Finite fields

I talked in class about the field with two elements

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

and we've used it in various examples and homework problems. In these notes I will introduce more finite fields

$$\mathbb{F}_p = \{0, 1, \ldots, p-1\}$$

for every prime number p. I'll say a little about what linear algebra looks like over these fields, and why you might care.

First problem is the definition. One of the basic features of what you learned about in elementary school about adding and multiplying integers is that *the last digit of the answer only depends on the last digits in the problem.* So if I try to tell you that $27 \cdot 38 = 1028$, you know immediately that there's a problem: because $7 \cdot 8 = 56$, the answer *must end in 6.* In this way we can define addition and multiplication *modulo ten.*

Definition 1. *Suppose $0 \leq a \leq 9$ and $0 \leq b \leq 9$ are integers. Choose any positive integers A and B with last digits a and b respectively. Write x for the last digit of $X = A + B$, and y for the last digit of $Y = A \cdot B$. Then addition and multiplication modulo 10 are defined by*

$$a +_{10} b = x, \qquad a \cdot_{10} b = y.$$

Write $\mathbb{Z}/10\mathbb{Z}$ for the set $\{0, 1, \ldots, 9\}$ endowed with this addition and multiplication.

If the context makes the 10 unambiguous, one usually writes just $+$ and $\cdot$ for addition and multiplication modulo 10.

The content of the definition is that the sum and product modulo ten are well-defined. Then we could say for example

$$\text{because } 13 \cdot 29 = 377,\ 3 \cdot_{10} 9 = 7.$$

It's an elementary algebra exercise to check that addtion and multiplication are really well-defined (that is, independent of the choices of A and B); I won't do that. The commutative, associative, and distributive laws are all inherited from $\mathbb{Z}$, so they are true in $\mathbb{Z}/10\mathbb{Z}$. The element 0 is an additive identity, and additive inverses exist; and 1 is a multiplicative identity. The only axiom for a field that is missing is the existence of multiplicative inverses. Some of these inverses exist, even for elements having no multiplicative inverse in $\mathbb{Z}$: for example $3 \cdot_{10} 7 = 1$, so 7 is a multiplicative inverse of 3 in $\mathbb{Z}/10\mathbb{Z}$.

But there is trouble right here in River City: $2 \cdot_{10} 5 = 0$, and it follows easily that the nonzero elements 2 and 5 have no multiplicative inverses. (Can you tell which elements of $\mathbb{Z}/10\mathbb{Z}$ *do* have multiplicative inverses?) So $\mathbb{Z}/10\mathbb{Z}$ is not a field.

There was nothing special about 10 in this discussion. If n is any integer greater than 1, we can make

Definition 2. *Suppose $0 \leq a < n$ and $0 \leq b < n$ are integers. Choose any positive integers A and B with last digits in base n equal to a and b respectively. (This means that the remainder when A is divided by n is equal to a.) Write x for the last base n digit of $X = A + B$, and y for the last base n digit of $Y = A \cdot B$. Then addition and multiplication modulo n are defined by*

$$a +_n b = x, \qquad a \cdot_n b = y.$$

Another way to say this is that x is the remainder when $a + b$ (or $A + B$) is divided by n; and y is the remainder when $a \cdot b$ (or $A \cdot B$) is divided by n. Write $\mathbb{Z}/n\mathbb{Z}$ for the set $\{0, 1, \ldots, n-1\}$ endowed with this addition and multiplication.

The addition and multiplication in $\mathbb{Z}/n\mathbb{Z}$ are commutative and associative and distributive, and we have identities $0 \neq 1$ and additive inverses. (The reason $0 \neq 1$ is that we are assuming $n > 1$.) The only question (for deciding whether $\mathbb{Z}/n\mathbb{Z}$ is a field) is whether nonzero elements have multiplicative inverses.

Theorem 3. *With the addition and multiplication just defined, $\mathbb{Z}/n\mathbb{Z}$ is a field if and only if n is a prime number.*

Proof. Suppose first that n is not prime: say $n = r \cdot s$, with $1 < r, s < n$. Then $r \cdot_n s = 0$, and it follows easily that r and s cannot have multiplicative inverses modulo n. So $\mathbb{Z}/n\mathbb{Z}$ is not a field.

