CSC310 – Information Theory Sam Roweis

Lecture 17:

Hamming Codes

November 14, 2005

Reminder: Linear Codes

- Recall that Shannon's second theorem tells us that for any noisy channel, there is some code which allows us to achieve error free transmission at a rate up to the capacity.
- However, this might require us to encode our message in very long blocks, which if we implemented codes naively would require memory and time exponential in the blocklength.
- So we need a way to define a code and to encode/decode that requires memory/time only polynomial in the block size.
- Linear codes provide this by defining a code using a set of basis codewords (rows of the generator matrix) or equivalently using a set of constraint equations (rows of the parity check matrix).

Reminder: Distance of a Code

- A code's *minimum distance* is the minimum of $d(\mathbf{u}, \mathbf{v})$ over all distinct codewords $\mathbf{u}$ and $\mathbf{v}$.
- If the minimum distance is $\geq 2t+1$, nearest neighbor decoding will always decode correctly when there are $\leq t$ errors.
- To find the minimum distance for a code with 2^K codewords, we will in general have to look at all $2^K(2^K-1)/2$ pairs of codewords.
- For linear codes, the minimum distance is the minimum weight of the 2^K-1 non-zero codewords, which is equal to the rank of the parity-check matrix H plus one.
- Special cases:
  - If H has a column of all zeros, then $d = 1$.
  - If H has two identical columns, then $d \leq 2$.
  - For binary codes, if all columns are distinct and non-zero, $d \geq 3$.

Hamming Codes

- We have seen that a binary $[N, K]$ code will correct any single error if all the columns in its parity-check matrix are non-zero & distinct.
- One way to achieve this: Make the $N-K$ bits in successive columns be the binary representations of the integers 1, 2, 3, etc.
- E.g. to get a parity-check matrix for a $[7, 4]$ code capable of correcting any single error (this was the assignment question):

$$\begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

- When N is a power of two minus one, the columns of H contain binary representations of all non-zero integers up to $2^{N-K}-1$.
- These are the called the *Hamming codes* after their inventor, Dick Hamming, a Los-Alamos scientist, Bell Labs engineer, collaborator of Shannon and later founder of the ACM.

The $[7, 4]$ Binary Hamming code

- The $[7, 4]$ Hamming code is defined over Z_2 by the following four basis vectors:

  $$1000101, \quad 0100110, \quad 0010111, \quad 0001011$$

  Since these basis vectors are independent, there are 16 codewords.
- We could also define the code by the following equations that are satisfied by any codeword $\mathbf{u}$:

  $$\begin{aligned} u_1 + u_2 + u_3 + u_5 &= 0 \\ u_2 + u_3 + u_4 + u_6 &= 0 \\ u_1 + u_3 + u_4 + u_7 &= 0 \end{aligned}$$

- This code is capable of correcting any single bit transmission error.
- There are other sets equations and other sets of basis vectors that define an equivalent code, just with the check bits permuted.

Distance of the $[7, 4]$ Hamming Code

- The $[7, 4]$ Hamming code is defined by the parity-check matrix:

  $$\begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

- Clearly, all the columns of H are non-zero, and they are all distinct. So $d \geq 3$. We can see that $d = 3$ by noting that the first three columns are linearly dependent, since

  $$\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

- Or we can observe that some codewords (e.g. 1110000) have weight of only 3.
- Since it has minimum distance 3, this code can correct any single bit transmission error.

