Explicit Examples of Codes from Curves

Mckenzie West
University of Wisconsin-Eau Claire

with María Chara, Ernest Guico, Beth Malmskog and Bianca Thompson

Math 380: Research Methods, October 4, 2021
What is a code?

Definition

A code is a method of encoding data as a word in a given alphabet. We call the words codewords.
What is a code?

**Definition**
A code is a method of encoding data as a word in a given alphabet. We call the words codewords.

**Example**
Data: *letters in the English alphabet*
Codewords: *binary representations of the integers 0-25*

<table>
<thead>
<tr>
<th>Letter</th>
<th>Codeword</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>00000</td>
</tr>
<tr>
<td>b</td>
<td>00001</td>
</tr>
<tr>
<td>c</td>
<td>00010</td>
</tr>
<tr>
<td>d</td>
<td>00011</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>z</td>
<td>11001</td>
</tr>
</tbody>
</table>
Linear Codes

Definition

A **linear code** is a code who’s codewords form a vector space.
**Definition**

A **linear code** is a code who’s codewords form a vector space.

**Example**

Data: *the integers 0 and 1*

Codewords: \((0,0,0,0,0)\) and \((1,1,1,1,1)\)

\[
\begin{align*}
0 & \leftrightarrow (0,0,0,0,0) \\
1 & \leftrightarrow (1,1,1,1,1)
\end{align*}
\]
Why do we need codes?

We need to store and transmit data
- effectively
- efficiently
- robustly
What is an error correcting code?

**Definition**
An *error correcting code* is a code that has an algorithmic method to detect and correct errors in transmitted or stored data.

<table>
<thead>
<tr>
<th>Data</th>
<th>Codewords</th>
</tr>
</thead>
<tbody>
<tr>
<td>the integers 0 and 1</td>
<td>(0,0,0,0,0) and (1,1,1,1,1)</td>
</tr>
</tbody>
</table>

0 ↔ (0,0,0,0,0)  
1 ↔ (1,1,1,1,1)

If the computer receives (0,0,0,1,0) it's decode it as 0. Detects and corrects ≤ 2 errors.
What is an error correcting code?

**Definition**

An error correcting code is a code that has an algorithmic method to detect and correct errors in transmitted or stored data.

**Example (Repetition Code)**

Data: *the integers 0 and 1*

Codewords: $(0,0,0,0,0)$ and $(1,1,1,1,1)$

\[
\begin{align*}
0 & \leftrightarrow (0,0,0,0,0) \\
1 & \leftrightarrow (1,1,1,1,1)
\end{align*}
\]

If the computer receives $(0,0,0,1,0)$ it’s decode it as 0. Detects and corrects $\leq 2$ errors.
Quantifying Quality

**Definition**

- **Hamming distance** between $u$ and $v$ - $d(u, v)$ - number of places in which $u$ and $v$ differ
Quantifying Quality

Definition

- **Hamming distance** between \(u\) and \(v\) - \(d(u, v)\) - number of places in which \(u\) and \(v\) differ
- **minimum distance** of a code - minimum Hamming distance between any two code words.

Note

If \(C\) is a linear code, minimum distance is equal to minimum weight of the nonzero vectors.

If \(d\) is the minimum distance

\(d - 1\) errors can be detected and \(\lfloor d - 1 \rfloor \) errors can be corrected.
Quantifying Quality

<table>
<thead>
<tr>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Hamming distance</strong> between $u$ and $v$ - $d(u, v)$ - number of places in which $u$ and $v$ differ</td>
</tr>
<tr>
<td><strong>minimum distance</strong> of a code - minimum Hamming distance between any two code words.</td>
</tr>
<tr>
<td><strong>weight</strong> of a codeword - the number of non-zero entries in the code word</td>
</tr>
</tbody>
</table>
Quantifying Quality

Definition

- **Hamming distance** between $u$ and $v$ - $d(u, v)$ - number of places in which $u$ and $v$ differ.
- **minimum distance** of a code - minimum Hamming distance between any two code words.
- **weight** of a codeword - the number of non-zero entries in the code word.

