Notes on MAT240: Algebra 1

Unversity of Toronto

DAVID DUAN

Last Updated: October 29, 2022 (draft)

Contents

Chapter 1

Introduction

Section 1. Intro to Abstract Algebra

1.1 Finite Set

1.1. Definition: A set is a collection of objects, viewed as an object itself. If it has a finite number of element, we call this set a finite set. The **cardinality** of a set is a measure of its size, denoted by

$$|S| \text{ where } \; S \in \mathbb{N}$$

1.2. Definition (Composition): The composition of $f : X \to Y$, and $g : Y \to Z$, is an operation which produces another function $h = g \circ f$. Two functions, f, g are only composable if the $\operatorname{codom}(f) \subseteq \operatorname{dom}(g)$

1.3. Definition (Identity map): The identity map is a speical map on any set X, which maps all members of x to the same element id_X.

$$\begin{aligned} id : X &\to X \\ x &\longmapsto x \end{aligned}$$

1.4. Remark: identity maps do not affect other maps when composed $f \circ I_x = f = I_y \circ f$

1.5. Definition (Category): A category is a collections that consists of objects, and morphisms for each pair of objects, such that

- Any morphism must have domain and codomain which are objects.
- The morphisms can be composed associatively.
- For each object, there exist an identity morphism.

And the simplest category is the category of finite sets.

1.6. Definition: Given a set Y, a subset X of Y, denoted $X \subseteq Y$, is a set for which all elements of X are in Y. The power set $\mathcal{P}(X)$ is the set consisting of all subsets of a set X.
If X_1, X_2 are subsets of Y

- $X_1 \cup X_2 = \{X \in Y : x \in X_1 \text{ or } x \in X_2\}$
- $X_1 \cap X_2 = \{X \in Y : x \in X_1 \text{ and } x \in X_2\}$

$\cup, \cap$ are binary operations on $\mathcal{P}(Y)$

1.7. Remark: If $f : Y \to Z$ is a map and $X \subseteq Y$. We can create a new map $f|_x : X \to Z$ called the "retriction of f to X".

1.2 Classification of Finite Sets

1.8. Definition: A map $f : X \Rightarrow$ is called:

- A map is **injective** when different input implies different output.
  $\forall x, x' \in X, f(x) = f(x') \Rightarrow x = x'$
- A map is **surjective** if every element of its codomain is mapped to by at least one element in its domain.
  $\forall y \in Y, \exists x \in X$ such that $f(x) = y$
- A map is **bijective** if it is both *injective* and *surjective*.

1.9. Definition: Let $\text{bij}(X)$ be the set of bijections of a map. This set is more structured i.e.

- $\text{Bij}(X)$ is eqiupped with a associative binary operation $f, g \in \text{Bij}(X) \Rightarrow f \circ g \in \text{Bij}(X)$.
- A distinguished element I_x.
- There exist an inverse for all elements.

1.10. Definition (image and pre-image): Given $f : X \to Y$, let $C \subseteq X$,the image of C under f is defined as

$$f(C) = \{f(x)|x \in C\}$$

Let $D \subseteq Y$, The *preimage* D under f is defined as

$$f^{-1}(D) = \{x \in X|f(x) \in D\}$$

1.11. Remark: Given a map $f : X \to Y$ we obtain

(1). $\text{Im}f \subseteq Y$

(2). Partition of X into preimages of elements in $\text{Im}f$.

$$P = \{f^{-1}(y) \mid y \in \text{Im}f\}$$

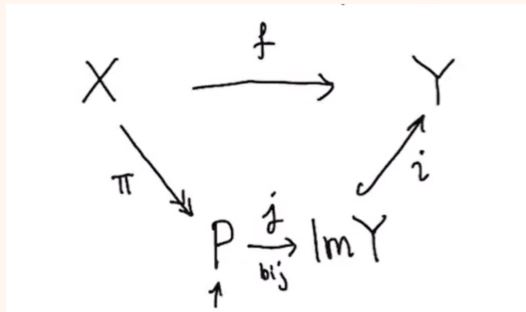
And more precisely, there is a map $j : P \to \text{Im}f$ which sends $f^{-1} \in P \longmapsto y \in \text{Im}f$

We also obtain two other maps from $f : X \to Y$:

