3. Linear Block Codes

3.1 Limitations

Problem: *As presented, block codes have no "helpful" structure.*

- How can one **design** a code for a given d_{min}, R, n?
- How can one find the **best** such code?
- To **encode** requires online storage of all the code words.
- To **decode** requires exponentially complex table lookup.

Challenge

- Encode information $\mathbf{i} = (i_0, i_1, \ldots, i_{k-1})$ into code word $\mathbf{c} = (c_0, c_1, \ldots, c_{n-1})$

$$\mathbf{c} = f(\mathbf{i}).$$

- Estimate transmitted information from received vector $\mathbf{y} = (y_0, \ldots, y_{n-1})$:

$$D : \mathbf{y} \rightarrow \hat{\mathbf{i}}$$

both subject to constraints that

- $f(\cdot)$ be a *linear* transformation and
- D be an *efficient* algorithm.

But

- The canonical form of a linear transformation is:

$$\mathbf{c} = \mathbf{iG}$$

  where $\mathbf{G}$ is a $k \times n$ matrix, and

- all the codewords $\{\mathbf{c}\}$ are distinct when the rank of $\mathbf{G}$ is k.

So, if

$$\mathbf{y} = \mathbf{c} + \mathbf{e}$$

there is hope of extracting $\mathbf{i}$ with an algorithm of moderate complexity.

3.2 Basic Definitions

Definition 1 *A* **linear block code** *is a $k-$dimensional vector subspace of the $n-$tuples over a field.* □

For now,

Definition 2 *A* **field** *is a set of elements in which one can do "ordinary arithmetic" without leaving the set. In a finite field, the set is of finite order.*

□

$$
\begin{aligned}
n &= \text{block length} \\
k &= \text{dimension} \\
M &= q^k \\
GF(q) &= \text{symbol field (more later)}
\end{aligned}
$$

Terminology: "(n, k) block code."

Lemma: *The* **code rate** *of an LBC is*

$$R = \frac{k}{n},$$

bits/symbol or bits/use_of_the_channel.

Proof: Follows from the definition for a block code. $\square$

3.3 Basic Properties of LBCs

Lemma

The linear combination of any subset of codewords is a codeword.

Proof: Follows from subspace definition. $\square$

Note: Many of the basic properties of an LBC, including manipulation of its generator matrix, directly follow from its nature as a vector subspace, and surely have been well covered in Linear Algebra.

Definition 3 *The* **minimum weight** *of a linear block code is:*

$$w_{min}(\mathcal{C}) = \min_{\mathbf{c} \in \mathcal{C}} w_H(\mathbf{c}).$$

□

Theorem 1 *For a linear block code (LBC),* $d_{min} = w_{min}$.

Proof:

$$\begin{aligned} d_{min} &= \min_{\mathbf{c}_i, \mathbf{c}_j \in \mathcal{C}} d(\mathbf{c}_i, \mathbf{c}_j) \\ &= \min w_H(\mathbf{c}_i - \mathbf{c}_j) \\ &= \min w_H(\mathbf{c}_k) \text{ for some } k (\text{by linearity}) \end{aligned}$$

□

Corollary: *An LBC can detect any error pattern for which*

$$w_H(e) \leq d_{min} - 1.$$

□

Lemma:

The ***undetectable*** *error patterns for an LBC are*

- independent of the codeword transmitted;
- the set of non-zero **codewords**;
- the set of words within $\lfloor (d_{min} - 1)/2 \rfloor$ of any other codeword.