Note

- If $C$ is a linear code, minimum distance is equal to minimum weight of the nonzero vectors.
Definition

- **Hamming distance** between $u$ and $v$ - $d(u, v)$ - number of places in which $u$ and $v$ differ
- **Minimum distance** of a code - minimum Hamming distance between any two code words.
- **Weight** of a codeword - the number of non-zero entries in the code word

Note

- If $C$ is a linear code, minimum distance is equal to minimum weight of the nonzero vectors.
- If $d$ is the minimum distance
  - $d - 1$ errors can be detected and
  - $\left\lfloor \frac{d-1}{2} \right\rfloor$ errors can be corrected.
Defining Quality

Note

d - Minimum distance  k - Dimension of Code

- We want $d$ to be high with respect to $n$.
- We want $k$ to be high with respect to $n$. 

Theorem (Singleton Bound)

For all linear codes, $\dim k + d \leq \dim n + 1$
Defining Quality

**Note**

- $d$ - Minimum distance
- $k$ - Dimension of Code

- We want $d$ to be high with respect to $n$.
- We want $k$ to be high with respect to $n$.

**Theorem (Singleton Bound)**

*For all linear codes,*

$$k + d \leq n + 1$$
Consider the subspace of $\mathbb{F}_2^{24}$ generated by the rows of the matrix:
Example

Consider the subspace of $\mathbb{F}_2^{24}$ generated by the rows of the matrix:

- Minimum Distance $d = 8$
Example

Consider the subspace of $\mathbb{F}_2^{24}$ generated by the rows of the matrix:

- Minimum Distance $d = 8$
- Detectable Errors: 7
Binary Golay Code

Example

Consider the subspace of $\mathbb{F}_2^{24}$ generated by the rows of the matrix:

- Minimum Distance $d = 8$
- Detectable Errors: 7
- Correctable Errors: 3
Binary Golay Code

Example

Consider the subspace of $\mathbb{F}_2^{24}$ generated by the rows of the matrix:

- Minimum Distance $d = 8$
- Detectable Errors: 7
- Correctable Errors: 3
Reed–Solomon Codes

Example

Let $F_q = \{0, 1, \alpha, \ldots, \alpha^{n-1}\}$ be finite field and $k \leq n$.

Data: polynomials of degree $< k$

Codewords: $p \leftrightarrow \text{ev}(p) := (p(1), p(\alpha), \ldots, p(\alpha^{n-1})) \in F_q^n$
Reed–Solomon Codes

Example

Let $\mathbb{F}_q = \{0, 1, \alpha, \ldots, \alpha^{n-1}\}$ be finite field and $k \leq n$.

Data: polynomials of degree $< k$

Codewords: $p \leftrightarrow \text{ev}(p) := (p(1), p(\alpha), \ldots, p(\alpha^{n-1})) \in \mathbb{F}_q^n$

- Minimum Distance $d = n - k + 1$
Reed–Solomon Codes

Example

Let $\mathbb{F}_q = \{0, 1, \alpha, \ldots, \alpha^{n-1}\}$ be finite field and $k \leq n$.

Data: polynomials of degree $< k$

Codewords: $p \leftrightarrow \text{ev}(p) := (p(1), p(\alpha), \ldots, p(\alpha^{n-1})) \in \mathbb{F}_q^n$

- Minimum Distance $d = n - k + 1$
  Note: This is the maximum possible distance for a linear code of size $(n, k)$. 
Example

Let $\mathbb{F}_q = \{0, 1, \alpha, \ldots, \alpha^{n-1}\}$ be finite field and $k \leq n$.

Data: polynomials of degree $< k$

Codewords: $p \leftrightarrow ev(p) := (p(1), p(\alpha), \ldots, p(\alpha^{n-1})) \in \mathbb{F}_q^n$

- Minimum Distance $d = n - k + 1$
  Note: This is the maximum possible distance for a linear code of size $(n, k)$.
- Detectable Errors $n - k$
Reed–Solomon Codes

Example

Let $\mathbb{F}_q = \{0, 1, \alpha, \ldots, \alpha^{n-1}\}$ be finite field and $k \leq n$.

Data: polynomials of degree $< k$

Codewords: $p \leftrightarrow \text{ev}(p) := (p(1), p(\alpha), \ldots, p(\alpha^{n-1})) \in \mathbb{F}_q^n$

- Minimum Distance $d = n - k + 1$
  
  Note: This is the maximum possible distance for a linear code of size $(n, k)$.

