ProblemSet 4 – Finite field projective spaces **Solutions**

George McNinch

2024-02-23

1. Find the irreducible factors of the polynomial $T^9 - 1$ in $\mathbb{F}_7[T]$.

   (You should include proofs that the factors you describe are irreducible).

   Note that the multiplicative group $\mathbb{F}_7^\times$ has order 6 and hence contains an element of order 3; in fact, 2 has order 3 since $2^3 = 8 \equiv 1 \pmod 7$.

   Now, $\mathbb{F}_{7^2}^\times$ has order $49 - 1 = 48$ which is not divisible by 9. And $\mathbb{F}_{7^3}^\times$ has order $7^3 - 1 \equiv (-2)^3 - 1 \equiv -9 \equiv 0 \pmod 9$. So $\mathbb{F}_{7^3}^\times$ has an element of order 9.

   Consider the polynomial $T^3 - 2$. Any root α of this polynomial satisfies $\alpha^3 = 2$ and $\alpha^9 = 1$; this shows that the multiplicative order of α is 9.

   In particular, $\mathbb{F}_7$ contains no roots of $f(T) = T^3 - 2$; since $f(T)$ has degree 3, it is irreducible over $\mathbb{F}_7$.

   If α is a root of $f(T)$, then α^7 and $\alpha^{7^2} = \alpha^4$ are also roots (note that $7^2 \equiv (-2)^2 = 4 \pmod 9$). Thus

   $$f(T) = (T - \alpha)(T - \alpha^7)(T - \alpha^4)$$

   and

   $$f(T) \mid T^9 - 1.$$

   Notice that $\mathbb{F}_{7^3}$ is a splitting field for $f(T)$ over $\mathbb{F}_7$.

   Note that $2^2 = 4$ is also an element of $\mathbb{F}_7^\times$ of order 3. Arguing as before, any root of $T^3 - 4$ is an element of multiplicative order 9.

   On the other hand, since $\gcd(2, 9) = 1$, $\alpha^2 \in \mathbb{F}_{7^3}$ is also an element of order 9.

   Moreover, the roots of its minimal polynomial $g(T)$ have the form α^2, $\alpha^{2\cdot 7} = \alpha^5$ (since $14 \equiv 5 \pmod 9$), and $\alpha^{2\cdot 7^2} = \alpha^{5\cdot 7} = \alpha^3$ (since $2 \cdot 7^2 \equiv 8 \pmod 9$).

   Thus

   $$g(T) = (T - \alpha^2)(T - \alpha^5)(T - \alpha^8).$$

   Now, notice that $(\alpha^2)^3 = (\alpha^3)^2 = 2^2 = 4 \in \mathbb{F}_7$. Thus the minimal polynomial $g(T)$ of α^2 divides $T^3 - 4$. It follows that

   $$g(T) = T^3 - 4 = (T - \alpha^2)(T - \alpha^5)(T - \alpha^8).$$

   Now, $g(T) \mid T^9 - 1$ and since $\gcd(f(T), g(T)) = 1$ we see that $f(T)g(T) \mid T^9 - 1$. Thus

   $$T^9 - 1 = f(T) \cdot g(T) \cdot (T - 1) \cdot (T - 2) \cdot (T - 4).$$

2. Let $0 < k, m \in \mathbb{N}$, put $n = mk$, and consider the subspace $C \subset \mathbb{F}_q^n$ defined by

   $$C = \{(v, v, \cdots, v) \mid v \in \mathbb{F}_q^k\} \subset \mathbb{F}_q^n.$$

   Find the *minimal distance* d of this code.

   For example, if $n = 6$, $k = 3$ and $m = 2$ then

   $$C = \{(a_1, a_2, a_3, a_1, a_2, a_3) \mid a_i \in \mathbb{F}_q\} \subset \mathbb{F}_q^6.$$

(Corrected)

If $\mathbf{v} = (v, v, \cdots, v) \in C$ for $v \in \mathbb{F}_q^n$, note that weight$(\mathbf{v}) = m \cdot \text{weight}(v)$.

In particular, for a non-zero vector we see that weight$(\mathbf{v}) \geq m$.

On the other hand, a standard basis vector $v = \mathbf{e}_i \in \mathbb{F}_q^n$ has weight 1, so if $\mathbf{w} = (\mathbf{e}_i, \mathbf{e}_i, \cdots, \mathbf{e}_i)$, then weight$(\mathbf{w}) = m$.

Thus

$$\min\{\text{weight}(\mathbf{v}) \mid 0 \neq \mathbf{v} \in C\} = m.$$

For a linear code, the minimal distance is simply the minimal weight of a non-zero vector; thus the minimal distance of C is m.