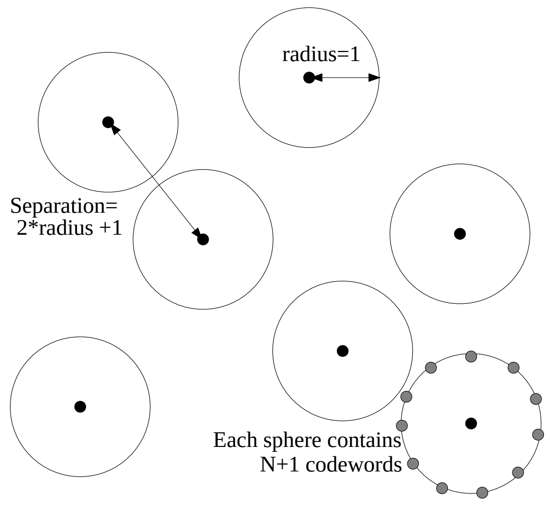
Hamming's Sphere-Packing Bound

- We'd like to make the minimum distance as large as possible, or alternatively, have as many codewords as possible for a given distance. There's a limit, however.
- Consider a binary code with $d = 3$, which can correct any single error. The "spheres" of radius one around each codeword must be disjoint — so that any single error leaves us closest to the correct decoding.
- For codewords of length N, each such sphere contains $1+N$ points. If we have m codewords the total number of points in all spheres will be $m\,(1+N)$, which can't be greater than the total number of points, 2^N.
- So for binary codes that can correct any single error, the number of codewords is limited by

  $$m \;\leq\; 2^N/(1+N)$$

Distances and Packing

- Here's a picture of codewords (black dots) for a code with minimum distance 3, showing the limits we just discussed:



A More General Version of the Bound

- A binary code of length N that is guaranteed to correct any pattern of up to t errors can't have more than this number of codewords:

$$2^N \left(1 + \binom{N}{1} + \binom{N}{2} + \cdots + \binom{N}{t}\right)^{-1}$$

- The kth term in the brackets is the number of possible patterns of k errors in N bits:

$$\binom{N}{k} \;=\; \frac{N!}{k!\,(N-k)!}$$

- If the above bound is actually reached, the code is said to be *perfect*. For a perfect code, the disjoint spheres of radius t around codewords cover all points.
- Very few perfect codes are known. Usually, we can't find a code with as many codewords as would be allowed by this bound.

Hamming Codes are Perfect

- For each positive integer c, there is a binary Hamming code of length $N = 2^c - 1$ and dimension $K = N - c$. These codes all have minimum distance 3, and hence can correct any single error.
- They are also perfect, since

$$2^N/(1+N) \;=\; 2^{2^c-1}/(1+2^c-1) \;=\; 2^{2^c-1-c} \;=\; 2^K$$

which is the number of codewords.

- One consequence: A Hamming code can correct any single error, but if there is more than one error, it will not be able to give any indication of a problem — instead, it will "correct" the wrong bit, making things worse.
- The *extended Hamming codes* add one more check bit (ie, they have one more row of all 1s to the parity-check matrix).
This allows them to detect when two errors have occurred.

The Gilbert-Varshamov Bound

- The sphere-packing bound is an *upper* limit on how many codewords we can have. There's also a *lower* limit, showing there **is** a code with at least a certain number of codewords.
- There is a binary code of length N with minimum distance d that has at least the following number of codewords:

$$2^N \left(1 + \binom{N}{1} + \binom{N}{2} + \cdots + \binom{N}{d-1}\right)^{-1}$$

- Why? Imagine spheres of radius $d-1$ around codewords in a code with fewer codewords than this. The number of points in each sphere is the sum above in brackets, so the total number of points in these spheres is less than 2^N. So there's a point outside these spheres where we could add a codeword that is at least d away from any other codeword.

Encoding Hamming Codes

- By rearranging columns, we can put the parity-check matrix for a Hamming code in systematic form. For the $[7,4]$ code, we get

$$H \;=\; \begin{pmatrix} 0&1&1&1&1&0&0 \\ 1&0&1&1&0&1&0 \\ 1&1&0&1&0&0&1 \end{pmatrix}$$

- Recall that a systematic parity check matrix $[P^T \,|\, I_{N-K}]$ goes with a systematic generator matrix $[I_K \,|\, P]$. In this case, we have

$$G \;=\; \begin{pmatrix} 1&0&0&0&0&1&1 \\ 0&1&0&0&1&0&1 \\ 0&0&1&0&1&1&0 \\ 0&0&0&1&1&1&1 \end{pmatrix}$$

- We encode a message block, $\mathbf{s}$, of four bits, by computing $\mathbf{t} = \mathbf{s}G$. The first four bits of $\mathbf{t}$ are the same as $\mathbf{s}$; the remaining three bits are "check bits". Note: The order of check bits may vary depending on how the code is constructed.