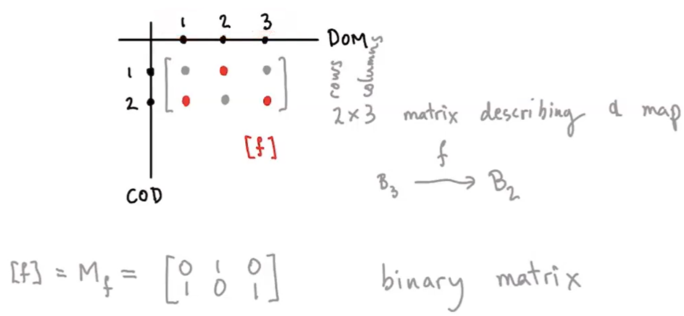
(1). $\pi : X \to P$, which sends $x \in X$ to the preimage that it belongs to, this map is surjective as $x \mapsto f^{-1}(f(x))$.

(2). $i : \text{Im}f \to Y$, which maps $y \in \text{Im}f$ to $y \in Y$. Also called the natural inclusion map for $\text{Im}f \subseteq Y$. This map is injective.

1.12. Proposition: *Any maps $f : X \to Y$ can be factorized into a composition of a surjective, bijective, and injective map:*



1.13. Remark (Explicit description of maps): Instead of drawing arrows, we encode a map The standard set of n elements $n = 0, 1, 2, ...$ is $B_n = \{1, 2, ..., n\}$
For a map $f : B_m \to B_n$, we encode it as a binary matrix as follows:



1.14. Remark (Graphs): Given sets $X_1, X_2, \ldots, X_k$, their cartesian product is a new set defined by

$$\prod_{i=1}^{k} X_i = \{(x_1, x_2, \ldots, x_k) : x_i \in X_i \forall i\}$$

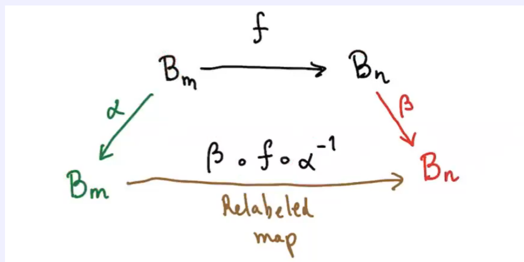
The graph of a map $f : X \to Y$ is the subset of $X \times Y$ defined by

$$\Gamma_f = \{(x, y) \in X \times Y : y = f(x)\}$$

1.15. Definition (Classification of maps): A labeling of a finite set X with cardinality n is a bijection

$$\beta : X \to B_n = \{1, 2, ..., n\}$$

Let f, g be maps $B_m \to B_n$. We say f, g are **"similar"**, and write $f \sim g$ when we can *relabel* the domain and codomain such that $\beta \circ f \circ \alpha^{-1} = g$



1.16. Definition (equivalence relation): A binary relation on a set S is said to be an *equivalence relation* if and only if:

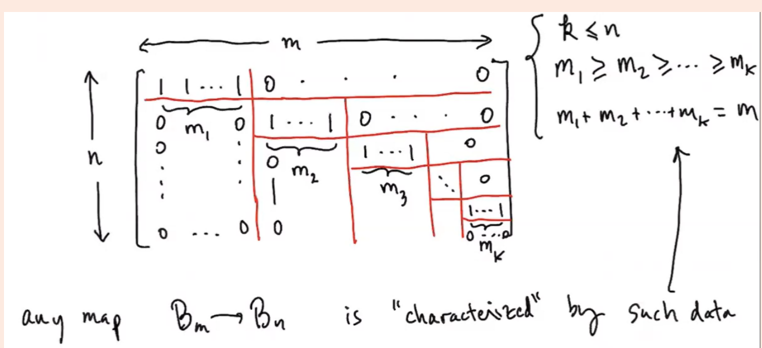
(1). Reflexive: $x \sim x, \forall x \in S$

(2). Symmetric: $x \sim y \Leftrightarrow y \sim x$

(3). Transitive: $x \sim y \text{and} y \sim z \Rightarrow x \sim z$

The similarity of maps is an exmaple of an equivalence relation.

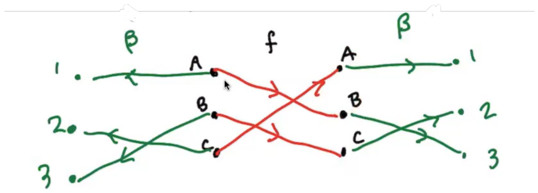
1.17. Theorem: *By relabeling domain and codomain, any map $B_m \to B_n$ is similar to one in "standard form":*



1.18. Definition: Let $f : X \to X$ be a self-map. A *Fixed Point* is an element $x \in X$ such that $f(x) = x$

*When classifying **bijective** self-maps, the strategy used for classifying $f : X \to Y$ is not strict enough and will result in infomation being lost such as fixed points*

To classfiy maps $X \to X$, we should only allow relabeling of the points of X only once.



