Error Correction via Linear Programming

Emmanuel Candes[†] and Terence Tao[♯]

† Applied and Computational Mathematics, Caltech, Pasadena, CA 91125

♯ Department of Mathematics, University of California, Los Angeles, CA 90095

March 2005

Abstract

Suppose we wish to transmit a vector $f \in \mathbf{R}^n$ reliably. A frequently discussed approach consists in encoding f with an m by n coding matrix A. Assume now that a fraction of the entries of Af are corrupted in a completely arbitrary fashion. We do not know which entries are affected nor do we know how they are affected. Is it possible to recover f exactly from the corrupted m-dimensional vector y?

This paper proves that under suitable conditions on the coding matrix A, the input f is the unique solution to the ℓ_1-minimization problem ($\|x\|_{\ell_1} := \sum_i |x_i|$)

$$\min_{g \in \mathbf{R}^n} \|y - Ag\|_{\ell_1}$$

provided that the fraction of corrupted entries is not too large, i.e. does not exceed some strictly positive constant ρ^* (numerical values for ρ^* are given). In other words, f can be recovered exactly by solving a simple convex optimization problem; in fact, a linear program. We report on numerical experiments suggesting that ℓ_1-minimization is amazingly effective; f is recovered exactly even in situations where a very significant fraction of the output is corrupted.

Keywords. Linear codes, decoding of (random) linear codes, sparse solutions to underdetermined systems, ℓ_1-minimization, linear programming, restricted orthonormality, Gaussian random matrices.

Acknowledgments. E. C. is partially supported in part by a National Science Foundation grant DMS 01-40698 (FRG) and by an Alfred P. Sloan Fellowship. T. T. is supported in part by a grant from the Packard Foundation. Many thanks to Rafail Ostrovsky for pointing out possible connections between our earlier work and the error correction problem.

This paper is in part an abridged version of the full companion unpublished report [4]; however, we introduce here a new proof of our main result yielding slightly sharper statements, and present new numerical evidence suggesting the wide applicability of our approach.

1 Introduction

1.1 The error correction problem

This paper considers the model problem of recovering an input vector $f \in \mathbf{R}^n$ from corrupted measurements $y = Af + e$. Here, A is an m by n matrix (we will assume throughout the paper that $m > n$), and e is an arbitrary and unknown vector of errors. The problem we consider is whether it is possible to recover f exactly from the data y. And if so, how?

In its abstract form, our problem is of course equivalent to the classical error correcting problem which arises in coding theory as we may think of A as a *linear code*; a linear code is a given collection of codewords which are vectors $a_1, \ldots, a_n \in \mathbf{R}^m$—the columns of the matrix A. Given a vector $f \in \mathbf{R}^n$ (the "plaintext") we can then generate a vector Af in $\mathbf{R}^m$ (the "ciphertext"); if A has full rank, then one can clearly recover the plaintext f from the ciphertext Af. But now we suppose that the ciphertext Af is corrupted by an arbitrary vector $e \in \mathbf{R}^m$ to that the corrupted ciphertext is of the form $Af + e$. The question is then: given the coding matrix A and $Af + e$, can one recover f exactly?

As is well-known, if the fraction of the corrupted entries is too large, then of course we have no hope of reconstructing f from $Af + e$; for instance, assume m is even and consider two distinct plaintexts f, f' and form a vector $g \in \mathbf{R}^m$ by concatenating the first half of Af together with the second half of Af'. Hence, g equals the vector Af with at most half of its entries being corrupted, but g also equals Af' with again at most half of its entries corrupted. This shows that no matter how large m is, accurate decoding is impossible when the size of the support of the error vector is too large. This situation raises an important question: for which fraction ρ of the corrupted entries is accurate decoding possible with practical algorithms? That is, with algorithms whose complexity is at most polynomial in the length m of the codewords?