Proof:

$$\mathbf{y} = \mathbf{c} + \mathbf{e}$$

- When $\mathbf{e} \in \mathcal{C}$, no error is detected.
- When
$$d_H(\mathbf{y}, \mathbf{c}') \leq \left\lfloor \frac{d_{min} - 1}{2} \right\rfloor,$$
for some $\mathbf{c}' \neq \mathbf{c}$, $\mathbf{c}' \in \mathcal{C}$, decoder will output $\mathbf{c}'$, committing an undetectable error. $\square$

3.4 Matrix Description of the LBC

3.4.1 Generator Matrix (G)

Write basis vectors $(\mathbf{g}_1, \mathbf{g}_2, \ldots \mathbf{g}_k)$ of $\mathcal{C}$ as rows of matrix $\mathbf{G}$ $(k \times n)$:

$$\mathbf{G} = \begin{bmatrix} --- & \mathbf{g}_1 & --- \\ --- & \mathbf{g}_2 & --- \\ & \vdots & \\ --- & \mathbf{g}_k & --- \end{bmatrix}.$$

- information: $\mathbf{a} = (a_1, \ldots, a_k)$;
- encoded uniquely as:

$$\mathbf{c} = \mathbf{a} \cdot \mathbf{G} = (a_1, \ldots, a_k) \cdot \mathbf{G},\ a_i \in GF(q).$$

3.4.2 Dual Code and Parity Check Matrix

Definition 4 *The dual code $\mathcal{C}^{\perp}$ of $\mathcal{C}$ is the orthogonal complement of $\mathcal{C}$.* □

Let $(\mathbf{h}_1, \ldots \mathbf{h}_{n-k})$ be a basis for $\mathcal{C}^{\perp}$. Then,

$$\mathbf{c} \in \mathcal{C} \Rightarrow \mathbf{c}\mathbf{H}^T = 0,$$

where the rows of $\mathbf{H}$ are $(\mathbf{h}_1, \ldots \mathbf{h}_{n-k})$.

Thus, we have an *error detection* algorithm:

- Transmit $\mathbf{c}$, receive $\mathbf{y} = \mathbf{c} + \mathbf{e}$.

$$\begin{aligned} \mathbf{y}\mathbf{H}^T &= \mathbf{c}\mathbf{H}^T + \mathbf{e}\mathbf{H}^T \\ &= 0 + \mathbf{e}\mathbf{H}^T. \end{aligned}$$

- $\mathbf{v}\mathbf{H}^T \neq 0 \Rightarrow \mathbf{e} \neq \mathbf{0}$ and the presence of errors is easily detected.

Theorem 2 *$\mathcal{C}$ contains a nonzero word of weight $w \Leftrightarrow$ a set of w columns of $\mathbf{H}$ is linearly dependent.*

Proof:

- ($\Rightarrow$): If $\mathbf{c} \in \mathcal{C}$, then $\mathbf{cH}^T = 0$. Hence, if $w_H[\mathbf{w}] = w$ then a set of w columns of $\mathbf{H}$ is linearly dependent.
- ($\Leftarrow$): If w columns of $\mathbf{H}$ are linearly dependent, there exists a linear combination of w columns which $= 0$; *i.e.*, $\mathbf{vH}^T = 0$ and $w_H[\mathbf{v}]$ must be w.

$\square$

3.4.3 To find the Parity Check Matrix

Corollary: *The fewest number of columns* $\mathbf{H}$ *that are linearly dependent is* d_{min}. □

To find a code having a required d_{min}:

- find a matrix of d_{min} linearly dependent columns such that no set of $d_{min} - 1$ columns is linearly dependent;
- use this matrix as the check matrix $\mathbf{H}$.

3.4.4 Equivalent Codes

Definition 5 *The following are* **elementary row operations** *on the generator of a vector subspace:*

- *interchange any pair of rows;*
- *multiply a row by a non-zero field element;*
- *add a multiple of one row to another;*
- *an inverse of any of these three operations* □

Theorem 3 *Performing* **elementary row operations** *on the generator* $\mathbf{G}$ *of a code produces another matrix* $\mathbf{G}'$ *with the same* **row space** *(up to an isomorphism).*

Proof: Any linear algebra book. □

Definition 6 *The* **leading term** *of a row of a matrix is the first nonzero term.*

□

Definition 7 *A matrix is said to be in* **standard form** *(***row echelon form***) if*

- *every* **leading term** *of a nonzero row is* 1*;*
- *every column containing a leading term is zero elsewhere;*
- *the leading term of any row is to the right of the leading term in every preceding row;*
- *all zero rows (if any) are below all nonzero rows.*

