CS3236: Tutorial 6
(Practical Channel Codes)

1. **[Simple Linear Code]**

   An error-correcting code for the binary symmetric channel with noise $\delta = 0.18$ is given below as a generator matrix $\mathbf{G}$:

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \end{bmatrix}$$

   (a) How many source bits and parity-check bits are there? Express the parity-check bits as a function of the message bits.

   (b) How many possible valid codewords are there?

   (c) Give the parity-check matrix $\mathbf{H}$.

   (d) Suppose that some bits of $\mathbf{x}$ are flipped to produce a noisy version $\mathbf{y}$, i.e., $\mathbf{y} = \mathbf{x} \oplus \mathbf{z}$. Give the decoding algorithm as pairs of (syndrome – procedure) that corrects up to one error.

   (e) What is the rate R of this code ?

   (f) Write down an expression for the probability that at most 1 bit flip occurs in the transmitted codeword, and compute its value when $\delta = 0.18$. How does this relate to the probability of decoding error?

2. **[Two Errors?]**

   A Hamming code with $k = 4$ and $n = 7$ can correct up to one bit flip. Explain why a code with $k = 8$ and $n = 14$ could not possibly be guaranteed to correct up to 2 bit flips.

   (Hint: Every error sequence $\mathbf{z}$ *of weight at most* 2 *would need to give a unique syndrome.)*

3. **[Hamming Code of Order r]**

   The Hamming code of order r is a linear code with $n = 2^r - 1$ code bits and $k = 2^r - r - 1$ message bits, given by the generator matrix

$$\mathbf{G} = [\mathbf{I}_k \;\; \mathbf{B}],$$

   where $\mathbf{B}$ is a $k \times r$ matrix whose rows are all the possible row vectors of length r with entries 0 and 1, and at least two 1's (the number of such vectors is equal to $2^r - r - 1$). The order in which these row vectors appear in $\mathbf{B}$ does not matter.

   (a) Write the code matrix for a Hamming code of order 3.

(b) Show that the Hamming code of order k has minimum weight 3, and can therefore correct one error.

(Hint: Recall that the codewords are linear combinations of the rows of $\mathbf{G}$*. Try analyzing three cases separately: (i) single rows, (ii) sums of* 3 *or more rows, and (iii) sums of exactly* 2 *rows.)*

4. **[Extended Hamming Code]**

Take the Hamming code $\mathcal{C}$ from the lecture, with generator matrix

$$\mathbf{G}_{\text{Hamming}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

and append an overall parity check to the codewords of that code. That is, each codeword of a Hamming code is extended by 1 bit which is 0 if the codeword contains an even number of 1's and 1 if the codeword contains an odd number of 1's.

For example, the codeword 0000000 becomes 00000000, and the codeword 1110000 becomes 11100001.

(a) Write down the generator matrix of the new code

(b) Show that the new code has minimum distance 4. *(Hint: You may use the fact that the Hamming code has minimum distance* 3*. Let* $\mathbf{x}$*,* $\mathbf{x}'$ *be two different codewords, let* $\mathbf{y}$*,* $\mathbf{y}'$ *be the vectors with the last bit removed, and try looking at different cases of* $d_{\text{H}}(\mathbf{y}, \mathbf{y}')$ *and whether or not* $p = p'$*.)*

5. **[From Non-Systematic to Systematic]**

Recall that a linear code is said to be *systematic* if its generator matrix takes the form

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & \dots & 0 & g_{1,k+1} & \dots & g_{1,n} \\ 0 & 1 & \dots & 0 & g_{2,k+1} & \dots & g_{2,n} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 & g_{k,k+1} & \dots & g_{k,n} \end{bmatrix}$$

with an identity matrix on the left. In other words, the first k codeword bits are exactly the message bits.

In this question, we will explore how to convert a non-systematic code into an equivalent systematic one (i.e., one with the same collection of codewords, but possibly assigned to different messages). The idea is to apply the following two operations that do not change distance properties:

- Swapping two rows of $\mathbf{G}$ just amounts to re-labeling which message bit is which.
- Letting $\mathbf{g}_i$ be the i-th row of $\mathbf{G}$, replacing a given row $\mathbf{g}_i$ by $\widetilde{\mathbf{g}}_i = \mathbf{g}_i \oplus \mathbf{g}_j$ for $j \neq i$ does not change the overall set of valid codewords. To see this, consider the linear combinations of the two rows (indexed by i and j) that can be obtained for the two codes:
  - If $u_i = 0$ then we have $(u_i\widetilde{\mathbf{g}}_i) \oplus (u_j\mathbf{g}_j) = u_j\mathbf{g}_j$ and $(u_i\mathbf{g}_i) \oplus (u_j\mathbf{g}_j) = u_j\mathbf{g}_j$, so the two codes produce the same codeword.
  - If $u_i = 1$, we have