1.2 Solution via ℓ_1-minimization

To recover f from corrupted data $y = Af + e$, we consider solving the following ℓ_1-minimization problem

$$(P_1) \qquad \min_{g \in \mathbf{R}^n} \|y - Ag\|_{\ell_1}. \tag{1.1}$$

This is a convex program which can be classically reformulated as a linear program. Indeed, (P_1) is equivalent to

$$\min \sum_{i=1}^{m} t_i, \qquad -t \le y - Ag \le t, \tag{1.2}$$

where the optimization variables are $t \in R^m$ and $g \in \mathbf{R}^n$ (as is standard, the generalized vector inequality $x \le y$ means that $x_i \le y_i$ for every coordinate i). Hence, (P_1) is an LP with inequality constraints and can be solved efficiently using standard or even specialized optimization algorithms, see [1].

The main claim of this paper is that for suitable coding matrices, the solution $f^\star$ to our linear program is actually exact; $f^\star = f$!

To develop a quantitative statement, however, we need to introduce the concept of *restricted isometries*. Consider a fixed p by m matrix B and let B_T, $T \subset \{1, \ldots, m\}$, be the $p \times |T|$ submatrix obtained by extracting the columns of B corresponding to the indices in T. Then [4] defines the S-restricted isometry constant δ_S of B which is the smallest quantity such that

$$(1 - \delta_S) \|c\|_{\ell_2}^2 \le \|B_T c\|_{\ell_2}^2 \le (1 + \delta_S) \|c\|_{\ell_2}^2 \tag{1.3}$$

for all subsets T with $|T| \le S$ and coefficient sequences $(c_j)_{j \in T}$. This property essentially requires that every set of columns with cardinality less than S approximately behaves like an orthonormal system.

Let us return to our error correction problem, and consider a matrix B which annihilates the $m \times n$ matrix A on the left, i.e. such that $BA = 0$ (B is any $(m-n) \times n$ matrix whose kernel is the range of A in $\mathbf{R}^m$). Apply B on both sides of the equation $y = Af + e$, and obtain

$$\tilde{y} = B(Af + e) = Be \tag{1.4}$$

since $BA = 0$. Therefore, the decoding problem is reduced to that of recovering the error vector e from the observations Be. Once e is known, Af is known and, therefore, f is also known since A has full rank.

To solve the underdetermined system of linear equations $\tilde{y} = Be$, we search among all vector $d \in \mathbf{R}^m$ obeying $Bd = Be = \tilde{y}$ for that with minimum ℓ_1-norm

$$(P_1') \qquad \min_{d \in \mathbf{R}^m} \|d\|_{\ell_1}, \qquad Bd = Be, \tag{1.5}$$

This convex program—(P_1') may be recast as an LP—also known by the name of *Basis Pursuit* [5] is equivalent to (P_1). The reason is that since on the one hand $y = Af + e$, we may decompose g as $g = f + h$ so that

$$(P_1) \quad \Leftrightarrow \quad \min_{h \in \mathbf{R}^n} \|e - Ah\|_{\ell_1}.$$

But on the other hand, the constraint $Bd = Be$ means that $d = e - Ah$ for some $h \in \mathbf{R}^n$ and, therefore,

$$\begin{aligned} (P_1') \quad &\Leftrightarrow \quad \min_{h \in \mathbf{R}^n} \|d\|_{\ell_1}, \qquad d = e - Ah \\ &\Leftrightarrow \quad \min_{h \in \mathbf{R}^n} \|e - Ah\|_{\ell_1}, \end{aligned}$$

which proves the claim. In conclusion, we can think of our decoding strategy either as the solution of (P_1) or as that of (P_1'). We now state the main results of this paper.

Theorem 1.1 *Suppose that $S \geq 1$ is such that*

$$\delta_{3S} + 3\delta_{4S} < 2, \tag{1.6}$$

and let e be an arbitrary vector supported on a set $T \subset \{1, 2, \ldots, m\}$ obeying $|T| \leq S$. Then the solution to (P_1') is unique and equal to e.