□

Matrix in Standard Form

$$\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & p_1 & p_2 & \cdots & p_n \\ 0 & 1 & 0 & \cdots & 0 & q_1 & q_2 & \cdots & q_n \\ & & & \vdots & & & & & \\ 0 & 0 & 0 & \cdots & 1 & w_1 & w_2 & \cdots & w_n \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \end{bmatrix}$$

Lemma *Any matrix can be placed in standard form by use of the elementary row operations.*

Proof: Obvious. □

Notes:

- Placing a matrix in standard form can reveal its dimension.
- If $\mathbf{G}$ is in standard form and of dimension k
  - the first k positions of the $n-$tuple $\mathbf{a} \cdot \mathbf{G}$ are exactly the contents of $\mathbf{a}$.

If $\mathbf{G}$ is in standard form and of dimension k, we can write:

$$\mathbf{G}_{sf} = [\mathbf{I}_k | \mathbf{P}]$$

Definition 8 *The code generated by $\mathbf{G}_{sf}$ is a* **systematic** *code.*

□

Column Permutations:

If we transpose the i^{th} and j^{th} symbols in every word of $\mathcal{C}$,

- d_{min} is unchanged;
- (n, k) are unchanged.
- The weight of no codeword is changed.
- The resulting code $\mathcal{C}_{eq}$ is said to be **equivalent** to $\mathcal{C}$.
- $\mathbf{G}_{eq}$ is obtained by interchanging the i^{th} and j^{th} columns of the original $\mathbf{G}$.

Lemma: If $\mathbf{G} = [\mathbf{I}_k|\mathbf{P}]$ then $\mathbf{H} = [-\mathbf{P}^T|\mathbf{I}_{n-k}]$.

Proof: It is easy to show that $\mathbf{G}\mathbf{H}^T = 0$. □

Theorem 4 *Every LBC is equivalent to some systematic code.*

Proof: Proof is by elementary row operations and/or column permutations. □

3.4.5 Additional Bounds for LBCs

Theorem 5 (The Singleton Bound): *For any* (n, k) *LBC,* $d_{min} \leq 1 + (n - k)$.

Proof: Write

$$\mathbf{G} = [\mathbf{I}_k|\mathbf{P}]$$

- $\mathbf{I}_k$ contributes 1 to w_{min}.
- $\mathbf{P}$ contributes at most $n - k$ to w_{min}.

□

Definition 9 *A* **maximum distance separable** *or MDS code is one which meets the Singleton Bound with equality.*

$\square$

Hamming Bound for a LBC:

$$\begin{aligned} r &= n-k \\ n-k &\geq \log_q V_q(n,t) \end{aligned}$$

Gilbert Bound for a LBC:

$$n-k \leq \log_q V_q(n,2t)$$

Perfect LBCs

$$n - k = \log_q \sum_{j=0}^{t} \binom{n}{j} (q-1)^j$$

For binary codes, this becomes

$$2^{n-k} = \sum_{j=0}^{t} \binom{n}{j}$$

3.5 The Standard Array and Decoding an LBC

An LBC is a vector subspace. Encoding and decoding will be simplified, compared with the general block code, by use of tools from linear algebra. Therefore, we must introduce elementary group theory before proceeding.

3.5.1 Groups and Cosets

Definition 10 *A* **group** $\mathcal{G}$ *is a set with a binary operation* $\star$ *which together satisfy:*

- **closure:** $a, b \in \mathcal{G} \Rightarrow c = a \star b \in \mathcal{G}$.
- **associativity:** *In* $\mathcal{G}$, $(a \star b) \star c = a \star (b \star c)$.
- **identity:** $\mathcal{G}$ *contains an element* i *such that* $a = a \star i$.
- **inverses:** *For every* $a \in \mathcal{G}$, *there exists* $a^{-1} \in \mathcal{G}$ *such that* $a \star a^{-1} = i$.