- Detectable Errors $n - k$

- Correctable Errors $\frac{n - k}{2}$. 
Another construction for a code - codes from curves

### Example (Evaluation Code)

**Given**

- \(X/F_q\) a curve of genus \(g\)
- \(F\) its function field
- \(\mathcal{P} = \{P_1, \ldots, P_n\}\) points of \(X\)
- \(D := P_1 + \cdots + P_n\) a divisor of \(X\)
- \(G\) a divisor of \(X\) with \(\text{Supp} G \cap \text{Supp} D = \emptyset\)
Another construction for a code - codes from curves

<table>
<thead>
<tr>
<th>Example (Evaluation Code)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Given</strong></td>
</tr>
<tr>
<td>- $X/\mathbb{F}_q$ a curve of genus $g$</td>
</tr>
<tr>
<td>- $F$ its function field</td>
</tr>
<tr>
<td>- $\mathcal{P} = {P_1, \ldots, P_n}$ points of $X$</td>
</tr>
<tr>
<td>- $D := P_1 + \cdots + P_n$ a divisor of $X$</td>
</tr>
<tr>
<td>- $G$ a divisor of $X$ with $\text{Supp } G \cap \text{Supp } D = \emptyset$</td>
</tr>
</tbody>
</table>

**Data:** $\mathcal{L}(G) = \{f \in F \mid \text{div}(f) + G \geq 0\} \cup \{0\}$
Another construction for a code - codes from curves

Example (Evaluation Code)

Given

- $X/F_q$ a curve of genus $g$
- $F$ its function field
- $\mathcal{P} = \{P_1, \ldots, P_n\}$ points of $X$
- $D := P_1 + \cdots + P_n$ a divisor of $X$
- $G$ a divisor of $X$ with $\text{Supp } G \cap \text{Supp } D = \emptyset$

Data: $\mathcal{L}(G) = \{f \in F \mid \text{div}(f) + G \geq 0\} \cup \{0\}$

Codewords: $f \leftrightarrow \text{ev}_\mathcal{P}(f) := (f(P_1), f(P_2), \ldots, f(P_n)) \in F_q^n$
Another construction for a code - codes from curves

Example (Evaluation Code)

Given
- $X/\mathbb{F}_q$ a curve of genus $g$
- $F$ its function field
- $\mathcal{P} = \{P_1, \ldots, P_n\}$ points of $X$
- $D := P_1 + \cdots + P_n$ a divisor of $X$
- $G$ a divisor of $X$ with $\text{Supp } G \cap \text{Supp } D = \emptyset$

Data: $\mathcal{L}(G) = \{f \in F | \text{div}(f) + G \geq 0\} \cup \{0\}$

Codewords: $f \leftrightarrow \text{ev}_\mathcal{P}(f) := (f(P_1), f(P_2), \ldots, f(P_n)) \in \mathbb{F}_q^n$

Dimension $k = \ell(G) - \ell(G - D)$
Another construction for a code - codes from curves

Example (Evaluation Code)

Given
- $X/\mathbb{F}_q$ a curve of genus $g$
- $F$ its function field
- $\mathcal{P} = \{P_1, \ldots, P_n\}$ points of $X$
- $D := P_1 + \cdots + P_n$ a divisor of $X$
- $G$ a divisor of $X$ with $\text{Supp } G \cap \text{Supp } D = \emptyset$

Data: $\mathcal{L}(G) = \{f \in F \mid \text{div}(f) + G \geq 0\} \cup \{0\}$

Codewords: $f \leftrightarrow \text{ev}_\mathcal{P}(f) := (f(P_1), f(P_2), \ldots, f(P_n)) \in \mathbb{F}_q^n$

- Dimension $k = \ell(G) - \ell(G - D)$
- Minimum Distance $d \geq n - \deg G$
Properties

If \( \deg G < n \) then

\[ \text{ev}_P \text{ is injective on } \mathcal{L}(G) \]
Properties

If \( \deg G < n \) then

1. \( \text{ev}_{\mathcal{P}} \) is injective on \( \mathcal{L}(G) \)
2. \( k = \ell(G) \geq \deg(G) + 1 - g \)
Properties

If $\text{deg } G < n$ then

1. $\text{ev}_\mathcal{F}$ is injective on $\mathcal{L}(G)$

2. $k = \ell(G) \geq \text{deg}(G) + 1 - g$

3. Given a basis $\{f_1, \ldots, f_k\}$ of $\mathcal{L}(G)$, we have a generator matrix for the code:

$$
\begin{bmatrix}
  f_1(P_1) & f_1(P_2) & \cdots & f_1(P_n) \\
  f_2(P_1) & f_2(P_2) & \cdots & f_2(P_n) \\
  \vdots & \vdots & \ddots & \vdots \\
  f_k(P_1) & f_k(P_2) & \cdots & f_k(P_n)
\end{bmatrix}
$$
Bounds