In turn this immediately gives:

Theorem 1.2 *Suppose B is such that $BA = 0$ and let $S \geq 1$ be a number obeying the hypothesis of Theorem 1.1. Then if e obeys the hypothesis of Theorem 1.1, the solution to (P_1) is unique and equal to f.*

This last theorem claims, perhaps, a rather surprising result. In effect, it says that minimizing ℓ_1 recovers *all* input signals $f \in \mathbf{R}^n$ *regardless of the corruption patterns*, provided of course that the support of the error vector is not too large. In particular, one can introduce errors of arbitrary large sizes and still recover the input vector f exactly, by solving a convenient linear program; in other words, as long as the fraction of corrupted entries is not too large, there is nothing a malevolent adversary can do to corrupt Af as to fool the simple decoding strategy (1.1).

The use of the ℓ_1-norm is of course critical. Consider instead a similar program in the ℓ_2-norm, i.e. minimize $\|y - Ag\|_{\ell_2}$. Then $f^\star$ is the least squares solution given by

$$f^\star = (A^T A)^{-1} A^T (Af + e) = f + (A^T A)^{-1} A^T e.$$

In general, the reconstruction error $(A^T A)^{-1} A^T e$ does not vanish and ℓ_2-minimization fails to recover the message f. In fact, as $\|e\|_{\ell_2}$ grows to infinity, the size of the reconstruction error $\|f - f^\star\|_{\ell_2}$ would also, in general, grow to infinity.

1.3 Restricted isometry constant for random matrices

For Theorem 1.2 to be of real interest, one should use matrices B with good restricted isometry constants δ_S; that is, such that the condition of Theorem 1.1 holds with large values of S. How to design such matrices is a delicate question, and we do not know of any matrix which provably obeys (1.6) for interesting values of S. However, if we simply sample a matrix B with i.i.d. entries, it will obey (1.6) for large values of S with overwhelming probability.

Theorem 1.3 *Assume $p \le m$ and let B be a p by m matrix whose entries are i.i.d. Gaussian with mean zero and variance $1/p$. Then the condition of Theorem 1.1 holds with probability at least $1 - O(e^{-\alpha m})$ for some fixed constant $\alpha > 0$, provided that $S \le \rho^* m$. For large values of p and m, one can show that $\rho^* \ge 1/3,000$ for $p = m/2$ $(m = 2n)$, and $\rho^* \ge 1/2,000$ for $p = 3m/4$ $(m = 4n)$.*

Suppose then that A is an m by n Gaussian matrix and set $p = m - n$. Then if the fraction of the corrupted entries does not exceed ρ^*, the solution to (P_1) is unique and equal to f.

Similar statements with different constants hold for other types of ensembles, e.g. for binary matrices with i.i.d. entries taking values $\pm 1/\sqrt{p}$ with probability 1/2. It is interesting that our methods actually give numerical values, instead of the traditional "for some positive constant ρ." However, the numerical bounds we derived in this paper are somewhat overly pessimistic. We are confident that finer arguments and perhaps new ideas will allow to derive versions of Theorem 1.3 with better bounds. Numerical experiments actually suggests that the threshold is indeed much higher, see Section 3.

The proof of Theorem 1.3 is adapted from that of Theorem 1.6 in [4], and we only sketch the main ideas. Take B to be a Gaussian matrix and fix a set of columns T. Then B_T is a fixed p by $|T|$ matrix and we wish to develop bounds on the largest and lowest eigenvalue of $B_T^* B_T$, or equivalently on the largest and lowest singular value of B_T, denoted by $\sigma_{\max}(B_T)$ and $\sigma_{\min}(B_T)$ respectively. It turns out that it is possible to invoke concentration of measure inequalities to obtain the deviation bounds [15]

$$\mathbf{P}\left(\sigma_{\max}(B_T) > 1 + \sqrt{|T|/p} + o(1) + t\right) \le e^{-pt^2/2} \tag{1.7}$$

$$\mathbf{P}\left(\sigma_{\min}(B_T) < 1 - \sqrt{|T|/p} + o(1) - t\right) \le e^{-pt^2/2}; \tag{1.8}$$

here, $o(1)$ is a small term tending to zero as $p \to \infty$ and which can be calculated explicitly, see [9]. Other important references include [14, 18] and the reader will certainly recognize

$1 \pm \sqrt{|T|/p}$ as the asymptotic behavior of the extreme singular values of a Wishart matrix as in the celebrated Marchenko-Pastur law [16]. Applying the union bound for all subsets T obeying $|T| \le S$ then gives explicit control of the singular value spread uniformly over all such sets. A careful study actually yields numerical values.