1.19. Definition (Cycle): A **cycle** for the self-map $f : X \to X$ is a subset $S \subseteq X$ of the form $S = \{x, f(x), f((x)), ...\}$ (iteration of self-map applied to a point x)
Main result: We obtain a partition of X into cycles, (a cycle of length 1 is a fixed point)

1.20. Proposition: *Any bijection $f : X \to X$ of a finite set is a product of disjoint cycles, and every bijcetive self-maps can be classified knowing how many disjoint cycles there are of each length.*

1.21. Warning: This is only a classification for $\mathrm{Bij}(X)$

1.3 Beyond Sets

Equipping sets with "algebraic structure", often called an "operation", which we require to satisfy certain axioms.

1.22. Definition (Operations on a non-empty set):

- 0-ary operation: $* : \{e\} \to A$
- Unary operation: $* : A \to A$
- binary operation: $* : A \times A \to A$
- ternary operation: $* : A \times A \times A \to A$

1.23. Definition (Magma): $(A, *)$ A set with a binary opertaion, no laws.

1.24. Definition (Semigroup): $(A, *)$ A magma s.t. $*$ satisfies associativity.

1.25. Definition (Monoid): $(A, *, e)$ A semigroup with an identity element $e \in A$, s.t. $e * a = a * e = a, \forall a \in A$

1.26. Definition (Group): $(G, *, e, i)$ A monoid with an additional unary operation $i : G \to G$ where i is the inversion operation(denoted g^{-1}).
When $*$ in a group satisfies the additional commutativity axiom, $a * b = b * a, \forall a, b, \in G$, we say the group is *commutative/abelian.* Cyclic groups/modular arithmatic are infinite family of abelian groups.

1.27. Remark: When we start with a set X and use an *equivalence relation* '$\sim$' to produce a set of *eqiuvalence classes*, this is called "taking a quotient", or quotient set. An example of this is $[Z]_n$, or cyclic groups

1.28. Definition (Rings): $((R, +, \circ, i), \cdot, 1)$ An abelian group with an additional associative binary operation with an identity for the operation, as well as distributivity (compatibility between the two binary operations).

1.29. Definition (Field): $(F, +, 0, \cdot, 1)$ A field is a commutative ring such that every non-zero element has a multiplicative inverse.
A sub-field of a field F is a subset $S \subset F$ such that it contains 0 and 1, is closed under addition and multiplication, and have additive and multiplicative inverses.

Section 2. Linear Algebra

2.1 Vector Spaces

2.1. Definition (Vector Spaces): Fix a field F, a vector space V over F is a set V with an Abelian group stucture and an additional binary operation between an element in F and an element in V, which we refer to as scalar multiplication

(1). $(a \cdot_{\mathbb{F}} b) \cdot_s v = a \cdot_s (b \cdot_s v)$

(2). $a \cdot_s (u +_v v) = (a \cdot_s u) +_v (a \cdot_s v)$

(3). $(a +_{\mathbb{F}} b) \cdot_s v = (a \cdot_s v) +_v (b \cdot_s v)$

(4). $1_{\mathbb{F}} \cdot_s v = v$

2.2. Remark (Polynomials): Let $\mathbb{F}$ be any field, $V = \mathcal{P}(\mathbb{F}) = a_0 + a_1 x + ... + a_n x^n : a_i \in \mathbb{F}$. This is a vector space called "polynomials in one variable x with coefficients in $\mathbb{F}$".
Warning:

- Multiplication of polynomials is not part of the vector space structure.
- Polynomials should not always be viewed as functions, as the map from $\mathcal{P}(\mathbb{F})$ to the set of functions $\mathbb{F} \to \mathbb{F}$ is not injective.
  Ex.

$$\mathbb{F} = \mathbb{Z}_2 \quad p = x + x^2 \tag{1.1}$$
$$0 \longmapsto 0 \tag{1.2}$$
$$1 \longmapsto 0 \tag{1.3}$$

  $x + x^2$ and the zero polynomial define the same function.
  Note however if $\mathbb{F} = \mathbb{R}$, it is injective.

