Lecture 18: Hamming codes; midterm review

DANIEL WELLER

TUESDAY, MARCH 26, 2019
This Venn diagram portrays the possible error states of four bits $d_1$, $d_2$, $d_3$, and $d_4$, using three “parity” checks with check digits $p_1$, $p_2$, and $p_3$. 

Image credit: Colin M. L. Burnett/Wikipedia
Recall: Matrix algebra

Given a binary matrix $A$:

\[
\begin{pmatrix}
1 & 0 & 0 & 1 \\
0 & 0 & 1 & 1 \\
1 & 1 & 0 & 1 \\
\end{pmatrix}
\]

We can multiply this matrix by a vector $x$ by expressing the result as a linear combination of column vectors:

\[
Ax = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} x_1 \oplus \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} x_2 \oplus \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} x_3 \oplus \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} x_4
\]

Example: $x = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}$, $Ax =$
Binary linear system

Recall a linear system of equations $Ax = y$ can be solved for $x$ by inverting the matrix $A$.

Now, suppose we have $A$ and $Ax$, how can we find $x$?

$$A = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad Ax = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \end{pmatrix}, \quad x = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

Hint: What does row 4 tell us about $x_1$? Then, what does row 1 tell us about $x_3$?
Binary linear system

We want to find a 3x4 binary matrix $B$ such that

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

For the matrix on the previous slide, we have

$$A = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{pmatrix}, \quad Ax = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \quad B(Ax) = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

Note: the choice of $B$ here is not unique. Why?
Block codes

A redundancy code will duplicate a single bit, possibly multiple times, to detect and correct errors.

A block code will add check bits for a whole block of data bits, allowing redundancy to be spread out among multiple input bits.
  ◦ This allows more favorable transmission rates for a certain error correction capability
  ◦ The block code defines a set of linear binary equations or constraints that the data and check bits must satisfy to be valid. These equations can be encoded as a binary matrix.

Example: 4 data bits, 3 check bits = 7-bit code block
  ◦ $2^7 = 128$ possible codes
  ◦ $2^4 = 16$ valid codes
Parity check code

Let’s start with a simple block code: seven data bits, and a single “parity” check bit:

\[ d_1 \oplus d_2 \oplus \cdots \oplus d_7 \oplus c_1 = 0 \quad \text{Even parity} \]

The parity bit is generated as \( c_1 = d_1 \oplus d_2 \oplus \cdots \oplus d_7 \)

If there is a single error, we can detect it. Can we correct it?

What about multiple errors?
Hamming code

The (7,4) Hamming code is a block code with 4 data bits and 3 check bits:

\[ c_1 = d_1 \oplus d_2 \oplus d_4 \]
\[ c_2 = d_1 \oplus d_3 \oplus d_4 \]
\[ c_3 = d_1 \oplus d_2 \oplus d_3 \]

Then, we transmit \([d_1, d_2, d_3, d_4, c_1, c_2, c_3]\).

The parity checks are:
\[ d_1 \oplus d_2 \oplus d_4 \oplus c_1 = 0 \]
\[ d_1 \oplus d_3 \oplus d_4 \oplus c_2 = 0 \]
\[ d_1 \oplus d_2 \oplus d_3 \oplus c_3 = 0 \]
Hamming code

Example: data bits = (1,0,1,1)

Check bits =

Hamming code:
Hamming code

What if there’s a single error? Can we correct it?

The (7,4) Hamming code can correct up to one error or detect up to two errors.

Possible codes:

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0000000</td>
<td>0001110</td>
<td>0010011</td>
<td>0011101</td>
<td></td>
</tr>
<tr>
<td>0100101</td>
<td>0101011</td>
<td>0110110</td>
<td>0111000</td>
<td></td>
</tr>
<tr>
<td>1000111</td>
<td>1001001</td>
<td>1010100</td>
<td>1011010</td>
<td></td>
</tr>
<tr>
<td>1100010</td>
<td>1101100</td>
<td>1110001</td>
<td>1111111</td>
<td></td>
</tr>
</tbody>
</table>

Hamming distance?
Hamming code

Better way than enumerating codes? Linear algebra!

Idea: generate a parity check matrix $A$ for this block code, where a valid (no errors) code satisfies the equation $Ax = 0$

\[
A = \begin{pmatrix}
1 & 1 & 0 & 1 & 1 & 0 & 0 \\
1 & 0 & 1 & 1 & 0 & 1 & 0 \\
1 & 1 & 1 & 0 & 0 & 0 & 1
\end{pmatrix}
\]

\[
x = \begin{pmatrix}
d_1 \\
d_2 \\
d_3 \\
d_4 \\
c_1 \\
c_2 \\
c_3
\end{pmatrix}
\]
Hamming code

Then, the parity check is the act of computing $s = Ax$ for the code $x$, where $s$ is called the syndrome.