To see why the decoding is exact, observe that we may think of the annihilator B as a matrix with independent Gaussian entries. Indeed, the range of A is a random space of dimension n embedded in $\mathbf{R}^m$ so that Be is the projection of e on a random space of dimension p. The claim follows from the fact that the range of a p by m matrix with independent Gaussian entries is a random subspace of dimension p.

2 Proof of Theorem 1.1

The proof of the theorem makes use of two geometrical special facts about the solution $d^\star$ to (P_1'). First, $Bd^\star = Be$ which geometrically says that $d^\star$ belongs to a known plane of co-dimension p. Second, because e is feasible, we must have $\|d^\star\|_{\ell_1} \le \|e\|_{\ell_1}$. Decompose $d^\star$ as $d^\star = e + h$. As observed in [7]

$$\|e\|_{\ell_1} - \|h_{T_0}\|_{\ell_1} + \|h_{T_0^c}\|_{\ell_1} \le \|e + h\|_{\ell_1} \le \|e\|_{\ell_1},$$

where T_0 is the support of e, and $h_{T_0}(t) = h(t)$ for $t \in T_0$ and zero elsewhere (similarly for $h_{T_0^c}$). Hence, h obeys the cone constraint

$$\|h_{T_0^c}\|_{\ell_1} \le \|h_{T_0}\|_{\ell_1} \tag{2.1}$$

which expresses the geometric idea that h must lie in the cone of descent of the ℓ_1-norm at e. Exact recovery occurs provided that the null vector is the only point in the intersection between $\{h : Bh = 0\}$ and the set of h obeying (2.1).

We begin by dividing T_0^c into subsets of size M (we will choose M later) and enumerate T_0^c as $n_1, n_2, \ldots, n_{m-|T_0|}$ in decreasing order of magnitude of $h_{T_0^c}$. Set $T_j = \{n_\ell, (j-1)M + 1 \le \ell \le jM\}$. That is, T_1 contains the indices of the M largest coefficients of $h_{T_0^c}$, T_2 contains the indices of the next M largest coefficients, and so on.

With this decomposition, the ℓ_2-norm of h is concentrated on $T_{01} = T_0 \cup T_1$. Indeed, the kth largest value of $h_{T_0^c}$ obeys

$$|h_{T_0^c}|_{(k)} \;\le\; \|h_{T_0^c}\|_{\ell_1}/k$$

and, therefore,

$$\|h_{T_{01}^c}\|_{\ell_2}^2 \;\le\; \|h_{T_0^c}\|_{\ell_1}^2 \sum_{k=M+1}^{N} 1/k^2 \;\le\; \|h_{T_0^c}\|_{\ell_1}^2/M.$$

Further, the ℓ_1-cone constraint gives

$$\|h_{T_{01}^c}\|_{\ell_2}^2 \;\le\; \|h_{T_0}\|_{\ell_1}^2/M \;\le\; \|h_{T_0}\|_{\ell_2}^2 \cdot |T_0|/M$$

and thus

$$\|h\|_{\ell_2}^2 \;=\; \|h_{T_{01}}\|_{\ell_2}^2 + \|h_{T_{01}^c}\|_{\ell_2}^2 \;\le\; (1 + |T_0|/M) \cdot \|h_{T_{01}}\|_{\ell_2}^2. \tag{2.2}$$

Observe now that

$$\begin{aligned} \|Bh\|_{\ell_2} = \|B_{T_{01}} h_{T_{01}} + \sum_{j\geq 2} B_{T_j} h_{T_j}\|_{\ell_2} &\geq \|B_{T_{01}} h_{T_{01}}\|_{\ell_2} - \|\sum_{j\geq 2} B_{T_j} h_{T_j}\|_{\ell_2} \\ &\geq \|B_{T_{01}} h_{T_{01}}\|_{\ell_2} - \sum_{j\geq 2} \|B_{T_j} h_{T_j}\|_{\ell_2} \\ &\geq \sqrt{1-\delta_{M+|T_0|}}\, \|h_{T_{01}}\|_{\ell_2} - \sqrt{1+\delta_M} \sum_{j\geq 2} \|h_{T_j}\|_{\ell_2}. \end{aligned}$$