Now assume that $n = p$ is a prime number. We can't make counterexamples in this way, but there could be a more subtle reason for $\mathbb{Z}/p\mathbb{Z}$ not to be a field: we need to prove that every nonzero element x of $\mathbb{Z}/p\mathbb{Z}$ really has a multiplicative inverse.

A basic fact about prime numbers and multiplication of integers is

if x and y are integers not divisible by p, then xy is not divisible by p.

In terms of multiplication in $\mathbb{Z}/p\mathbb{Z}$, this means

if x and y are nonzero in $\mathbb{Z}/p\mathbb{Z}$, then $x \cdot_p y$ is not zero.

Using the distributive law, we can translate this formulation to

if $0 \neq x \in \mathbb{Z}/p\mathbb{Z}$ and $z \neq z' \in \mathbb{Z}/p\mathbb{Z}$, then $x \cdot_p z \neq x \cdot_p z'$.

That is, if x is not zero in $\mathbb{Z}/p\mathbb{Z}$, then the p multiples of x

$$\{x \cdot_p 0, x \cdot_p 1, x \cdot_p 2, \ldots x \cdot_p (p-1)\}$$

must all be *distinct*. Therefore they must be *all* of the p elements of $\mathbb{Z}/p\mathbb{Z}$. In particular, one of them must be equal to 1: there is a z with

$$x \cdot_p z = 1.$$

This element z is the multiplicative inverse of x. **QED**

The field $\mathbb{Z}/p\mathbb{Z}$ is called $\mathbb{F}_p$. Here is a result which connects finite fields with counting problems, and is one of the reasons they are so interesting.

Theorem 4. *Suppose V is an m-dimensional vector space over $\mathbb{F}_p$.*
a) The cardinality of V is $|V| = p^m$.

Suppose $T: V \to W$ is a linear map. Write n for the dimension of the null space of T, and r for the dimension of the range.

b) The cardinality of the range of T is p^r.
c) The preimage of every vector in the range of T has p^n elements.

This exhibits V as the union of p^r disjoint pieces, each of size p^n; so V has $p^r \cdot p^n = p^{n+r} = p^m$ elements, (where $n + r = m$ is Theorem 3.4 in the text).

So what finite fields can exist? Suppose F is any finite field. Start with the element $0 \in F$ and add $1 \in F$ repeatedly, getting a string

$$0,\ 1,\ 1+1,\ 1+1+1,\ 1+1+1+1, \ldots$$

of elements of F. The first two elements are distinct because of the axiom $0 \neq 1$ for a field. But the field is finite, so the string has to repeat itself eventually. It isn't hard to see that the string must be *periodic* of some period $n > 1$: that the first n terms (which could naturally be called $(0, 1, 2, \ldots, n-1)$) are all distinct, and then

$$(n-1) + 1 = 0,$$

and the sequence repeats. The number n defined in this way is called the *characteristic* of the finite field F. It's not hard to see

Theorem 5. *Suppose n is the characteristic of the finite field F. Then F contains $\mathbb{Z}/n\mathbb{Z}$, with the addition and multiplication given in Definition 2. In particular (because F has multiplicative inverses) the characteristic must be a prime number p, and so F contains $\mathbb{F}_p$.*

The field F is then a vector space of some positive (since $1 \neq 0$) dimension m over $\mathbb{F}_p$; so $|F| = p^m$.

This much is easy. What's a bit more subtle is

Theorem 6. *Suppose p is a prime number, m is a positive integer, and $q = p^m$. Then there is (up to isomorphism) exactly one field $\mathbb{F}_q$ having p^m elements.*

Neither the existence nor the uniqueness of $\mathbb{F}_q$ is obvious. Here are the addition and multiplication tables for $\mathbb{F}_4 = \{0, 1, x, x+1\}$.

$+$	0	1	x	$1+x$
0	0	1	x	$1+x$
1	1	0	$1+x$	x
x	x	$1+x$	0	1
$1+x$	$1+x$	x	1	0

Addition in $\mathbb{F}_4$

$\cdot$	0	1	x	$1+x$
0	0	0	0	0
1	0	1	x	$1+x$
x	0	x	$1+x$	1
$1+x$	0	$1+x$	1	x

Multiplication in $\mathbb{F}_4$

Computing in finite fields

Addition and multiplication in $\mathbb{F}_p$ can be done using the algorithms for $\mathbb{Z}$, followed by computing the remainder after division by p.

