$ when $c_k \neq 0$, and $z_k = \eta$ when $c_k = 0$. The variance $\sigma_\eta^2 \leq (S-1)\sigma_a^2 c^2$, and the signal to noise ratio is $\frac{c}{\sigma_\eta} \geq \frac{1}{\sigma_a\sqrt{S-1}}$. If the ratio is high enough, we have $|z_k| \approx c$ for $c_k \neq 0$ and $|z_k| \approx 0$ for $c_k = 0$. Then even if $2S > M$, it is easy to pick the true signal components just by observing the $|z_k|$ magnitudes.

Chapter 5

Experimental Results

We tested the performance of the algorithm for $y = \Phi c$ where Φ is a $M * N$ random matrix with unit norm columns. We constructed Φ from random column permutations of $[I|G]$, where I is the $M * M$ identity matrix and G a $M * (N - M)$ matrix consisting of $\pm\frac{1}{\sqrt{(M)}}$, with signs equally likely positive and negative. We found that this construction gives a lower variance σ_a^2 than a similar Gaussian random matrix.

In table 5.1 we show the results for different values of M, N and S. For each set of parameters we tested 50 different matrices Φ and found the number of outer iterations needed to find the true signal components with MaxIter=25 and $\beta = 10$. The Recovery Fraction column indicates the fraction of trials (out of 50) where the true components were perfectly recovered.

Table 5.1: Recovery results for Laplacian coefficients with $\beta = 10$

N	M	S	σ_a^2	Recovery Fraction	Mean Iterations
900	800	400	0.000263	1.0	1.01
900	800	450	0.000263	1.0	1.02
900	800	500	0.000263	1.0	1.46
900	800	550	0.000263	1.0	1.1
1000	800	400	0.000450	1.0	1.1
1000	800	450	0.000450	1.0	1.1
1000	800	500	0.000450	1.0	1.01
1100	800	400	0.000589	1.0	1.0
1100	800	450	0.000589	1.0	1.0
1100	800	500	0.000589	0.92	4.8

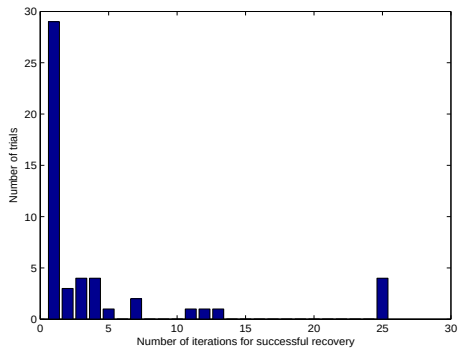

Figure 5.1: Histogram of number of iterations needed to recover all true bases over 50 trials for N=1100,M=800,S=500 and Laplacian coefficients with $\beta = 10$

Bibliography

[1] E. Candes and T. Tao, "Near optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Trans. Inf. Theory*, vol. 52, pp. 5406–5425, Dec. 2006.

[2] E. Candes, "The Restricted Isometry Property and its implications for compressed sensing," *Compte Rendus de l'Academie des Sciences, Paris, Series I*, pp. 589–592, Dec. 2008.

[3] J. Tropp and A. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. on Information Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.

[4] D. Donoho, Y. Tsaig, I.Drori, and J.L.Starck, "Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit," preprint.

[5] D. Needell and J. Tropp, "Cosamp: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, pp. 301–321, 2009.

[6] H.Rauhut, "On the impossibility of uniform sparse reconstruction using greedy methods," *Sampling Theory Signal and Image Processing*, 2008.

[7] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 31, no. 2, 2009, preprint: http://perception.csl.uiuc.edu/ jnwright/Wright09-PAMI.pdf.

[8] S. Ferrando, E. Doolittle, A.J.Bernal, and L. Bernal, "Probabilistic matching pursuit with gabor dictionaries," *Signal Processing*, vol. 80, pp. 2099–2120, 2000.