Combining a lower and upper bound

If $\deg(G) < n$, then

$$n + 1 \geq k + d \geq n + 1 - g.$$
Combining a lower and upper bound

If \( \deg(G) < n \), then

\[ n + 1 \geq k + d \geq n + 1 - g. \]

Goal

Construct *asymptotically good curves*. 
Asymptotically Good Curves

**Definition**

For an \([n, k, d]\) code

- **information rate** is \(R = k/n\)
- **relative distance** is \(\delta = d/n\)

**Theorem (Gilbert-Varshamov Bound)**

For any fixed \(q\) and \(\delta \leq 1 - 1/q\), and an arbitrarily small \(\epsilon > 0\), there is an infinite family of codes with

\[
R \geq 1 - h_q(\delta) - \epsilon,
\]

where \(h_q(x)\) is the entropy function,

\[
h_q(x) := x \log_q(q - 1) - x \log_q(x) - (1 - x) \log_q(1 - x).
\]
Hermitian Curves with $q = 2$

$$y^2z + y = x^3$$

Example ($q = 2$)

- $\mathbb{F}_4 = \{0, 1, a, a + 1\}$
- $C : y^2z + yz^2 + x^3$
- $F = \text{Frac}(\mathbb{F}_q[x, y, z]/(y^2z + z^2y + x^3))$
Hermitian Curves with $q = 2$

Example ($q = 2$)

- $\mathbb{F}_4 = \{0, 1, a, a + 1\}$
- $C : y^2z + yz^2 + x^3$
- $F = \text{Frac}(\mathbb{F}_q[x, y, z]/(y^2z + z^2y + x^3))$
- $\mathcal{P}$ is the affine points of $C$
Hermitian Curves with $q = 2$

Example ($q = 2$)

- $\mathbb{F}_4 = \{0, 1, a, a + 1\}$
- $C : y^2z + yz^2 + x^3$
- $F = \text{Frac}(\mathbb{F}_q[x, y, z]/(y^2z + z^2y + x^3))$
- $\mathcal{P}$ is the affine points of $C$
- $G = 5(0 : 1 : 0)$
Hermitian Curves with $q = 2$

Example ($q = 2$)

- $\mathbb{F}_4 = \{0, 1, a, a + 1\}$
- $C : y^2z + yz^2 + x^3$
- $F = \text{Frac}(\mathbb{F}_q[x, y, z]/(y^2z + z^2y + x^3))$
- $\mathcal{P}$ is the affine points of $C$
- $G = 5(0 : 1 : 0)$
- $\mathcal{L}(G)$ has basis $\{1, x, y, x^2, xy\}$
Example

Continuing, $\mathbb{F}_4 = \{0, 1, a, a + 1\}$, $C : y^2z + yz^2 + x^3$, $\mathcal{L}(G) = \langle 1, x, y, x^2, xy \rangle$, and

$$\mathcal{P} = \{(0, 0), (0, 1), (a, a), (a, a + 1), (a + 1, a), (a + 1, a + 1), (1, a), (1, a + 1)\}.$$
Hermitian Curve with $q = 2$, Generator Matrix

Example

Continuing, $\mathbb{F}_4 = \{0, 1, a, a + 1\}$, $C : y^2 z + yz^2 + x^3$, $\mathcal{L}(G) = \langle 1, x, y, x^2, xy \rangle$, and

$$\mathcal{P} = \{(0, 0), (0, 1), (a, a), (a, a + 1), (a + 1, a), (a + 1, a + 1), (1, a), (1, a + 1)\}.$$

Generating matrix:

$$
\begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & a & a & a + 1 & a + 1 & 1 & 1 & 1 \\
0 & 1 & a & a + 1 & a & a + 1 & a & a + 1 \\
0 & 0 & a + 1 & a + 1 & a & a & 1 & 1 & 1 \\
0 & 0 & a + 1 & 1 & 1 & a & a & a & a + 1
\end{bmatrix}
$$
Hermitian Curve with $q = 2$, Generator Matrix