2.3. Definition (Function Spaces): Let X be a set and $\mathbb{F}$ a field.
The vector space $V = \mathbb{F}^x = \{f : X \to \mathbb{F}\}$ are all functions on X with values in F. Its defined as:

$$\forall f_1, f_2 \in \mathbb{F}^x \qquad (f_1 +_v f_2) : x \in X \mapsto f_1(x) +_{\mathbb{F}} f_2(x) \in \mathbb{F}$$
$$(\lambda \cdot_s f) : x \in X \mapsto \lambda \cdot_{\mathbb{F}} f(x)$$
$$0_v : x \in X \mapsto 0 \in \mathbb{F}$$

2.4. Definition: Let U, V be vectors spaces over the field $\mathbb{F}$. A **linear map** $L : U \to V$ is a map (morphism) between the sets preserving the structure.

2.5. Definition (Sum):

(1). A sum $u_1 + u_2 = \{u_1 + u_2 : u_1 \in U_1 \text{ and } u_2 \in U_2\}$

(2). A sum $u_1 + u_2$ of subspaces if called **direct** if any vector $v \in u_1 + u_2$ has a unique expression as $v = u_1 + u_2, u_1 \in U_1 \quad u_2 \in U_2$. We write $u_1 \bigoplus u_2$

2.2 Terminologies

2.6. Definition: A **inear subspace** of V is a subset $U \subseteq V$ which inherits the vector space structure form V, i.e.

- $0_v \in U$
- $u_1, u_2 \in U \implies u_1 +_v u_2 \in U$
- $\lambda \in \mathbb{F}, u \in U \implies \lambda \cdot_v u \in U$

2.7. Definition (span): The span of the list of vectors $v_1, ..., v_n$ is the linear combinations of the vectors:

$$span(v_1, ..., v_n) = \{\lambda_1 v_1 + ... + \lambda_n v_n : (\lambda_1, ..., \lambda_n) \in \mathbb{F}^n\}$$

2.8. Remark: If $(v_1, ..., v_k)$ is a list of vectors in V.

$$Span(v_1, ...v_k) = \{\lambda_1 v_1 + \lambda_2 v_2 + ... + \lambda_k v_k : \lambda_1, ..., \lambda_k \in \mathbb{F}\}$$

$$= span(v_1) + span(v_2) + ... + span(v_k)$$

2.9. Definition (Linear dependence): A list of vectors $(v_1, ..., v_k)$ is linearly dependent when it is non-empty and there exists $a_1, ..., a_n \in \mathbb{F}$, not all zero, such that

$$a_1 v_1 + a_2 v_2 + ... + a_n v_n = 0$$

We call this "a non-trivial linera combination equal to zero".
Otherwise, the list is linearly independent.

2.10. Theorem: *Let $(v_1, ..., v_n)$ is a list of non-zero vectors. Then $(v_1, ..., v_n)$ is linearly independent $\Leftrightarrow$ $span(v_1) + ... + span(v_n)$ is direct.*

Proof. $\Rightarrow$ Assume $(v_1, ..., v_n)$ is linearly independent. Then

$$a_1 v_1 + ... + a_n v_n = 0 \implies a_1 = ... = a_n = 0$$

Suppose $span(v_1) + ... + span(v_n)$ is not direct, then by definition.

$$\exists v = u_1 + ... + u_n = w_1 + ... + w_n$$

where $u_i, w_i \in span(v_i)$ and $u_k \neq w_k$ for some k.

$$\begin{aligned} 0 = v - v &= (u_1 - w_1) + ... + (u_k - w_k) + ... + (u_n - w_n) \\ &= \lambda_1 v_1 + ... + \lambda_k v_k + ... \lambda_n v_n \end{aligned}$$

where $\lambda_k \neq 0$. We reach a contradiction as we assumed linear independence, and therefore $span(v_1) + ... + span(v_n)$ is direct.
$\Leftarrow$ Assume $span(v_1) + ... + span(v_n)$ is direct. Then

$$0 \in span(v_1) + ... + span(v_n)$$

$$0 = \quad 0 \quad + ... + \quad 0$$
$$0 = a_1 v_1 + ... + a_n v_n$$

By definition of direct sum, $\forall i \quad a_i v_i = 0 \implies a_i = 0$ since no $v_i = 0$

□

2.11. Definition: A vector space V is **Finite Dimensional** when it is spanned by a finite list of vectors. $V = Span(v_1, ..., v_n)$, otherwise V is infinite dimensional.