3. By an $[n, k, d]_q$-system we mean a pair $(V, \mathcal{P})$, where V is a finite dimensional vector space over $\mathbb{F}_q$ and $\mathcal{P}$ is an ordered finite family

$$\mathcal{P} = (P_1, P_2, \cdots, P_n)$$

of points in V (in general, points of $\mathcal{P}$ need not be distinct – you should view $\mathcal{P}$ as a *list* of points which may contain repetitions) such that $\mathcal{P}$ spans V as a vector space. Evidently $|\mathcal{P}| \geq \dim V$.

The parameters $[n, k, d]$ are defined by

$$n = |\mathcal{P}|, \quad k = \dim V, \quad d = n - \max_H |\mathcal{P} \cap H|.$$

where the maximum defining d is taken over all linear hyperplanes $H \subset V$ and where points are counted with their multiplicity – i.e. $|\mathcal{P} \cap H| = |\{i \mid P_i \in H\}|$.

Gjven a $[n, k, d]_q$-system $(V, \mathcal{P})$, let V^* denote the dual space to V and consider the linear mapping

$$\Phi : V^* \to \mathbb{F}_q^n$$

defined by

$$\Phi(\psi) = (\psi(P_1), \cdots, \psi(P_n)).$$

a. Show that Φ is injective.

Φ is a linear mapping, so we just need to show that $\ker \Phi = \{0\}$.

Suppose that $\psi \in V^*$ and $\Phi(\psi) = 0$. This means that $\psi(P_j) = 0$ for $1 \leq j \leq n$. Since ψ is linear, it follows that ψ vanishes at any linear combination of the vectors $\{P_j\}$.

Since $\mathcal{P}$ *spans* V by assumption, it follows that $\psi = 0$. This proves that Φ is injective.

b. Write $C = \Phi(V^*)$ for the image of Φ, so that C is an $[n, k]_q$-code. Show that the minimal distance of the code C is given by d.

Write d' for the minimal weight of C; we must argue that

$$d' = d = n - \max_H |\mathcal{P} \cap H|.$$

Let $\mathbf{v} = \Phi(\psi) \in C$ be a non-zero vector. We have

$$\text{weight}(\mathbf{v}) = |\{j \mid \psi(P_j) \neq 0\}|.$$

Write $H = \ker \psi$ and note that

$$|\mathcal{P} \cap H| = |\{j \mid \psi(P_j) = 0\}|.$$

Thus

$$(*) \quad \text{weight}(\mathbf{v}) = n - |\mathcal{P} \cap H|.$$

In $\max_H |\mathcal{P} \cap H|$ the *hyperplanes* H are precisely the kernels $H = \ker \psi$ of functionals $0 \neq \psi \in V^*$. Thus $(*)$ shows that

$$\min_{\mathbf{v}=\Phi(\psi)\neq 0} \text{weight}(\mathbf{v}) = n - \max_{H=\ker \psi, \psi \neq 0} |\mathcal{P} \cap H|;$$

it follows that $d' = d$.

c. Conversely, let $C \subset \mathbb{F}_q^n$ be an $[n,k,d]_q$-code, and put $V = C^*$. Let $e^1, \cdots, e^n \in (\mathbb{F}_q^n)^*$ be the dual basis to the standard basis. The restriction of e^i to the subspace C determines an element P_i of $C^* = V$. Write $\mathcal{P} = (P_1, P_2, \cdots, P_n)$ for the resulting list of vectors in V..

Prove that the minimum distance d of the code C satisfies

$$d = n - \max_H |\mathcal{P} \cap H|.$$

We have $V^* = (C^*)^* = C$; the mapping $\Phi : V^* = C \to \mathbb{F}_q^n$ is just the given inclusion. Indeed, let $x = (x_1, x_2, \cdots, x_n) \in C \subset \mathbb{F}_q^n$. The mapping $\Phi : V^* \to \mathbb{F}_q^n$ is given by $\Phi(x) = (e^1(x), \cdots, e^n(x)) = (x_1, \dots, x_n)$.

Now the equality

$$d = n - \max_H |\mathcal{P} \cap H|$$

follows from the result of part (b).

4. Let C be the linear code over $\mathbb{F}_5$ generated by the matrix

$$G = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 2 \\ 0 & 1 & 0 & 1 & 2 & 1 \\ 0 & 0 & 1 & 2 & 1 & 1 \end{pmatrix}.$$

a. Find a *check matrix* H for C.

```
k = GF(5)
V = VectorSpace(k,6)

C =V.subspace([ V([1,0,0,1,1,2]),
                V([0,1,0,1,2,1]),
                V([0,0,1,2,1,1])])

# generator matrix, in standard form
G = MatrixSpace(k,3,6).matrix(C.basis())
G
=>
[1 0 0 1 1 2]
[0 1 0 1 2 1]
[0 0 1 2 1 1]

A = MatrixSpace(k,3,3).matrix([b[3:6] for b in G])

# construct the check matrix, as a block matrix
H = block_matrix([[-A.transpose(),
                   MatrixSpace(k,3,3).one()]],
               subdivide=False)

H
=>
[4 4 3 1 0 0]
[4 3 4 0 1 0]
[3 4 4 0 0 1]

## verification:
H * G.T
=>
[0 0 0]
[0 0 0]
[0 0 0]
```

b. Find the minimum distance of C.