Example

Continuing, $\mathbb{F}_4 = \{0, 1, a, a + 1\}$, $C : y^2z + yz^2 + x^3$, $\mathcal{L}(G) = \langle 1, x, y, x^2, xy \rangle$, and

$$\mathcal{P} = \{(0, 0), (0, 1), (a, a), (a, a + 1), (a + 1, a), (a + 1, a + 1), (1, a), (1, a + 1)\}.$$

Generating matrix:

$$
\begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & a & a & a + 1 & a + 1 & 1 & 1 \\
0 & 1 & a & a + 1 & a & a + 1 & a & a + 1 \\
0 & 0 & a + 1 & a + 1 & a & a & 1 & 1 \\
0 & 0 & a + 1 & 1 & 1 & a & a & a + 1 \\
0 & 0 & a + 1 & 1 & 1 & a & a & a + 1
\end{bmatrix}
$$

- Dimension $k = 5$
Hermitian Curve with $q = 2$, Generator Matrix

Example

Continuing, $\mathbb{F}_4 = \{0, 1, a, a + 1\}$, $C : y^2 z + yz^2 + x^3$, $\mathcal{L}(G) = \langle 1, x, y, x^2, xy \rangle$, and

$$\mathcal{P} = \{(0, 0), (0, 1), (a, a), (a, a + 1), (a + 1, a),$$
$$ (a + 1, a + 1), (1, a), (1, a + 1)\}.$$ 

Generating matrix:

$$
\begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & a & a & a + 1 & a + 1 & 1 & 1 \\
0 & 1 & a & a + 1 & a & a + 1 & a & a + 1 \\
0 & 0 & a + 1 & a + 1 & a & a & 1 & 1 \\
0 & 0 & a + 1 & 1 & 1 & a & a & a + 1 \\
\end{bmatrix}
$$

- Dimension $k = 5$
- Minimum Distance $d = 3$
Hermitian Curves Generally

Define $C : y^q z + y z^q = x^{q+1}$ over $\mathbb{F}_{q^2}$.

Take $\mathcal{P}$ again to be all of the affine points of $C$ and $G = rP_{\infty}$.

**Proposition**

For each $r \geq 0$, let $I$ be the set of pairs of integers $(i, j)$ satisfying

- $0 \leq i$,
- $0 \leq j \leq q - 1$,
- $iq + j(q + 1) \leq r$.

Then $\mathcal{L}(rP_{\infty}) = \langle x^iy^j \mid (i, j) \in I \rangle$. 
Hermitian Curves Generally

Define $C : y^q z + yz^q = x^{q+1}$ over $\mathbb{F}_{q^2}$.
Take $\mathcal{P}$ again to be all of the affine points of $C$ and $G = rP_\infty$.

Proposition

For each $r \geq 0$, let $I$ be the set of pairs of integers $(i, j)$ satisfying
- $0 \leq i$,
- $0 \leq j \leq q - 1$,
- $iq + j(q + 1) \leq r$.

Then $\mathcal{L}(rP_\infty) = \langle x^i y^j \mid (i, j) \in I \rangle$.

Corollary

The generating matrix for the code corresponding to $rP_\infty$ and $\mathcal{P}$ is
$$(\alpha^i \beta^j)$$ where the rows correspond to fixed values of $(i, j)$ and the
columns correspond to fixed values of $(\alpha, \beta)$ such that
$\beta^q + \beta = \alpha^{q+1}$. 
Bounds

Proposition

Let $N(r)$ be the number of pairs $(i, j)$ for a given $r$. Then for the code described on the previous slide,

- $n = q^3$
- $k = \begin{cases} 
N(r) & 0 \leq r < q^3 \\
(n - N(r)) & q^3 \leq r \leq q^3 + q^2 - q - 2 
\end{cases}$
- $d \geq n - r$
Locally Recoverable Code (LRC)

Definition

A code is \textit{locally recoverable} if its codewords can be reconstructed using only a portion of their values.
Locally Recoverable Code (LRC)

**Definition**

- A code is *locally recoverable* if its codewords can be reconstructed using only a portion of their values.
- We say that $C$ has *locality* $r$ if each codeword can be recovered by accessing at most $r$ other symbols.
Locally Recoverable Code (LRC)

**Definition**

- A code is **locally recoverable** if its codewords can be reconstructed using only a portion of their values.
- We say that \( C \) has **locality** \( r \) if each codeword can be recovered by accessing at most \( r \) other symbols.
- A LRC code for which each symbol has \( t \) recovery sets for each symbol is said to have **availability** \( t \).
Locally Recoverable Code (LRC)

**Definition**
- A code is **locally recoverable** if its codewords can be reconstructed using only a portion of their values.
- We say that $C$ has **locality** $r$ if each codeword can be recovered by accessing at most $r$ other symbols.
- A LRC code for which each symbol has $t$ recovery sets for each symbol is said to have **availability** $t$.