In order to do linear algebra, you also need to be able to *invert* elements of $\mathbb{F}_p$. The proof above of the existence of multiplicative inverses is not constructive. If you want to write a program to do linear algebra in $\mathbb{F}_{379721}$, you don't want to calculate the inverse of 17 by trying all 379720 nonzero elements of the field. One way to proceed is using the *(extended) Euclidean algorithm*. This requires approximately $\log p$ steps, in each of which the most complicated is a division-with-remainder of integers smaller than p (a calculation requiring a some multiple of $\log p$ steps). This could be a reasonable way to calculate inverses for p of size around 2^{32} or 2^{64}. For much larger primes, the division-with-remainder steps can slow things down enough that other tricks are useful.

There is a (not very difficult) theorem saying that if $0 \neq a \in \mathbb{F}_q$, then

$$a^{-1} = a^{q-2}.$$

So you can compute the inverse as a power. At first glance this looks slow, since it asks for $q-3$ multiplications. But the right way to compute powers is to compute the 2^k powers using

$$a^{2^{k+1}} = a^{2^k} \cdot a^{2^k};$$

so $\log_2 q$ multiplications to compute all of the 2^k powers of a for $2^k \leq q-2$. Now write $q-2$ in base 2, and multiply together the a^{2^k} for k corresponding to the nonzero bits in $q-2$: for example,

$$a^{25} = a^{16} \cdot a^8 \cdot a.$$

This is about $\log_2 q$ additional multiplications, for a total of $2\log_2 q$ multiplications in $\mathbb{F}_q$ to compute a^{-1}. At first glance this looks comparable to the Euclidean algorithm in speed, but I don't know anything about practice.[1]

For relatively small finite fields (I've seen this done with $q \leq 2^8$) you can make a double array of values of $+$ and $\times$, and do arithmetic by looking up elements of the array.

Why do this?

I know of two completely different reasons. First, many problems in mathematics concern *integers*: how many integer solutions are there to some equation? An integer solution automatically provides a solution in $\mathbb{Z}/n\mathbb{Z}$ for every positive integer n, and in particular a solution in $\mathbb{F}_p$ for every prime number p. So people study equations in $\mathbb{F}_p$ and try to use what they learn to say something about integer solutions. This idea has been fantastically successful, in extremely surprising ways. (My own research involves some differential equations that can arise from quantum mechanics; I try to understand what kinds of solutions those equations can have mathematically, in the hope that the mathematically interesting solutions might have some meaning in physics. The best answers to those questions that we know involve counting solutions over finite fields.)

[1] Yogi Berra said, "In theory there is no difference between theory and practice. In practice there is."

A second reason is just as a trick for computational efficiency: computations with big integers can be replaced by (more) computations with small integers, saving memory. Here is an example. Suppose you want to solve a linear algebra problem (like a system of a hundred simultaneous equations in a hundred unknowns, with integer coefficients). Suppose you know that all the (ten thousand) coefficients in your equations are smaller than 2^{63}, and that your hundred integer solutions are all smaller than 2^{63}. You need 8 bytes of memory for each of the $10,000$ coefficients, and 8 for each of the 100 solutions: $80,800$ bytes of memory altogether. (That's not so much, but you can change 100 to a billion if you like.)

Here's another way to proceed. Solve these same equations in $\mathbb{Z}/n\mathbb{Z}$, for each of the nine values

$$n = 229,\ 233,\ 239,\ 241,\ 247,\ 251,\ 253,\ 255,\ 256.$$

At each step, each number in your calculation is smaller than 256, and so fits in one byte of memory; so the calculation requires just 10,100 bytes of memory (saving a factor of eight). Because the arithmetic involves smaller numbers, you can hope that each calculation is faster than the original with 8-byte integers; but in any case you've slowed down by no worse than a factor of nine.

At the end, your nine solutions in $\mathbb{Z}/n\mathbb{Z}$ can be combined by the Chinese Remainder Theorem to give a solution in $\mathbb{Z}/N\mathbb{Z}$, where

$$N = 229 \cdot 233 \cdot 239 \cdot 241 \cdot 247 \cdot 251 \cdot 253 \cdot 255 \cdot 256.$$

This combined solution is the reduction module N of the actual integer solution you wanted. Because $N > 2^{64}$, there is no reduction: you have found an exact integer solution of your problem.

So you have managed to reduce your use of memory by a factor of 8 at the cost of increasing time by a factor of about 9 (or less if you can do small arithmetic faster). This is sometimes a good bargain.