$$\begin{aligned} (u_i\widetilde{\mathbf{g}}_i) \oplus (u_j\mathbf{g}_j) &= \widetilde{\mathbf{g}}_i \oplus (u_j\mathbf{g}_j) && \text{(since } u_i = 1\text{)} \\ &= \mathbf{g}_i \oplus \mathbf{g}_j \oplus (u_j\mathbf{g}_j) && \text{(definition of } \tilde{g}_i\text{)} \\ &= \mathbf{g}_i \oplus ((u_j \oplus 1)\mathbf{g}_j) && \text{(by linearity)} \end{aligned}$$

  so whatever was produced by $u_j = 1$ in the old code is produced by $u_j = 0$ in the new code, and vice versa. Therefore, both codes produce the same set of codewords.

(a) Use the above operations to convert the non-systematic code

$$\mathbf{G} = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}$$

into an equivalent systematic one.

(Hint: Try to get the first row starting with 1*, then the second row starting with* 01*, and the third with* 001*. Then work backwards and try to get the second row starting with* 010*, and the first with* 100*. This is is a form of Gaussian elimination (row reduction), but done with modulo-*2 *arithmetic.)*

(b) List the codewords for the original non-systematic code and the systematic code you found in part (a), and verify that the two codes have just as many codewords of each weight.

6. **(Optional) [Distance Bounds]**

Here we look at the existence (or non-existence) of binary codes with a given *minimum distance* (i.e., smallest Hamming distance between any two (different) valid codewords).

(a) Show that any binary block code of block length n and minimum distance d has *at most*

$$\frac{2^n}{\sum_{i=0}^{\lfloor (d-1)/2 \rfloor} \binom{n}{i}}$$

codewords. This is called the *sphere packing* bound.

(Hint: The clue is in the name! Note that with minimum distance d*, the "spheres of radius* $\lfloor (d-1)/2 \rfloor$*" (i.e., the set of all binary codewords within that distance) centered at each codeword must be non-overlapping. How many sequences lie in each sphere?)*

(b) Show that given integers n and d, there exists a code with block length n, minimum distance $d_{\min} \geq d$, and *at least*

$$\frac{2^n}{\sum_{i=0}^{d-1} \binom{n}{i}}$$

codewords. This is known as the *Gilbert-Varshamov Bound*. Notice that the summation goes to $d-1$, rather than $\lfloor (d-1)/2 \rfloor$ in part (a).

(Hint: Ok, no clue in this name. Try to design an iterative procedure that keeps a list of "feasible codewords". Initially everything is feasible, but after a codeword is selected, everything within distance $d-1$ *is marked as infeasible. How many iterations can we do before having no more feasible codewords?)*

7. **(Optional) [Singleton Bound / Maximum Distance Separable Codes]**

While we have focused on binary codes with codewords $\mathbf{x} = (x_1, \ldots, x_n)$ such that $x_i \in \mathcal{X} = \{0, 1\}$, there are also very powerful codes defined on larger alphabets. Here we consider q-ary codes, with $\mathcal{X} = \{0, 1, \ldots, q-1\}$. The Hamming distance is still defined as the number of differing symbols: $d_{\mathrm{H}}(\mathbf{x}, \mathbf{x}') = \sum_{i=1}^{n} \mathbf{1}\{x_i \neq x_i'\}$.

We consider (possibly non-linear) codes whose number of codewords is $M = q^k$ for some integer $k < n$. We write $n = k + t$, so that t represents how much higher n is than k. (Note that since $M = q^k$, we can view the code as mapping length-k q-ary sequences to length-k q-ary codewords.)

Show that in this setup, any code must have minimum distance at most $d_{\min} \leq t + 1$.

(Hint: The argument is actually very simple. Take the first k *symbols of each codeword, and consider two possible cases: (i) There exist two codewords giving the same first* k *symbols; (ii) All of the* $M = q^k$ *codewords have a distinct sequence of first* k *symbols.)*

This result is known as the *singleton bound*, and codes achieving $d_{\min} = t + 1$ are said to be *maximum distance separable* (MDS). This is not such a useful concept for binary codes ($q = 2$), because the only MDS codes in that case are trivial. But for higher q, things get more interesting. A famous class of (non-binary linear) codes called *Reed-Solomon codes* are MDS whenever q is a prime power with $q \geq n$.

Hints

1. Mostly follows the lecture.
2. Compare the number of error patterns to the number of syndromes.
3. Hint given in the question.
4. For (a) just take $\mathbf{G}$ and add another suitably-chosen column. For (b) a hint was already given – to elaborate, first argue that $d_{\mathrm{H}}(\mathbf{y}, \mathbf{y}') \geq 3$ and then handle the cases $= 3$ and ≥ 4 separately.
5. Hint given in the question.
6. Hints given in the question.
7. Hint given in the question.