2.12. Definition: A **basis** for V is a linearly independent list which spans V. If V is finite-dimensional, dim V is the length of a basis for V.

2.13. Lemma: *If $(v_1, ..., v_n)$ is a linearly dependent, then 1), $\exists v_j$ in the span of previous vectors in the list and 2), we may remove v_j without affecting the span.*

Proof. 1) $(v_1, ..., v_n)$ is linearly dependent implies $\exists (a_1, ..., a_n) \neq (0, ..., 0)$ such that

$$a_1 v_1 + ... + a_n v_n = 0$$

Let a_k be the last nonzero coefficient. Thus

$$v_k = -a_k^{-1}(a_1 v_1, a_2 v_2, ..., a_{n-1} v_{n-1})$$

2) Let $A = span(v_1, ..., v_n), B = span(v_1, ..., v_{k-1}, v_{k+1}, ..., v_n)$. We want to show $A \subseteq B$ and $B \subseteq A$.
$B \subseteq A$: obvious since $(v_1, ..., v_{k-1}, v_{k+1}, ..., v_n)$ is a sublist of $(v_1, ..., v_n)$
$A \subseteq B$: Let $v \in A$, then $v = \lambda_1 v_1 + ... + \lambda_k v_k + ... + \lambda_n v_n$, but $v_k = -a_k^{-1}(a_1 v_1 + ... + a_{k-1} v_{k-1})$, so

$$\begin{aligned} v &= \lambda_1 v_1 + ... + \lambda_{k-1} v_{k-1} + \lambda_k(-a_k^{-1}(a_1 v_1 + ... + a_{k-1} v_{k-1})) + \lambda_{k+1} v_{k+1} + ... + \lambda_n v_n \\ &= (\lambda_1 - \frac{\lambda_k a_1}{a_k}) v_1 + ... + (\lambda_{k-1} - \frac{\lambda_k a_{k-1}}{a_k}) v_{k-1} + \lambda_{k+1} v_{k+1} + ... + \lambda_n v_n) \in B \end{aligned}$$

□

2.14. Theorem: *Length of linear independent list of vectors $\leq$ length of spanning list.*

Proof. Let $(u_1, ..., u_m)$ be linearly independent, let $(w_1, ..., w_n)$ span V. Want to show $m \leq n$.
Algorithm:
We start with the spanning list $(w_1, ..., w_n)$ and we adjoin u_1: $(u_1, w_1, ..., w_n)$. Since $u_1 \in span(w_1, ..., w_n)$, list is linearly dependent and by the Lemma above, $\exists w_j \in span(u_1, w_1, ..., w_{j-1})$, and we eliminate w_j. As a result, $(u_1, ..., w_{j-1}, w_{j+1}, ..., w_n)$ still spans and has the same length n.
We continue and add (u_2): $(u_1, u_2, w_1, ..., w_{j-1}, w_{j+1}, ..., w_n)$, and we can eliminate another w using Lemma, this removed element cannot be a u since $(u_1, ..., u_n)$ is linearly independent.
In this way we can match each u_i with a unique w_j which implies $m \leq n$. □

2.15. Theorem: *If V is finite dimensional, then it has a basis.*

Proof. Since V is finite dimensional, there exist a spanning list $(v_1, ..., v_n)$.
We try to prune this list:

- If $v_1 = 0$, delete.
- Else, move to v_2:
  - If $v_2 \in span(v_1)$, delete v_2
  - If not, move to v_3
- continue for n steps.

In this way, we produce a new list that still spans V, but there doesn't a v_k such that it is in the span of previous vectors which implies that it is linear independent. Thus a basis. □

2.16. Definition: If V is finite dimensional, dim$V$ = length of basis.

2.17. Remark (Fitting Curves to Data): Suppose we have a complicated dataset, and try to measure quantity c_i at point a_i. It is possible to find a polynomial that fits the data perfectly.

$$d_k(x) = \frac{(x-a_0)(x-a_1)\cdots(x-a_{k-1})(x-a_{k+1})\cdots(x-a_n)}{(a_k-a_0)(a_k-a_1)\cdots(a_k-a_{k-1})(a_k-a_{k+1})\cdots(a_k-a_n)})$$

This serves as an indicator function, as

$$d_k(a_i) = \begin{cases} 0 & i \neq k \\ 1 & i = k \end{cases} i = 1, ..., n$$

So if

$$f = c_0 d_0 + c_1 d_1 + ... + c_n d_n$$

This captures the data exactly. We call this Lagrange Interpolation.
Notice: $(d_0, d_1, ..., d_n)$ is another basis for $\mathcal{P}_n(\mathbb{R})$.