Set $\rho_M = |T_0|/M$. As we shall see later,

$$\sum_{j\geq 2} \|h_{T_j}\|_{\ell_2} \leq \sqrt{\rho_M} \cdot \|h_{T_0}\|_{\ell_2}, \tag{2.3}$$

and since $Bh = 0$, this gives

$$[\sqrt{1-\delta_{M+|T_0|}} - \sqrt{\rho_M}\,\sqrt{1+\delta_M}] \cdot \|h_{T_{01}}\|_{\ell_2} \leq 0. \tag{2.4}$$

It then follows from (2.2) that $h = 0$ provided that the quantity $\sqrt{1-\delta_{M+|T_0|}} - \sqrt{\rho_M}\,\sqrt{1+\delta_M}$ is positive. Take $M = 3|T_0|$ for example. Then this quantity is positive if $\delta_{3|T_0|} + 3\delta_{4|T_0|} < 2$.

It remains to argue about (2.3). Observe that by construction, the magnitude of each coefficient in T_{j+1} is less than the average of the magnitudes in T_j:

$$|h_{T_{j+1}}(t)| \leq \|h_{T_j}\|_{\ell_1}/M.$$

Then

$$\|h_{T_{j+1}}\|_{\ell_2}^2 \leq \|h_{T_j}\|_{\ell_1}^2/M$$

and (2.3) follows from

$$\sum_{j\geq 2} \|h_{T_j}\|_{\ell_2} \leq \sum_{j\geq 1} \|h_{T_j}\|_{\ell_1}/\sqrt{M} \leq \|h_{T_0}\|_{\ell_1}/\sqrt{M} \leq \sqrt{|T_0|/M} \cdot \|h_{T_0}\|_{\ell_2}.$$

3 Numerical Experiments

In this section, we empirically investigates the performance of our decoding strategy. Of special interest is the location of the breakpoint beyond which ℓ_1 fails to decode accurately. To study this issue, we performed a first series of experiments as follows:

1. select n (the size of the input signal) and m so that with the same notations as before, A is an m by n matrix; sample A with independent Gaussian entries and select the plaintext f at random;

2. select S as a percentage of m;

3. select a support set T of size $|T| = S$ uniformly at random, and sample a vector e on T with independent and identically distributed Gaussian entries, and with standard deviation about that of the coordinates of the output (Af) (the errors are then quite large compared to the "clean" coordinates of Af)[1];

[1]The results presented here do not seem to depend on the actual distribution used to sample the errors.

4. make $\tilde{y} = Af + e$, solve (P_1) and obtain $f^\star$; compare f to $f^\star$;
5. repeat 100 times for each S, and for various sizes of n and m.

The results are presented in Figure 1. In these experiments, we choose $n = 128$, and set $m = 2n$ (Figure 1(a)) or $m = 4n$ (Figure 1(b)). Our experiments show that the linear program recovers the input vector *all the time* as long as the fraction of the corrupted entries is less or equal to 15% in the case where $m = 2n$ and less or equal to 35% in the case where $m = 4n$. We repeated these experiments for different values of n, e.g. $n = 256$ and obtained very similar recovery curves.

It is clear that versions of Theorem 1.3 exist for other type of random matrices, e.g. binary matrices. In the next experiment, we take the plaintext f as a binary sequence of zeros and ones (which is generated at random), and sample A with i.i.d entries taking on values in $\{\pm 1\}$, each with probability 1/2. To recover f, we solve the linear program

$$\min_{g \in \mathbf{R}^n} \|y - Ag\|_{\ell_1} \quad \text{subject to} \quad 0 \le g \le 1, \tag{3.1}$$

and round up the coordinates of the solution to the nearest integer. We follow the same procedure as before except that now, we select S locations of Af at random (the corruption rate is again S/m) and flip the sign of the selected coordinates. We are again interested in the location of the breakpoint.

