Math 127, Wed Mar 10

- Use a laptop or desktop with a large screen so you can read these words clearly.
- In general, please turn off your camera and mute yourself.
- Exception: When we do groupwork, please turn both your camera and mic on. (Groupwork will not be recorded.)
- Please always have the chat window open to ask questions.
- Reading for today: 6.1–6.2.
- Reading for ~~Wed~~: 6.3–6.4. Mon
- PS05 outline due tonight, full version due Mon Mar 15.
- Problem session Fri Mar 12, 10am–noon.

Linear algebra: Questions to resolve

- Is it possible for a subspace W to have one basis with 5 vectors and another basis with 7 vectors? In other words, is it possible for the dimension of W to be both 5 and 7?
- Is it possible for F^8 to contain a subspace of dimension 10? In other words, is it possible for a smaller space to have a larger dimension?
- Can we find a subspace of F^n that doesn't have a basis at all?

Math 39: OK for $F = \mathbb{R}$ ✓

OK for $F = \mathbb{F}_2$?

Thank goodness, it all works

F field $\quad s\overbrace{\left[\text{fat}\right]}^{\ell}$

Theorem (Comparison Theorem)

Let W be a subspace of F^n. If $\{\mathbf{v}_1, \ldots, \mathbf{v}_s\}$ spans W and $\{\mathbf{w}_1, \ldots, \mathbf{w}_\ell\}$ is a linearly independent subset of W, then $\ell \leq s$.

I.e.: **ANY** linearly independent subset is no larger than **ANY** spanning set.

Why: If $s < \ell$, then we can set up with s linear equations in ℓ variables, which must have a nonzero solution. That nonzero solution contradicts linear independence of $\{\mathbf{w}_1, \ldots, \mathbf{w}_\ell\}$.

l.i. sets

span sets

bases

Consequences of Comparison Thm

W subsp.

Corollary (Dimension Theorem)

Any two bases for W must have the same size k (i.e., W cannot have more than one dimension).

lin ind & span W.

Proof: S'pose

$\{\underline{v}_1 \ldots \underline{v}_k\}$ basis for W

$\{\underline{w}_1 \ldots \underline{w}_m\}$ " " W

$\underline{v}$'s span, $\underline{w}$'s lin ind $\Rightarrow k \geq m$.

$\underline{w}$'s span, $\underline{v}$'s lin ind $\Rightarrow m \geq k$

$\Rightarrow m = k$

Corollary

If $\dim W = k$, *any linearly independent set must have size* $\leq k$ *and any span set must have size* $\geq k$.

Proof: PS05.

So how can we be sure that every subspace has a basis?

Definition

Let W be a subspace of F^n. A **maximal linearly independent subset of** W is a linearly independent subset $\{\mathbf{v}_1, \ldots, \mathbf{v}_k\}$ of W such that for any $\mathbf{x} \in W$, $\{\mathbf{v}_1, \ldots, \mathbf{v}_k, \mathbf{x}\}$ is linearly dependent.

Theorem

Let W be a subspace of F^n, and suppose $\{\mathbf{v}_1, \ldots, \mathbf{v}_k\}$ is a maxmimal linearly independent subset of W. Then $\{\mathbf{v}_1, \ldots, \mathbf{v}_k\}$ is a basis for W.

(proof omitted)

Corollary

If W is a subspace of F^n, then W has a basis.

One more consequence

Corollary (Subspace Size Theorem)

If W is a subspace of a subspace V of F^n, then $\dim W \leq \dim V \leq n$. In particular, any subspace of F^n has dimension at most n.

$W \subseteq V$

(A) $\dim W = k$ $\dim V = m$

Since dim W = k, there exists a basis B for W with k vectors in it.
Since dim V = m, there exists a basis B' for V with m vectors in it.
We know that the k vectors in B span W, and that they are linearly independent.
We also know that the m vectors in B' span V and that are linearly independent.
So: The m vectors in B' span V, and the k vectors in B are a linearly independent set inside V.
By Comparison Theorem, the spanning set B' is bigger than the linearly independent set B, i.e., m >= k.

(C) $k \leq m$

$\{\underline{v}_1, \ldots, \underline{v}_k\}$ in W

dimension

basis

span | lin ind

lin comb

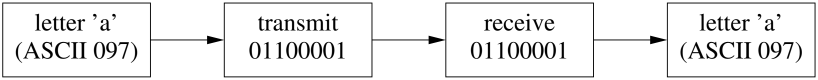
Every $\underline{x} \in W$ is l.c of $\{\underline{v}_1 \ldots \underline{v}_k\}$

Only l.c. of $\{\underline{v}_1, \ldots \underline{v}_k\} = \underline{0}$ is all-0's.

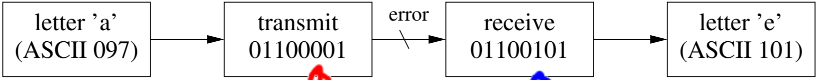
Properties of bases: You can use a basis to list all vectors in W, by taking all linear combinations of vectors in that basis. I.e., by definition, those vectors span W.

How to send an 'a'

Sending an 'a' over a communications channel:



But mistakes happen:



Motivating Problem

Is there some way that we can detect that an error or errors has occurred? Better yet, is there some way that we can correct an error or errors?