□

Definition 11 : *If* $a \cdot b = b \cdot a$*, we say that the group operation is* **commutative** *and that* $\mathcal{G}$ *is a commutative or* **Abelian** *group.*

$\square$

Examples of Groups:

1. the integers $\mathcal{Z}$ under addition;
2. the integers under addition modulo p (prime) (Proof: exercise);
3. the permutations on n symbols under *composition*; for $n = 3$ are a *non-Abelian* group.
   - $g_0 : [(123) \rightarrow (123)] \leftarrow \ identity$
   - $g_1 : [(123) \rightarrow (231)]$
   - $g_2 : [(123) \rightarrow (312)]$
   - $g_3 : [(123) \rightarrow (213)]$
   - $g_4 : [(123) \rightarrow (132)]$
   - $g_5 : [(123) \rightarrow (321)]$

Note: The integers $\mathcal{Z}$ under multiplication do *not* form a group:

- closure: $a, b \in \mathcal{Z} \Rightarrow ab = c \in \mathcal{Z}$.
- associativity: $(ab)c = a(bc)$
- identity: $1 \cdot a = a$
- inverses: The inverse of 3 under multiplication does not exist!

Example: The integers $\mathcal{Z}_p$ under addition $\bmod p$

$+$	0	1	2
0	0	0	0
1	0	1	2
2	0	2	1

3.5.1.1 The Subgroup

Let $\mathcal{G}$ be a group with operation "$\star$" and $\mathcal{H} \subset \mathcal{G}$.

Definition 12 : *$\mathcal{H}$ is a subgroup of $\mathcal{G}$ if it is a group under the operation "$\star$."* □

Lemma: *$\mathcal{H} \subset \mathcal{G}$ is a subgroup of $\mathcal{G}$ if*

- $\mathcal{H}$ is *closed* under "$\star$."
- $\mathcal{H}$ contains the *inverse* of every element of $\mathcal{H}$.

Proof: Exercise □

Examples of subgroups:

- $\mathcal{H}_1 =$ {Even integers} is a subgroup of $\mathcal{Z}$ under addition.
- $\mathcal{H}_2 = \{z \in \mathcal{Z}\ s.t. |z| = 3k,\ k = 0, 1, \ldots\}$ is a subgroup of $\mathcal{Z}$ under addition.
- **Note:** There is no multiplication in $\mathcal{H}_2$. $3k$ is "shorthand" for $k + k + k$.

Definition 13 *:* $h^j \equiv \underbrace{h \star h \star h \cdots h}_{j \text{ times}}$ *where* $\cdot$ *is the group operation.*

$\square$

Lemma: *If* $h \in \mathcal{G}$*, a finite group, then* $\mathcal{H}_3 = \{h, h^2, h^3, \ldots\}$ *is a subgroup of* $\mathcal{G}$*.*

Proof:

$$\begin{aligned} \mathcal{G} \text{ finite} &\Rightarrow \mathcal{H}_3 \text{ finite} \\ \mathcal{H}_3 \text{ finite} &\Rightarrow \text{ series } h^j \text{ repeats} \end{aligned}$$

Therefore, $h^m = h$ for some m. $\square$

3.5.1.2 Coset Decomposition of $\mathcal{G}$

Let $\mathcal{H} = \{e, h_2, \ldots, h_n\}$ be a subgroup of finite group $\mathcal{G}$:

$$\begin{array}{c|cccc}
e & h_2 & h_3 & \cdots & h_n \\
\hline
g_2 \star e & g_2 \star h_2 & g_2 \star h_3 & \cdots & g_2 \star h_n \\
g_3 \star e & g_3 \star h_2 & g_3 \star h_3 & \cdots & g_3 \star h_n \\
\vdots & & & & \\
g_m \star e & g_m \star h_2 & g_m \star h_3 & \cdots & g_m \star h_n
\end{array}$$

- *standard array* or *coset decomposition* of $\mathcal{G}$ (w.r.t. H).
- Each *row* is called a *(left) coset* (of $\mathcal{G}$ in $\mathcal{H}$).
- In the i^{th} row, element g_i is the *coset leader*.
- g_i does not appear in any previous row (by construction).