**Example (Let’s draw a picture)**
Fiber Products!

General Idea - (Barg–Tamo–Vlăduț, Haymaker–Malmskog–Matthews)

\[ \mathcal{X} = \mathcal{Y}_1 \times_\mathcal{Y} \mathcal{Y}_2 \times_\mathcal{Y} \cdots \times_\mathcal{Y} \mathcal{Y}_t \]

Data: Rational Functions over \( \mathcal{X} \) with poles above \( \mathcal{G} \).

Encoding: Evaluation at \( \text{Supp}(\mathcal{D}) \)
Fiber Products!

General Idea - (Barg–Tamo–Vlăduţ, Haymaker–Malmskog–Matthews)

\[ \chi = Y_1 \times_y Y_2 \times_y \cdots \times_y Y_t \]

Take \( G \) a divisor on \( Y \) and \( D \) a divisor on \( \chi \).

\( \bullet \) Data: Rational Functions over \( \chi \) with poles above \( G \).

Encoding: Evaluation at \( \text{Supp}(D) \)
Example

Let \( p = 3 \) and \( q = 3^4 \).
For \( a \in A = \{ x \in \mathbb{F}_q \mid a^9 + a = 0 \} \) define

\[
\mathcal{Y}_a : y^3 - y = ax^{3^2+1}
\]
Example

Let $p = 3$ and $q = 3^4$.

For $a \in A = \{ x \in \mathbb{F}_q \mid a^9 + a = 0 \}$ define

$$\mathcal{Y}_a : y^3 - y = ax^{3^2+1}$$

Take two $a_1, a_2 \in A$ that generate $A$ over $\mathbb{F}_3$.

We will use $\mathcal{Y}_{a_1} = \mathcal{Y}_1$ and $\mathcal{Y}_{a_2} = \mathcal{Y}_2$. With $h_1$ and $h_2$ being the projection onto the $x$-coordinate.
Example

Now

- \( \deg(g_i) = 3 \) are ramified only above the point at infinity,
- \( \mathcal{X} \) has genus 36,
- \( \#X(\mathbb{F}_q) = 729 + 1 \)
Example

Now
- \( \text{deg}(g_i) = 3 \) are ramified only above the point at infinity,
- \( \mathcal{X} \) has genus 36,
- \( \# X(\mathbb{F}_q) = 729 + 1 \)

We choose

\[ \mathcal{P} = \{(x, y_1, y_2) \mid y_i^3 + y = a_i x^{3^2+1} \forall i\}. \]
Example

Now
- \( \deg(g_i) = 3 \) are ramified only above the point at infinity,
- \( X \) has genus 36,
- \( \#X(\mathbb{F}_q) = 729 + 1 \)

We choose

\[
\mathcal{P} = \{(x, y_1, y_2) \mid y_i^3 + y = a_i x^{3^2+1} \forall i \}.
\]

Let \( G = rP_{\infty x} \), then the data for our code is

\[
\mathfrak{L}(G) \quad V = \text{Span}\{x^j y_1^{e_1} y_2^{e_2} \mid 0 \leq j \leq r, 0 \leq e_i \leq 1 \}.
\]
The code constructed from the fiber product $X$ is a locally recoverable $[n, k, d]$ code over $\mathbb{F}_q$ with availability $t$ and locality $p - 1$ where

- $n = p^t q$
- $k = (r + 1)(p - 1)^t$
- $d \geq n - p^t r - t(p - 2)p^{t-1}(\sqrt{q} + 1)$. 
The code constructed from the fiber product $X$ is a locally recoverable $[n, k, d]$ code over $\mathbb{F}_q$ with availability $t$ and locality $p - 1$ where

- $n = p^t q$
- $k = (r + 1)(p - 1)^t$ and
- $d \geq n - p^t r - t(p - 2)p^{t-1}(\sqrt{q} + 1)$.

