13 Lecture 13, November 12, 2014

13.1 Limited linear independence, limited statistical independence, and error correcting codes.

We were introduced in a previous lecture to the notion of linear error-correcting codes. We worked over the base field $GF(2)$, also known as $\mathbb{Z}/2$. (Which is to say, we added bit-vectors using XOR.) Encoding of messages in such a code is simply multiplication of the message, as a vector $v \in F^m$, by the *generator matrix* C of the code; the result, if C is $m \times n$, is an n-bit codeword.

$$(\text{ message } v) \begin{pmatrix} \text{gen. matrix} \\ \\ C \end{pmatrix} = (\ldots \text{ codeword } vC \ldots)$$

The set of codewords is exactly Rowspace(C).

A strong property for such a code to have is that, for some large value of k, every codeword has weight (number of 1's) at least $k+1$. We call such a code *k-error-detecting* because the property ensures

1. Error *detection* up to k errors
2. Error *correction* up to $\lfloor k/2 \rfloor$ errors.

This property is *not* possessed by codes achieving near-optimal rate in Shannon's coding theorem. It is too strong a property for that purpose. This property protects against adversarial, not just random, noise.

Error detection can be performed with the aid of the *parity check matrix* M:

$$\text{Nullspace}(M) = \text{Rowspace}(C)$$

$$\begin{pmatrix} \text{generator matrix} \\ \\ C \end{pmatrix} \begin{pmatrix} \text{parity} \\ \text{check} \\ \text{matrix} \\ M \end{pmatrix} = \begin{pmatrix} 0 \end{pmatrix}$$

$$wM = 0 \iff w \in \text{Rowspace}(C) \iff w \text{ is a codeword}$$

$$\left.\begin{array}{l} \text{Every vector in} \\ \text{Rowspace}(C) \text{ has} \\ \text{weight } \geq k+1 \end{array}\right\} \iff \left\{\begin{array}{l} \text{Every } k \text{ rows} \\ \text{of } M \text{ are linearly} \\ \text{independent} \end{array}\right.$$

For any fixed values of n and k, the code is most efficient when the message length m, which is the number of rows of C, is as large as possible; equivalently, the number of columns of M, $\ell = n - m$, is as small as possible. So we'll want to design a matrix M with few columns in which every k rows are linearly independent.

But first, let's see a connection between linear and statistical independence.

Let B be a $k \times \ell$ matrix over $GF(2)$, with full row rank.

If $x \in_U (GF(2))^\ell$ then $y = Bx \in_U (GF(2))^k$

$$\begin{pmatrix} y \end{pmatrix} = \begin{pmatrix} \quad B \quad \end{pmatrix} \begin{pmatrix} x \end{pmatrix}$$

because the pre-image of any particular y is an affine subspace (a translate of the right nullspace of B).

Now, if we have a matrix M with n rows, of which every k are linearly independent, then every k bits of $z = Mx$ are uniformly distributed in $(GF(2))^k$.

$$\begin{pmatrix} \\ z \\ \\ \end{pmatrix} = \begin{pmatrix} \quad \\ M \\ \quad \end{pmatrix} \begin{pmatrix} x \end{pmatrix}$$

We've exhibited *dual applications of the parity check matrix:*

- Action on row vectors: checking validity of a received word y as a codeword. (And computing the "syndrome" of y, for those who have had a course in coding theory.)
- Action on column vectors: converting few uniform iid bits into many k-wise independent uniform bits.

Now we can see an entire sample space on n bits that are uniform and k-wise-independent. At the right end we place the uniform distribution on all 2^ℓ vectors of the vector space $GF(2)^\ell$.

$$\begin{pmatrix} \\ \Omega \\ \\ \end{pmatrix} = \begin{pmatrix} \quad \\ M \\ \quad \end{pmatrix} \begin{pmatrix} 0 & 0 & \dots & 1 & 1 \\ \dots & \text{unif.} & \text{dist.} & \text{on} & \text{cols} \\ 0 & 1 & \dots & 0 & 1 \end{pmatrix}$$

Ω is the uniform distribution on the columns on the LHS.

Maximizing the transmission rate $\frac{m}{n} = \frac{n-\ell}{n}$ of a binary, k-error-detecting code, is equivalent to *minimizing* the size $|\Omega| = 2^\ell$ of a linear k-wise independent binary sample space.

So how big does $|\Omega|$ have to be?

Theorem 59 *1. For all n there is a sample space on n uniform k-wise independent bits of size $O(n^{\lfloor k/2 \rfloor})$.*

2. For all n, any sample space on n k-wise independent bits, none of which is a.s. constant, has size $\Omega(n^{\lfloor k/2 \rfloor})$.

Existing example

Before proving the upper bound for general k, let's revisit a construction we've seen. It uses $GF(p)$ for a large prime p rather than (as we will use here) an extension field of $GF(2)$, but other than that the idea is the same. We set $H = \{h_{a,b}\}_{a,b \in GF(p)} : GF(p) \to GF(p)$, $h_{a,b}(x) = a \cdot x + b$. For any x, $h_{a,b}(x)$ is uniformly distributed (due to uniformity of b), and for any $x \neq x'$, $h_{a,b}(x) - h_{a,b}(x') = a \cdot (x - x')$ is uniformly distributed (due to uniformity of a).