Theorem 6 *Each $g_i \in \mathcal{G}$ appears exactly once in the standard array.*

Proof:

1. Each appears at least once by construction.
2. If 2 entries in same coset are equal:

$$\begin{aligned} g_i h_j &= g_i h_k \\ (g_i^{-1}) g_i h_j &= (g_i^{-1}) g_i h_k \\ h_i &= h_j \Rightarrow \text{Contradiction} \end{aligned}$$

3. If 2 entries in different cosets are equal:

$$\begin{aligned} g_i h_j &= g_k h_m, \; i < k \\ g_i h_j (h_m^{-1}) &= g_k \end{aligned}$$

But this puts g_k in the i^{th} coset which contradicts construction that coset leaders are not previously used. □

Corollary: *The order of $\mathcal{H}$ divides the order of G.*

Proof: $\mathrm{o}rd(H) =$ the number of columns of standard array. $\square$

Definition 14 *The* **order** *of $g \in \mathcal{G}$ is the smallest integer m s.t. $g^m = e$.*

Corollary: *The order of a group is divisible by the order of any of its elements.*

Proof:

- The set $\{g, g^2, \ldots, g^{ord(g)}\}$ is a (cyclic) subgroup. (Exercise: prove it is a subgroup.)
- Form standard array with respect to that cyclic subgroup. $\square$

This ends the intro to group theory.

3.5.2 Coset Decomposition of the $n-$tuples

- Consider space of $n-$tuples over $GF(q)$.
- Code $\mathcal{C}$ is a subspace (subgroup).
- Construct the standard array with respect to $\mathcal{C}$.
  - First coset: $\mathcal{C}$. Coset leader $= \mathbf{0}$
  - Next coset leader: Any unused $n-$tuple of *lowest weight*.
  - Repeat until space of $n-$tuples is exhausted.

Coset Decomposition of the n-tuples

$$
\begin{array}{lllll}
0 & \mathbf{c}_2 & \mathbf{c}_3 & \cdots & \mathbf{c}_{q^k} \\
\hline
0+\mathbf{v}_1 & \mathbf{c}_2+\mathbf{v}_1 & \mathbf{c}_3+\mathbf{v}_1 & \cdots & \mathbf{c}_{q^k}+\mathbf{v}_1 \\
\vdots & & & & \\
0+\mathbf{v}_t & \mathbf{c}_2+\mathbf{v}_t & \mathbf{c}_3+\mathbf{v}_t & \cdots & \mathbf{c}_{q^k}+\mathbf{v}_t \\
0+\mathbf{v}_{t+1} & \mathbf{c}_2+\mathbf{v}_{t+1} & \mathbf{c}_3+\mathbf{v}_{t+1} & \cdots & \mathbf{c}_{q^k}+\mathbf{v}_{t+1} \\
\vdots & & & & \\
0+\mathbf{v}_l & \mathbf{c}_2+\mathbf{v}_l & \mathbf{c}_3+\mathbf{v}_l & \cdots & \mathbf{c}_{q^k}+\mathbf{v}_l
\end{array}
$$

Lemma: *Let* $t = \lfloor(d_{min} - 1)/2.\rfloor$ *No more than one vector of weight* t *or less can exist in any coset.*

Proof: Exercise. $\square$

- Every correctable error pattern is a coset leader.
- To decode:
  - Find the received word in the standard array.
  - Codeword at top of its column is the most likely transmitted.
  - Corrects all guaranteed error patterns, perhaps others.
- Computational work *still* grows rapidly with n.

3.5.3 Syndrome Decoding

The standard array motivates a simpler but equivalent decoder.