Theorem (Kottler)

If $r \leq q - t\sqrt{q} - t - 1$ then the code obtained by evaluating functions in $V$ at points in $\mathcal{P}$ has minimum distance

$$d = n - p^t r - t(p - 2)p^{t-1}(\sqrt{q} + 1)$$
General Minimum Distance Theorem

Let $\mathcal{X}$ be a fiber product of curves of the form $f(y_i) = g(x)$ over $\mathbb{F}_q$. Let $\mathcal{P} \subseteq \mathcal{X}(\mathbb{F}_q)$ and choose $F_0 = F_x, F_1, \ldots, F_t$ with $F_i \subseteq y_i(P) \subseteq \mathbb{F}_q$, so that

- $\#F_x = r$
- $\#F_i = \text{deg}(h_i)$ for all $i = 1, \ldots, t$
General Minimum Distance Theorem

Let $X$ be a fiber product of curves of the form $f(y_i) = g(x)$ over $\mathbb{F}_q$. Let $\mathcal{P} \subseteq X(\mathbb{F}_q)$ and choose $F_0 = F_x, F_1, \ldots, F_t$ with $F_i \subseteq y_i(P) \subseteq \mathbb{F}_q$, so that

- $\#F_x = r$
- $\#F_i = \deg(h_i)$ for all $i = 1, \ldots, t$
- For all $j \neq k$ with $0 \leq j, k \leq t$ there is no $P \in \mathcal{P}$ with $y_j$ coordinate in $F_j$ and $y_k$-coordinate in $F_k$. 

Theorem (Kottler, Chara–Guico–Malmskog–Thompson–W.)

Given the above set up, the code obtained by evaluation functions in $V$ at points in $B$ has minimum distance $d = n - 1 - \sum_{i=1}^{t} (\deg(h_i) - 2) \deg(y_i),$ where $n = \#P$ is the length of the code.
General Minimum Distance Theorem

Let $X$ be a fiber product of curves of the form $f(y_i) = g(x)$ over $\mathbb{F}_q$. Let $\mathcal{P} \subseteq X(\mathbb{F}_q)$ and choose $F_0 = F_x, F_1, \ldots, F_t$ with $F_i \subseteq y_i(P) \subseteq \mathbb{F}_q$, so that

- $\#F_x = r$
- $\#F_i = \deg(h_i)$ for all $i = 1, \ldots, t$
- For all $j \neq k$ with $0 \leq j, k \leq t$ there is no $P \in \mathcal{P}$ with $y_j$ coordinate in $F_j$ and $y_k$-coordinate in $F_k$.
- For all $i$ with $1 \leq i \leq t$, the projection $y_i : X \to \mathbb{F}_q$ is not ramified over any points in $F_i$. 

Theorem (Kottler, Chara–Guico–Malmskog–Thompson–W.)
Given the above set up, the code obtained by evaluation functions in $V$ at points in $B$ has minimum distance $d = n - 1 - t \sum_{i=1}^{t} (\deg(h_i) - 2) \deg(y_i)$, where $n = \#P$ is the length of the code.
General Minimum Distance Theorem

Let \( \mathcal{X} \) be a fiber product of curves of the form \( f(y_i) = g(x) \) over \( \mathbb{F}_q \). Let \( \mathcal{P} \subseteq \mathcal{X}(\mathbb{F}_q) \) and choose \( F_0 = F_x, F_1, \ldots, F_t \) with \( F_i \subseteq y_i(P) \subseteq \mathbb{F}_q \), so that

- \( \#F_x = r \)
- \( \#F_i = \deg(h_i) \) for all \( i = 1, \ldots, t \)
- For all \( j \neq k \) with \( 0 \leq j, k \leq t \) there is no \( P \in \mathcal{P} \) with \( y_j \) coordinate in \( F_j \) and \( y_k \)-coordinate in \( F_k \).
- For all \( i \) with \( 1 \leq i \leq t \), the projection \( y_i: \mathcal{X} \to \mathbb{F}_q \) is not ramified over any points in \( F_i \).

Theorem (Kottler, Chara–Guico–Malmskog–Thompson–W.)

Given the above set up, the code obtained by evaluation functions in \( V \) at points in \( B \) has minimum distance

\[
  d = n - 1 \deg(x) - \sum_{i=1}^{t} (\deg(h_i) - 2) \deg(y_i),
\]

where \( n = \#\mathcal{P} \) is the length of the code.