In terms of our general approach this construction can be viewed as follows. The $p \times 2$ parity check matrix M (over $GF(2)$) has p rows and two columns; the first column is always a 1. It maps the uniform distribution on all p^2 column vectors (each of which is a pair $(b, a)^\dagger$ specifying a hash function $h_{a,b}$), to the sample space Ω we constructed in a previous lecture.

$$\Omega \quad = \quad \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \\ 1 & \dots \\ 1 & p-1 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & \dots & 1 & 1 & \dots & b & \dots & p-1 \\ 0 & 1 & \text{unif.} & \text{dist.} & \text{on} & \text{cols} & \dots & a & \dots & p-1 \end{pmatrix}$$

(Although $k = 2$, this sample space is of quadratic, not linear size, basically because we're producing random variables in a large set, $GF(p)$, where $p = n$ the number of rvs. In the next section we'll go back to looking at binary-valued rvs.)

13.2 Proof of Thm (59) Part (1): Upper bound on the size of k-wise independent sample spaces

Instead of using the fields modulo a prime, this construction uses the finite fields whose cardinalities are powers of 2. These are called extension fields of $GF(2)$. If you are not familiar with this, just keep in mind that for each integer $r \geq 1$ there is a (unique) field with 2^r elements. We can add, subtract, multiply and divide these without leaving the set; in particular, in the usual way of representing the elements of the field as bit strings of length r, addition is simply XOR addition.[11] Specifically, we can think of the elements of $GF(2^r)$ as the polynomials of degree $\leq r-1$ over $GF(2)$. Multiplication is modulo some fixed irreducible polynomial of degree r. The usual respresentation of an element of the field is the list of coefficients of this polynomial.

But all we really need today are three things: (a) Like $GF(2)$, $GF(2^r)$ is a field of characteristic 2, i.e., $2x = 0$. (b) For matrices over $GF(2^r)$ the usual concepts of linear independence and rank apply. (c) There is a bijective mapping $b : GF(2^r) \to (GF(2))^r$ such that $b(x) + b(y) = b(x+y)$.

Now, round n up to the nearest $n = 2^r - 1$. and let $a_1, \dots, a_n$ denote the nonzero elements of the field. Let M_1 be the following Vandermonde matrix over the field $GF(2^r)$:

$$M_1 = \begin{pmatrix} 1 & a_1 & a_1^2 & \dots & a_1^{k-1} \\ 1 & a_2 & a_2^2 & \dots & a_2^{k-1} \\ \dots & \dots & \dots & \dots & \dots \\ 1 & a_n & a_n^2 & \dots & a_n^{k-1} \end{pmatrix}$$

Exercise: Every k rows of M_1 are linearly independent over $GF(2^r)$.

M_1 is an $n \times k$ matrix over $GF(2^r)$. Next, expand each of its entries as a row vector of bits, thus forming an $n \times kr$ matrix M_2 over $GF(2)$:

$$M_2 = \begin{pmatrix} b(1) = 001 & b(a_1) = 001 & \dots & b(a_1^{k-1}) = 001 \\ b(1) = 001 & b(a_2) = 010 & \dots & b(a_2^{k-1}) = \dots \\ \dots & \dots & \dots & \dots \\ b(1) = 001 & b(a_n) = 111 & \dots & b(a_n^{k-1}) = \dots \end{pmatrix}$$

Corollary: Every k rows of M_2 are linearly independent over $GF(2)$.

Actually it is possible to even further reduce the number of columns while retaining the corollary.

First, we can drop the leading 0's in the first entry.

Second, we can strike out all batches of columns generated by positive even powers.

[11] See any introduction to Algebra, for instance Artin [5].

$$M_3 = \begin{pmatrix} 1 & b(a_1) = 001 & b(a_1^3) = 001 & \dots & \dots \\ 1 & b(a_2) = 010 & b(a_2^3) = \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ 1 & b(a_n) = 111 & b(a_n^3) = \dots & \dots & \dots \end{pmatrix}$$

Lemma 60 *Every set of rows that is linearly independent (over $GF(2^r)$) in M_1 is also linearly independent (over $GF(2)$) in M_3. Hence every k rows of M_3 are linearly independent.*

Proof: In a $GF(2)$ matrix, stating that a set of rows is lin. indep. is equivalent to stating that any subset of those rows has nonzero sum.

Let a set of rows R be independent in M_1; we show the same is true in M_3. Specifically, we show that for any subset $\emptyset \subset S \subseteq R$, the sum of the rows S in M_3 is nonzero.

Since these rows are independent in M_1, their sum is nonzero. If the first entry of the M_1-sum-vector is nonzero—i.e., if $|S|$ is odd—then the same is true in M_1. Otherwise, let $t > 0$ be the smallest value such that $\sum_{i \in S} a_i^t \neq 0$; it is enough to show that t is odd. Suppose not, so $t = 2t'$. Then, since Characteristic$(GF(2^r)) = 2$,

$$\sum_{i \in S} a_i^{2t'} = \left(\sum_{i \in S} a_i^{t'} \right)^2$$

so $\sum_{i \in S} a_i^{t'} \neq 0$, contradicting minimality of t. □

Finally, recalling that $n = 2^r - 1$, we have $|\Omega| = 2^{1+r\lfloor k/2 \rfloor} \in O(n^{\lfloor k/2 \rfloor})$.

Comment:
If you want n k-wise independent bits but the marginals are not uniform, then this construction doesn't work. The best known construction in general, due to Koller and Megiddo [44], is of size $O(n^k)$.

13.3 Back to Gale-Berlekamp

We now see a deterministic polynomial-time algorithm for playing the G-B game. As we demonstrated last time, it is enough to use a 4-wise independent sample space in order to achieve $\Omega(n^{3/2})$ expected performance. The above construction gives us a 4-wise independent sample space of size $O(n^2)$. All we have to do is exhaustively list the points of the sample space until we find one with performance $\Omega(n^{3/2})$.