Definition 15 *For any received vector* $\mathbf{v}$*, the* **syndrome** *of* $\mathbf{v}$ *is*

□

$$\mathbf{s} = \mathbf{v}\mathbf{H}^T$$

Theorem 7 *All vectors in the same coset have the same syndrome. That syndrome is unique to the coset.*

Proof: Let $\mathbf{u}$ and $\mathbf{v}$ belong to the coset having leader $\mathbf{x}$. Then

$$\begin{aligned} \mathbf{u} &= \mathbf{x} + \mathbf{c}_j \\ \mathbf{v} &= \mathbf{x} + \mathbf{c}_k \\ \mathbf{s} &= \mathbf{u}\mathbf{H}^T = \mathbf{x}\mathbf{H}^T \\ \mathbf{s}' &= \mathbf{v}\mathbf{H}^T = \mathbf{x}\mathbf{H}^T \end{aligned}$$

$\square$

Syndrome Decoding Algorithm:

- compute the syndrome of the received vector;
- find the corresponding coset leader;
- subtract coset leader from received word.
- If there are $\left\lfloor \frac{d_{min}-1}{2} \right\rfloor$ or fewer errors decoding will be correct.

This decoder is equivalent to the standard array decoder but requires less storage.

Notes:

- Code *guarantees* to correct only t errors per codeword.
- Standard array or syndrome decoding can correct 2^{n-k} error patterns.
- Usually,

$$\sum_{j=0}^{t} \binom{n}{j} < 2^{n-k}.$$

- Equality holds only for *a perfect code.*

3.5.4 Examples

3.5.4.1 Hamming Codes – Binary

Problem: Design an LBC with $d_{min} \geq 3$ for some block length $n = 2^m - 1$.

- If $d_{min} = 3$, then every pair of columns of $\mathbf{H}$ is independent.
- *i.e.*, for binary code, this requires only that
  - no two columns are equal;
  - all columns are nonzero.

- But there are $2^m - 1$ distinct, nonzero, binary $m-$tuples.
- Therefore, we can construct $m-$dimensional $\mathbf{H}$. (why?)
- Therefore, $\mathcal{C}$ has dimension $k = 2^m - 1 - m$ (why?). LBC.

3.5.5 Perfect Codes

Definition 16 *The* **packing radius** *is the radius of the largest sphere that can be drawn around every codeword in $n-$space such that no two spheres intersect.*

□

The value of this radius is $\lfloor (d_{min} - 1)/2 \rfloor$.

Definition 17 *The* **covering radius** *of a code is the radius of the smallest sphere that can be drawn about every codeword such that every point in $n-$space is included.*

□

Definition 18 *A* **perfect code** *is one whose packing and covering radii are equal.* □

(Notice the equivalence to the earlier definition.)

Note: A perfect code satisfies the *Hamming bound* with equality. (See Problem 1.5.)

Recall Examples:

- the Hamming codes;
- the binary $(23, 12)$ Golay code and the ternary $(11, 6)$ Golay codes.

Definition 19 *A* **quasi-perfect code** *is one for which the covering radius equals the packing radius plus one.*

□

3.5.6 New Codes from Existing Codes

Why?

1. as alternative to designing new code, to wit:
   - May already know the properties of some code.
     - The properties of the new code would be easy to infer.
   - Decoder for the modified code often can be used with little or no modification.
2. when existing code doesn't quite fit an application:
   - block code words representing data of certain size;
   - to fit a codeword into allocated fields in network protocol.

How?

Definition 20 *Adding a check symbol* **expands** *a code.*

□

Definition 21 *Adding an info symbol* **lengthens** *a code.*

□

Definition 22 *Dropping a check symbol* **punctures** *a code.*

□

Definition 23 *Dropping an info symbol* **shortens** *a code.*

□

Definition 24 *Increasing k but not n* **augments** *a code.*

□

Definition 25 *Decreasing k but not n* **expurgates** *a code.*

□

Example: Expansion

- Consider a binary (n, k) code with odd minimum distance d_{min}.

- Add one additional position which checks (even) parity on all n positions.

  - The dimension k of the code is unchanged.

  - d_{min} increases by one. (Why?)

  - The code length n increases by one.

The transpose of the parity check matrix of the expanded code has the following form:

$$\mathbf{H}^T = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & & & \\ \cdot & & & \\ \cdot & & & \\ \cdot & H & & \\ 0 & & & \end{bmatrix}$$

As an example of an expanded code, consider an expanded binary $(2^m, 2^m - m)$ Hamming code with $d_{min} = 4$.

End of introduction to linear block codes.