The minimal distance of C is 4.

We check the weight of a vector using the following function:

```
def weight(v):
    r = [x for x in v if x != 0]
    return len(r)
```

Now, we can just find the minimal weight of the non-zero vectors of V, as follows:

```
min([ weight(v) for v in C if v != 0])
=>
4
```

Alternatively, you can investigate the columns of the check matrix H.

```
W = VectorSpace(k,3)  # column space

# return the ith column of the 3xm matrix M
def col(M,i):
    return W([ b[i] for b in M ])

# check whether the columns of the 3xm matrix M
# specified by the list ll of indices are lin indep
def cols_lin_indep(M,ll):
    vecs =  [ col(M,i) for i in ll ]

    # the method `linear_dependence` returns a list
    # of *linear relations*

    # so we return True if `W.linear_dependence(vecs)` is
    # the empty list

    return W.linear_dependence(vecs) == []

# check whether all collections of r columns of the
# 3xm matrix M are linearly independent
def check(M,r):
    # get the number of columns of M.
    l = len(list(M.T))

    # get all lists of r-element subses of the numbers 0,...,l-1
    al = map(list,Subsets(range(l),r))

    # return True iff `cols_lin_indep(M,ll)` is true for every
    # r-element subset ll of range(l)
    return all([ cols_lin_indep(M,ll) for ll in al])

check(H,3)
=>
True

check(H,4)
=>
False
```

This shows that every collection of 3 columns of `H` is linearly independent, while there is some collection of 4 columns of `H` that is linearly dependent; thus $d = 4$.

c. Decode the received vectors $(0, 2, 3, 4, 3, 2)$ and $(0, 1, 2, 0, 4, 0)$ using syndrome decoding.

The minimal distance of the code C is 4, so we should expect to correct $\lfloor (4-1)/2 \rfloor = \lfloor 3/2 \rfloor = 1$ error.

We first make the lookup table

```
lookup = { tuple(H*v):v for v in V if weight(v) <= 1 }
lookup
=>
{(0, 0, 0): (0, 0, 0, 0, 0, 0),
 (4, 4, 3): (1, 0, 0, 0, 0, 0),
 (3, 3, 1): (2, 0, 0, 0, 0, 0),
 (2, 2, 4): (3, 0, 0, 0, 0, 0),
 (1, 1, 2): (4, 0, 0, 0, 0, 0),
 (4, 3, 4): (0, 1, 0, 0, 0, 0),
 (3, 1, 3): (0, 2, 0, 0, 0, 0),
 (2, 4, 2): (0, 3, 0, 0, 0, 0),
 (1, 2, 1): (0, 4, 0, 0, 0, 0),
 (3, 4, 4): (0, 0, 1, 0, 0, 0),
 (1, 3, 3): (0, 0, 2, 0, 0, 0),
 (4, 2, 2): (0, 0, 3, 0, 0, 0),
 (2, 1, 1): (0, 0, 4, 0, 0, 0),
 (1, 0, 0): (0, 0, 0, 1, 0, 0),
 (2, 0, 0): (0, 0, 0, 2, 0, 0),
 (3, 0, 0): (0, 0, 0, 3, 0, 0),
 (4, 0, 0): (0, 0, 0, 4, 0, 0),
 (0, 1, 0): (0, 0, 0, 0, 1, 0),
 (0, 2, 0): (0, 0, 0, 0, 2, 0),
 (0, 3, 0): (0, 0, 0, 0, 3, 0),
 (0, 4, 0): (0, 0, 0, 0, 4, 0),
 (0, 0, 1): (0, 0, 0, 0, 0, 1),
 (0, 0, 2): (0, 0, 0, 0, 0, 2),
 (0, 0, 3): (0, 0, 0, 0, 0, 3),
 (0, 0, 4): (0, 0, 0, 0, 0, 4)}
```

Now we can decode using the lookup table

```
def decode(v):
  return v-lookup[tuple(H*v)]

[ (decode(v), decode(v) in C) for v in [ V([0,2,3,4,3,2]),
                                         V([0,1,2,0,4,0])]]
=>
[((1, 2, 3, 4, 3, 2), True), ((0, 1, 2, 0, 4, 3), True)]
```

Bibliography