- If the code is valid, the syndrome equals zero.
- If the code is not valid, the syndrome value tells us which parity check(s) are violated.

Example: code = (1,0,1,1,0,1,0):

$$s = Ax = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix}$$
Hamming code

Now let’s suppose there’s a transmission error. So \( x = (1,1,1,0,1,0) \). What’s the syndrome?

\[
s = Ax = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{pmatrix} \]

How about another erroneous code: \( x = (1,0,1,0,0,1,0) \). What’s the syndrome?

\[
s = Ax = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \end{pmatrix} \]
Hamming code

How do we find the error?

We know the erroneous code $x = \text{true code } x_{\text{true}} + \text{error } e$.

Since $s = Ax = A(x_{\text{true}} + e) = Ax_{\text{true}} + Ae$, by linearity.

Note: $x_{\text{true}}$ has no error, so what must $Ax_{\text{true}}$ be?
  ◦ This means that $x_{\text{true}}$ is in the null space of $A$!

Thus, the syndrome is $Ae$. If the error is a single “1” bit, then the syndrome is a single column of $A$. Thus, we just find the corresponding column of $A$ equal to $s$, and that is the erroneous bit.
  ◦ Thus, we can correct for a single bit error!
  ◦ What are we assuming about $A$? The columns must be distinct.
Hamming code

So we’ve proven the (7,4) code can detect and correct a single error.

Why can we detect a pair of errors? (but not necessarily correct them)

The (7,4) Hamming code works well when the errors are spaced far apart.
  ◦ “Burst” errors that occur in many-in-a-row are not handled well.

More robustness: interleave bits so parity checks are more spread out. This way, burst errors are easier to detect/correct.
Hamming code

We have constructed a Hamming code with 4 data bits and 3 check bits. There are more general codes:

For $r$ check bits, the Hamming code length can be up to $2^r - 1$ to detect and correct single errors. Why? Up to $2^r$ distinct syndromes with $r$ bits, and we need to reserve one (zero vector) for no error condition.

- This means there are up to $2^r - r - 1$ data bits.
- For $r = 3$, $2^3 - 3 - 1 = 4$ data bits
- For $r = 4$, $2^4 - 4 - 1 = 11$ data bits
- For $r = 5$, $2^5 - 5 - 1 = 26$ data bits
- More check bits/block enables more efficient encoding of data bits, but fixed single error correction limits error reduction.
- Another way of expressing this: for $m$ data bits, $r$ check bits, Hamming code satisfies $m + r < 2^r$
Hamming code

Example: generate parity check matrix for (15,11) Hamming code:

\[
\begin{pmatrix}
1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0
\end{pmatrix}
\]

- Last columns are for check bits, so use “identity matrix”
- Select remaining columns to be distinct from each other (and not all zeros)
Hamming code

For a Hamming code with $m$ data bits and $r$ check bits, we have significant redundancy to enable error detection:

- $2^{m+r}$ possible codes, but only $2^m$ valid codes

What about other parity-check codes? These codes can also be described by a parity check matrix:

- Single-parity check: $A = \begin{pmatrix} 1 & 1 & 1 & 1 \end{pmatrix}$
- Triple-redundancy check: $A = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}$
Correcting multiple errors

Hamming code can correct one error. How to correct for multiple errors with a parity check code?

◦ For two errors, we need a parity check matrix such that the sum of every pair of columns is different (different syndrome for every pair of errors).

◦ For three errors, we need the sum of every three columns to be different

◦ How does this relate to linear dependence/independence of groups of columns?

Example: quintuple-redundancy check can correct two errors:

\[ A = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix} \]
Your turn

Design a Hamming code for a block containing at least 15 data bits:
Announcements

Next class: Midterm #2 (review in a bit)

Lecture next week: Multiple access communication
Midterm #2

When: Thursday, March 28, during class (9:30 – 10:45 AM)

Where: In class (Olsson 120)

What: All the material up through and including last lecture (on linear algebra). This exam is comprehensive, so you might reuse your note sheet from last test to help study.
  ◦ We’ll review in class on Tuesday.

Policies:
  ◦ Bring two sheets (single sided 8½ x 11”) of notes, no photocopies allowed on the note sheets
  ◦ No books or other course materials are allowed
  ◦ Calculators are welcome but unnecessary (this is not a test on how to use a calculator)
  ◦ Make-up: please notify Prof. Weller ahead of time (if possible); being busy is not an excuse
Midterm #2 review