APPENDIX: Review of Vector Spaces

Definition 26 *A set $\mathcal{V}$ is said to be a* **vector space** *over the field F if:*

- *$\mathcal{V}$ is an Abelian group under vector addition.*
- *$\mathcal{V}$ is closed under multiplication by scalar; i.e.,*

$$c \in F,\ \mathbf{v} \in \mathcal{V} \Rightarrow c\mathbf{v} \in \mathcal{V}.$$

$\square$

Properties of $\mathcal{V}$:

- *identity*: $1_F \mathbf{v} = \mathbf{v},\ \forall \mathbf{v} \in \mathcal{V}$.
- *distributive law*: For $c_1, c_2, c \in F$ and $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v} \in \mathcal{V}$,

$$(c_1 + c_2)\mathbf{v} = c_1\mathbf{v} + c_2\mathbf{v}$$

$$c(\mathbf{v}_1 + \mathbf{v}_2) = c\mathbf{v}_1 + c\mathbf{v}_2.$$

- *associative law* $(c_1c_2)\mathbf{v} = c_1(c_2\mathbf{v})$.

Warnings:

- 0_V and 0_F are distinct.
- $+$ in $\mathcal{V}$ is distinct from $+$ in F.

We distinguish from the context.

Examples:

- $n-$tuples over a field:

$$\mathbf{v} = (v_1, v_2, \ldots, v_n), v_i \in F.$$

- L_2 real-valued functions:

$$\int_{-\infty}^{\infty} |f(x)|^2 dx < \infty.$$

- Polynomials in x, coefficients in $GF(q)$, vector addition is the addition of polynomials:

$$\begin{aligned} \mathbf{v} &= (a_0 + a_1 x + a_2 x^2 + \cdots),\ a_i \in GF(q) \\ c\mathbf{v} &= (ca_0 + ca_1 x + ca_2 x^2 + \cdots),\ ca_i \in GF(q). \end{aligned}$$

Exercise: Verify each.

Definitions (Linear Algebra):

- $u = a_1\mathbf{v}_1 + a_2\mathbf{v}_2 + a_3\mathbf{v}_3$ is a *linear combination* of $\mathbf{v}_1,\ \mathbf{v}_2,\ \mathbf{v}_3$.
- $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\}$ are said to be *linearly dependent* if there exist $\{a_i\}_{i=1}^n$, not all zero, such that

$$\sum_{i=1}^{n} a_i\mathbf{v}_i = 0.$$

- A set of vectors that is not linearly independent is said to be *linearly dependent.*
- A set $\{\mathbf{v}_1, \ldots, \mathbf{v}_N\}$ is said to *span* $\mathcal{V}$ if every $\mathbf{v} \in \mathcal{V}$ is equal to a linear combination of the set.

More Definitions (More Linear Algebra):

- A linearly independent set of vectors spanning V is said to be a *basis* of $\mathcal{V}$.
- The *dimension* N of $\mathcal{V}$ is the number of vectors in its basis.
- When N is finite, $\mathcal{V}$ is a *finite-dimensional* vector space.
- Otherwise, $\mathcal{V}$ is said to be $\infty-$dimensional.

Theorem 8 *Any linearly independent set of N vectors from $\mathcal{V}$ forms a basis for $\mathcal{V}$.*

□

Definition 27 *A* **vector subspace** *is any* $\mathcal{W} \subset \mathcal{V}$ *which itself is a vector space under the (inherited) operations of* $\mathcal{V}$*.*

$\square$

Lemma: *To determine if a subset is a subspace, one need test only for closure under each operation.*

Proof: Exercise. $\square$

Theorem 9 *Let $\mathcal{V}$ be a vector space and $\mathcal{W} \subset \mathcal{V}$ such that*

$$\mathcal{W} =< \mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k >, \mathbf{v}_i \in \mathcal{V},\ i = 1, \ldots, k.$$

Then $\mathcal{W}$ is a subspace of $\mathcal{V}$.

