Math 75 notes, Lecture 13 outline

P. Pollack and C. Pomerance

References below are to Pretzel's *Error-correcting codes and finite fields*:

- We saw that the minimum distance $d(C)$ of a linear code (over a finite field F) is the same as the minimum weight of a nonzero codeword (p. 32).
- We reviewed the theorem from the previous lecture that one should be able to detect up to $d(C) - 1$ errors for the code C, and up to $\frac{1}{2}(d(C) - 1)$ errors can be corrected. We illustrated this for our three example codes: the (8,7) parity check, the (9,3) triple repetition, and the (6,3) triple check.
- We discussed four fundamental algorithmic problems in connection with coding theory:

  (1) Have an encoding algorithm, which is a function that sends "real" words (vectors in F^m) to code words (vectors in $C \subseteq F^n$). It should be a one-to-one correspondence between F^m and C.

  (2) Have a decoding algorithm, which is the inverse function of the encoding algorithm. That is, it sends code words to real words.

  (3) Be able to recognize when a word in F^n is a code word or not (error detection).

  (4) Be able to find the closest code word to a given word in F^n if there is a unique closest code word (error correction).
- We usually have in the case of a linear code, an encoding function that is a linear transformation from the vector space F^m to the vector space C (in F^n). Say this transformation is denoted E (for encoder). (Section 3.6)
- Corresponding to given basis vectors in F^m and in C, we have a generator matrix; it is $n \times m$ with entries in the finite field F. (Section 3.7)
- We usually use the standard basis $e_1, e_2, \ldots, e_m$ of F^m. Note that e_i has 1 in the ith place and 0's in the other $m - 1$ places. (The book calls these "unit words.") Then the basis for C is just $E(e_i)$ for $i = 1, \ldots, m$. The generator matrix then has for its ith column the code word $E(e_i)$. (Section 3.8)