Parity check code

$\mathbb{F}_2 = \{0, 1\}$

Suppose we have n data bits $x_1, \ldots, x_n$ $\in \mathbb{F}_2$ to transmit. We can add a **parity check** bit

$$x_0 = x_1 + \cdots + x_n \pmod{2}$$

to our message, and transmit the $(x_0, x_1, \ldots, x_n)$. Note:

Any error-free msg must satisfy this linear eqn.

$$x_0 + x_1 + \cdots + x_n = 0 \quad (\text{in } \mathbf{F}_2)$$

$+1 = -1$

Example: $n = 7$

msg: 0 1 1 0 1 0 1 (x_1 … x_7)

$x_0 = x_1 + \cdots + x_7$ $\leftarrow 0$

transm: 0 0 1 1 0 1 0 1

recv: 0 0 1 1 0 0 0 1 → check $0+0+1+1+0+0+0+1$ = 1

read: error

Two errors

0110101

00110101

00100001 → sum = 0

read 0100001

Parity check code detects one error, but not two.

Example: Repetition code

Can we do better and **correct** an error in transmission? Yes, with the **repetition code**.

$x = 1$

Suppose we want to transmit one data bit $x \in \mathbf{F}_2$. We repeat x three times:

msg 1

transm. 111

So if one error occurs in transmission:

recv 101

read 1

we can fix it because other two bits still correct (**majority logic**).
I.e., we can correct one error at the cost of transmitting 3 times as much data.

Q: Can we correct errors more cheaply?

Binary linear codes

Definition

We define a **bit** to be an element of $\mathbf{F}_2$, and we define a **bitstring of length** n to be an element of $\mathbf{F}_2^n$.

Definition

A **code** is a subset $\mathcal{C}$ of $\mathbf{F}_2^n$. Elements (vectors) of a code are called **codewords**.

I.e., codewords are the possible messages that could have been transmitted without errors. The idea is that if you pick your code well, it should be possible to tell if an error has occurred, and if you pick it really well, it should be possible to correct the error.

Definition

A **binary linear code** $\mathcal{C}$ **of length** n is a subspace $\mathcal{C}$ of $\mathbf{F}_2^n$.

Standard framework for discussing codes

$$\boxed{\begin{array}{c}\text{Intended msg}\\ \mathbf{m}\end{array}} \xrightarrow{\text{encode}} \boxed{\begin{array}{c}\text{Transmit}\\ \mathbf{x}\end{array}} \xrightarrow{\text{errors}} \boxed{\begin{array}{c}\text{Receive}\\ \mathbf{y}\end{array}} \xrightarrow{\text{decode}} \boxed{\begin{array}{c}\text{Read msg}\\ \mathbf{m}'\end{array}}$$

1. Xavier wants to send a bitstring $\mathbf{m}$.
2. Xavier **encodes** the message $\mathbf{m}$ to some codeword $\mathbf{x} \in \mathcal{C}$.
3. Xavier transmits $\mathbf{x}$, Yolanda receives $\mathbf{y}$.
4. Yoland **decodes** $\mathbf{y}$ to the message $\mathbf{m}'$, in steps:
   - First, Yolanda **corrects** $\mathbf{y}$ to a valid codeword $\mathbf{y}' \in \mathcal{C}$.
   - Yolanda then **reads** $\mathbf{y}'$ as a message $\mathbf{m}'$.

Algebraic model for errors: Let $\mathbf{e}_i$ be the vector in $\mathbf{F}_2^n$ whose ith coordinate is 1 and whose other coordinates are all 0. One error in bit i means:

$$\mathbf{y} = \mathbf{x} + \mathbf{e}_i.$$

Two errors in bits i and j:

$$\mathbf{y} = \mathbf{x} + \mathbf{e}_i + \mathbf{e}_j.$$

How to define/describe a binary linear code

Two ways:

Definition

Let G be an $n \times k$ matrix over $\mathbf{F}_2$. To say that G is the **generator matrix** of $\mathcal{C}$ of length n means that $\mathcal{C} = \mathrm{Col}(G)$.

Definition

Let H be a $k \times n$ matrix over $\mathbf{F}_2$. To say that H is the **parity check matrix** of a binary linear code $\mathcal{C}$ of length n means that $\mathcal{C} = \mathrm{Null}(H)$.

I.e.: A generator matrix defines a code as a column space, and a parity check matrix defines a code as a nullspace.

Back to our examples

Parity check code: The parity check code of length $n+1$ is the nullspace $\mathcal{C}$ of the $1 \times (n+1)$ matrix $H = [1 \ \ldots \ 1]$. In other words, $\mathbf{x} \in \mathbf{F}_2^{n+1}$ is in $\mathcal{C}$ exactly when $H\mathbf{x} = 0$.
Encoding is $x_0 = x_1 + \cdots + x_n$, transmit $(x_0, x_1, \ldots, x_n)$. If received message $\mathbf{y}$ satisfies $H\mathbf{y} = \mathbf{0}$, read off bits $x_1, \ldots, x_n$ as message; otherwise notify Xavier that there was an error.

Repetition code: The repetition code of length n is the span $\mathcal{C}$ of the column of the generator matrix $G = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}$.
Encoding the bit x means multiplying Gx, transmit Gx, correct received bits by majority logic, then use any bit as the message bit.

Error-correcting codes in practice

TN-29-63 Error Correction Code (ECC) in SLC NAND
Systematic Codes

Figure 1: Block Codes