Decoding Hamming Codes

- Consider the original (non-systematic) parity-check matrix:

$$H = \begin{pmatrix} 0&0&0&1&1&1&1 \\ 0&1&1&0&0&1&1 \\ 1&0&1&0&1&0&1 \end{pmatrix}$$

- Suppose $\mathbf{t}$ is sent, but $\mathbf{r} = \mathbf{t} + \mathbf{n}$ is received ($\mathbf{n}$ is channel noise).
- The receiver can compute the *syndrome* for $\mathbf{r}$:

$$\mathbf{z} = \mathbf{r}H^T = (\mathbf{t}+\mathbf{n})H^T = \mathbf{t}H^T + \mathbf{n}H^T = \mathbf{n}H^T$$

  Note that $\mathbf{t}H^T = \vec{0}$ since $\mathbf{t}$ is a codeword.
- If there were no errors, $\mathbf{n} = \vec{0}$, so $\mathbf{z} = \vec{0}$.
- If there is one error, in position i, then $\mathbf{n}H^T$ will be the ith column of H — which contains the binary representation of the number i!
- So to decode, we compute the syndrome, and if it is non-zero, we flip the bit it identifies. Easy! (If we rearranged H to systematic form, we modify this procedure in corresponding fashion.)

Syndrome Decoding in General

- For any linear code with parity-check matrix H, we can find the nearest-neighbor decoding of a received block, $\mathbf{r}$, using the syndrome, $\mathbf{z} = \mathbf{r}H^T$.
- If the received data is $\mathbf{r} = \mathbf{t} + \mathbf{n}$, where $\mathbf{t}$ is the transmitted codeword, and $\mathbf{n}$ is the *noise pattern*, then $\mathbf{z} = \mathbf{n}H^T$ (since $\mathbf{t}H^T = \vec{0}$).
- A nearest-neighbor decoding can be found by finding an noise pattern, $\mathbf{n}$, that produces the observed syndrome $\mathbf{z}$, and which has the smallest possible weight. Then we decode $\mathbf{r}$ as $\mathbf{r} - \mathbf{n}$.
- So encoding involves a matrix multiplication and so does decoding! However, decoding also involves a table lookup...

Building a Syndrome Decoding Table

- We can build a table indexed by the syndrome $\mathbf{z}$ that gives the noise pattern $\mathbf{n}$ of minimum weight for each syndrome.
- We initialize all entries in the table to be empty.
- We then consider the non-zero noise patterns, $\mathbf{n}$, in some order of non-decreasing weight. For each $\mathbf{n}$, we compute the syndrome, $\mathbf{z} = \mathbf{n}H^T$, and store $\mathbf{n}$ in the entry indexed by $\mathbf{z}$, *provided* this entry is currently empty.
- We stop when the table has no empty entries left to fill.
- Problem: The size of the table is exponential in the number of check bits — it has $2^{N-K} - 1$ entries for an $[N, K]$ code.

Example: The $[5, 2]$ Code

- Recall the $[5, 2]$ code with this parity-check matrix:

$$\begin{pmatrix} 1&1&0&0&0 \\ 0&0&1&1&0 \\ 1&0&1&0&1 \end{pmatrix}$$

- Here is a syndrome decoding table for this code:

$\mathbf{z}$	$\mathbf{n}$
001	00001
010	00010
011	00100
100	01000
101	10000
110	10100
111	01100

  The last two entries are not unique.