The results are presented in Figure 2. In these experiments, we choose $n = 128$ as before, and set $m = 2n$ (Figure 2(a)) or $m = 4n$ (Figure 2(b)). Our experiments show that the linear program recovers the input vector *all the time* as long as the fraction of the corrupted entries is less or equal to 22.5% in the case where $m = 2n$ and less than about 35% in the case where $m = 4n$. We repeated these experiments for different values of n, e.g. $n = 256$ and obtained similar recovery curves.

In conclusion, our error correcting strategy seems to enjoy a wide range of effectiveness.

4 Discussion

A first impulse to find the sparsest solution to an underdetermined system of linear equations might be to solve the combinatorial problem

$$(P_0) \qquad \min_{d \in \mathbf{R}^m} \|d\|_{\ell_0} \quad \text{subject to} \quad Bd = Be. \tag{4.1}$$

To the best of our knowledge, solving this problem essentially require exhaustive searches over all subsets of columns of B and is NP-hard [17]. Our results, however, establish a formal equivalence between (P_0) and (P_1') provided that the unknown vector e is sufficiently sparse. In this direction, we would like to mention a series of papers [7, 8, 13, 19] showing the exact equivalence between the two programs (P_0) and (P_1') for special matrices obtained by concatenation of two orthonormal bases. In this literature, equivalence holds if e has fewer than $\rho \cdot \sqrt{m}$ entries; compare with Theorem 1.3 which tolerates a fraction of nonzero entries proportional to m.

For Gaussian random matrices, however, very recent work [6, 20] proved that the equivalence holds when the number of nonzero entries may be as large as $\rho \cdot m$, where $\rho > 0$ is some very small and unspecified positive constant independent of m. This finding is of course similar to ours but the ideas in this paper go much further. First, the paper establishes *deterministic* results showing that exact decoding occurs provided that B obeys the conditions of Theorem 1.1. It is of interest because our own work [2, 3] shows that the condition of Theorem 1.1 holds with large values of S for many other types of matrices, and especially matrices obtained by sampling rows or columns of larger Fourier matrices. These alternatives might be of great practical interest because they would come with fast algorithms for applying A or A^* to an arbitrary vector g and, hence, speed up the computations to find the ℓ_1-minimizer. And second of course, the paper links solutions to sparse underdetermined systems to a linear programming problem for error correction, which we believe is new.

In our linear programming model, the plaintext and ciphertext had real-valued components. Another intensively studied model occurs when the plaintext and ciphertext take values in the finite field $F_2 := \{0, 1\}$. In recent work of Feldman et al. [10–12], linear programming methods (based on relaxing the space of codewords to a convex polytope) were developed to establish a polynomial-time decoder which can correct a constant fraction of errors, and also achieve the information-theoretic capacity of the code. There is thus some intriguing parallels between those works and the results in this paper, however there appears to be no direct overlap as our methods are restricted to real-valued texts, and the work cited above requires texts in F_2. Also, our error analysis is deterministic and is thus guaranteed to correct arbitrary errors provided that they are sufficiently sparse.

The ideas presented in this paper may be adapted to recover input vectors taking values from a finite alphabet. We hope to report on work in progress in a follow-up paper.

In conclusion, we showed that if the coding matrix A is Gaussian for example, one can correct a fraction of completely arbitrary errors by linear programming. We are aware of several refinements allowing to prove sharper versions of Theorems 1.1 and 1.2 (with less stringent conditions) but have ignored such refinements in this paper as to keep the argument as simple as possible. In fact, there is little doubt that other researchers will be able to follow up on our methods and show that ℓ_1-minimization succeeds for corruption rates higher than those we established here.

In this direction, we would like to point out that for Gaussian matrices, say, there is a critical point ρ_c (depending on n and m) such that accurate decoding occurs for all plaintexts and corruption patterns (in the sense of Theorem 1.2) as long as the fraction of corrupted entries does not exceed ρ_c. It would be of theoretical interest to identify this critical threshold, at least in the limit of large m and n, with perhaps n/m converging to a fixed ratio. From a different viewpoint, this is asking about how far the equivalence between a combinatorial and a related convex problem holds. We pose this as an interesting challenge.