Proof:

- $0 \in \mathcal{W}$ by scalar multiplication.
- $\mathbf{u}, \mathbf{w} \in \mathcal{W}$ are linear combinations of $\{\mathbf{v}_i, i = 1, \ldots k\}$.
  - Therefore so is $\mathbf{u} + \mathbf{w}$, hence belongs to $\mathcal{W}$. If $c \in F$, then $c\mathbf{u} \in \mathcal{W}$.
- Similarly, $c \in F \Rightarrow c(\mathbf{u} + \mathbf{v}) \in \mathcal{W}$

Therefore $\mathcal{W}$ is a vector subspace. $\square$

Corollary *If* $\mathcal{W}$ *is a vector subspace of* $\mathcal{V}$ *s.t.* $dim(\mathcal{W}) = dim(\mathcal{V})$*, then* $\mathcal{W} = \mathcal{V}$*.* □

Example: *The* $n-$*tuples over* F*.* Let $a_i \in F,\ i = 1, \ldots, n$

$$(a_1, a_2, \ldots, a_n) \in F^n$$

Note: Any $n-$dimensional vector space is isomorphic to F^n.

Proof: Consider coefficients in the linear combination. □

Definition 28 *The* **scalar** *or* **inner** *product of* $\mathbf{a} = (a_1, \ldots, a_n)$ *and* $\mathbf{b} = (b_1, \ldots, b_n)$ *in* F^n *is*

$$\mathbf{a} \cdot b = \sum_{i=1}^{n} a_i b_i.$$

□

Some Properties:

- $\mathbf{u} \cdot \mathbf{v} = \mathbf{v} \cdot \mathbf{u}$
- $(c\mathbf{u}) \cdot \mathbf{v} = c(\mathbf{u} \cdot \mathbf{v})$
- $w \cdot (\mathbf{u} + \mathbf{v}) = \mathbf{w} \cdot \mathbf{u} + \mathbf{w} \cdot \mathbf{v}$

Orthogonality

- If $\mathbf{u} \cdot \mathbf{v} = 0$, we say that $\mathbf{u}$ is **orthogonal** to $\mathbf{v}$.
- Over finite fields, it is possible that $\mathbf{u} \cdot \mathbf{u} = 0$ (self-orthogonality).
- If $\mathcal{W} = \{w_i, i = 1, \ldots, M\}, \mathcal{W} \subset \mathcal{V}$ and if $\mathbf{u}$ is orthogonal to every $w_i, i = 1, \ldots, M$, then we say $\mathbf{u}$ is orthogonal to $\mathcal{W}$. (This notion requires $\mathcal{V}$ and $\mathcal{W}$ to be sets only.)
- If every member of $\mathcal{U} \subset \mathcal{V}$ is orthogonal to $\mathcal{W} \subset \mathcal{V}$, then we say that $\mathcal{U}$ is the **orthogonal complement** of $\mathcal{W}$.

Theorem 10 *Let $\mathcal{W}$ be a vector subspace of $\mathcal{V}$. The orthogonal complement $\mathcal{U}$ of $\mathcal{W}$ is a vector subspace.*

Proof:

- $0 \in \mathcal{W}$
- Then, for all $\mathbf{u}_1, \mathbf{u}_2 \in \mathcal{U}$ and all $\mathbf{w} \in \mathcal{W}$,

$$\begin{aligned} \mathbf{w} \cdot \mathbf{u}_1 &= 0 \\ \mathbf{w} \cdot \mathbf{u}_2 &= 0 \end{aligned}$$

Therefore,

$$\mathbf{w} \cdot (\mathbf{u}_1 + \mathbf{u}_2) = 0$$

and $(\mathbf{u}_1 + \mathbf{u}_2)$ is a member of the orthogonal complement. This can be shown to hold for $c\mathbf{u}$ as well. $\square$

Notes:

- If a vector $\mathbf{u}$ is orthogonal to every element of the basis of $\mathcal{W}$, then $\mathbf{u}$ is an element of the orthogonal complement of $\mathcal{W}$.
- The orthogonal complement of the orthogonal complement of $\mathcal{W}$ is $\mathcal{W}$ itself.