References

[1] S. Boyd, and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.

[2] E. J. Candès, J. Romberg, and T. Tao, Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. Submitted to *IEEE Transactions on Information Theory*, June 2004. Available on the ArXiV preprint server: `math.GM/0409186`.

[3] E. J. Candès, and T. Tao, Near optimal signal recovery from random projections: universal encoding strategies? Submitted to *IEEE Transactions on Information Theory*, October 2004. Available on the ArXiV preprint server: `math.CA/0410542`.

[4] E. J. Candès and T. Tao. Decoding by linear programming. Submitted, December 2004. Available on the ArXiV preprint server: `math.MG/0502327`.

[5] S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM J. Scientific Computing* **20** (1999), 33–61.

[6] D. L. Donoho. For Most Large Underdetermined Systems of Linear Equations the Minimal ℓ_1-norm Solution is also the Sparsest Solution. Manuscript, September 2004.

[7] D. L. Donoho and X. Huo. Uncertainty principles and ideal atomic decomposition. *IEEE Transactions on Information Theory*, **47** (2001), 2845–2862.

[8] D. L. Donoho and M. Elad. Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ_1 minimization, *Proc. Natl. Acad. Sci. USA* **100** (2003), 2197–2202.

[9] N. El Karoui. New Results about Random Covariance Matrices and Statistical Applications. Stanford Ph. .D. Thesis, August 2004.

[10] J. Feldman. *Decoding Error-Correcting Codes via Linear Programming*. Ph.D. Thesis 2003, Massachussets Institute of Technology.

[11] J. Feldman. LP decoding achieves capacity, 2005 ACM-SIAM Symposium on Discrete Algorithms (SODA), preprint (2005).

[12] J. Feldman, T. Malkin, C. Stein, R. A. Servedio, M. J. Wainwright. LP Decoding Corrects a Constant Fraction of Errors. *Proc. IEEE International Symposium on Information Theory (ISIT)*, June 2004.

[13] R. Gribonval and M. Nielsen. Sparse representations in unions of bases. *IEEE Trans. Inform. Theory* **49** (2003), 3320–3325.

[14] I. M. Johnstone. On the distribution of the largest eigenvalue in principal components analysis. *Ann. Statist.* **29** (2001), 295–327.

[15] M. Ledoux. *The concentration of measure phenomenon*. Mathematical Surveys and Monographs **89**, American Mathematical Society, Providence, RI, 2001.

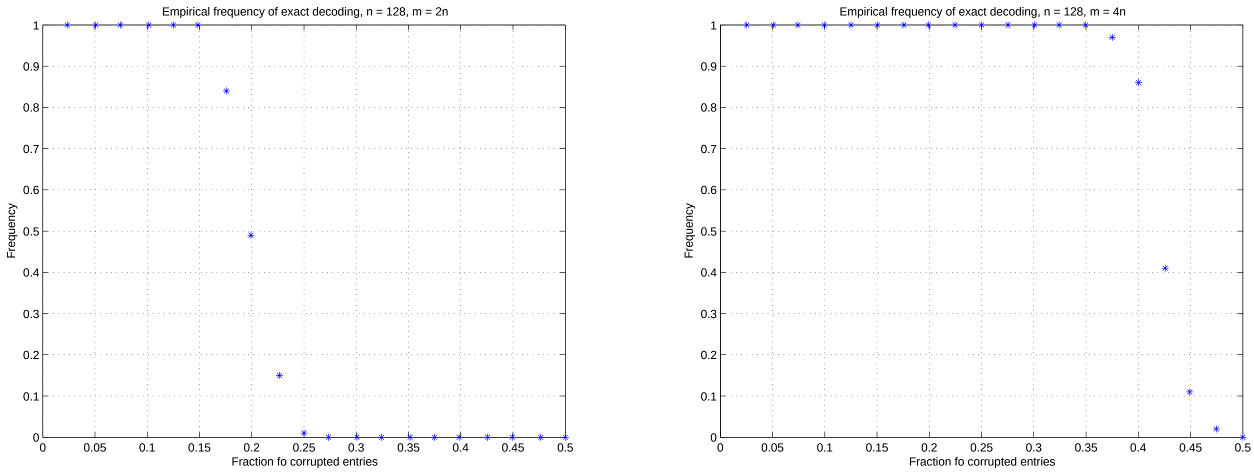
[16] V. A. Marchenko and L. A. Pastur. Distribution of eigenvalues in certain sets of random matrices. *Mat. Sb. (N.S.)* **72** (1967), 407–535 (in Russian).

[17] B. K. Natarajan. Sparse approximate solutions to linear systems. *SIAM J. Comput.* **24** (1995), 227-234.

[18] S. J. Szarek. Condition numbers of random matrices. *J. Complexity* **7** (1991), 131–149.

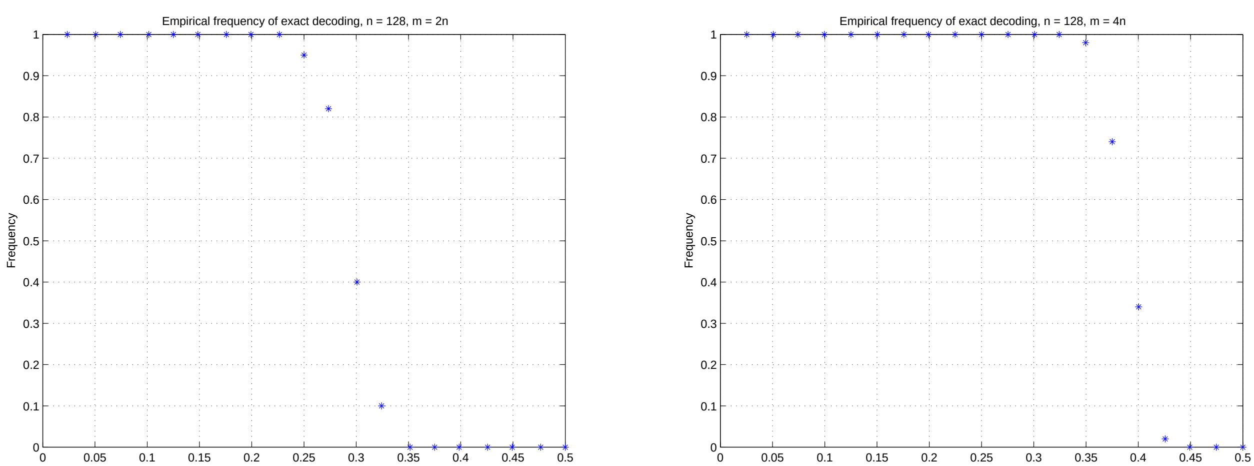
[19] J. A. Tropp. Greed is good: Algorithmic results for sparse approximation. Technical Report, The University of Texas at Austin, 2003.

[20] M. Rudelson and R. Vershynin. Geometric approach to error correcting codes and reconstruction of signals. Technical Report, UC Davis, February 2005.



(a) (b)

Figure 1: ℓ_1-recovery of an input signal from $y = Af + e$ with A an m by n matrix with independent Gaussian entries. In these experiments, we set $n = 128$. (a) Success rate of (P_1') for $m = 2n$. (b) Success rate of (P_1') for $m = 4n$. On the left, exact recovery occurs as long as the corruption rate does not exceed 15%. On the right, the breakdown is near 35%.



(a) (b)

Figure 2: ℓ_1-recovery of a binary sequence from corrupted data y; A is an m by n matrix with independent binary entries and the vector of errors is obtained by randomly selecting coordinates of Af and flipping their sign. In these experiments, we set $n = 128$ (a) Success rate of for $m = 2n$. (b) Success rate for $m = 4n$. On the left, exact recovery occurs as long as the corruption rate does not exceed 22.5%. On the right, the breakdown is near 35%.