Proof. Suppose we have a linear relation $\lambda_0 d_0 + ... + \lambda_n d_n = 0$. We want to show $\lambda_0 = ... = \lambda_n = 0$. If we evaluate relation at $x = a_0$ then this implies $\lambda_0 \cdot 1 = 0$. At $x = a_1 \implies \lambda_1 \cdot 1 = 0$. Therefore, evaluating $x = a_n$ and we get $\lambda_0 = ... = \lambda_n = 0$ and it is linearly independent.
We also have to show that it spans $\mathcal{P}_n(\mathbb{R})$. For any $p \in \mathcal{P}_n(\mathbb{R})$, it can be expressed as a linear combination of the indicator functions, where the coefficients are just the evaluation of the polynomial at those points. So

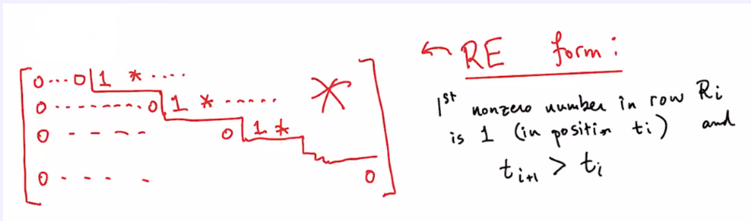
$$p = \sum_{i=0}^{n} p(a_i) d_i$$

We can verify this by noticing that $(d_0, ..., d_n)$ has the same length as the standard basis and is linear independent, which implies it spans.

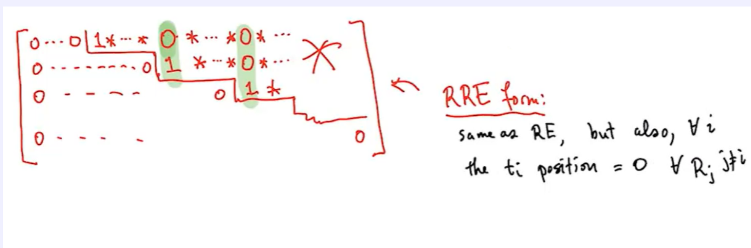
□

2.3 Gaussian Elimination

2.18. Definition (Gaussian Elimination): Input: A list of vectors $(v_1, ..., v_k)$ in V as $k \times n$ matrix of $a_{ij} \in \mathbb{F}$ with respect to a basis $(e_1, ..., e_n)$ of V
Output:
A list of vectors $(w_1, ..., w_k)$ that is more organized such that its matrix relative to the same basis, is in "**Row Echelon Form**".



Or even more simplified, "**Reduced Row Echelon Form**"



With Gaussian Elimination, we can:

- answer whether it is linear independent
- find the dimension of span$(v_1, ..., v_k)$
- find a basis for span$(v_1, ..., v_k)$
- compare spans of two list
- solve linear systems.

2.19. Definition (Elementary Row operations):

- switching: exchange two rows$(v_1, ..., v_i, ..., v_j, ..., v_k) \xmapsto{R_i \Leftrightarrow R_j} (v_1, ..., v_j, ..., v_i, ...v_k)$
- scaling by $\lambda \neq 0$: $(v_1, ..., v_i, ..., v_k) \xmapsto{R_i \to \lambda R_i} (v_1, ..., \lambda v_j, ...v_k)$
- shearing by $\lambda \in \mathbb{F}$: $(v_1, ..., v_i, ..., v_j, ..., v_k) \xmapsto{R_i \to R_i + \lambda R_j} (v_1, ..., v_i + \lambda v_j, ..., v_j, ...v_k)$

Each of these operations are reversible and does not change the span of the list. But they change the list and the matrix representing it.

2.20. Algorithm (Row Echelon Form): "Forward pass"

(1). Step 1.

- Find the row with the earliest non-zero entry a_{ei} and switch it with the first row
- Scale new first row by a_{ei}^{-1}
- For any row with nonzero ith entry, use first row to shear it such that the ith entry becomes 0, i.e. $v_m \mapsto v_m - a_{mi}(v_1)$

(2). Step 2: repeat for $(v_2, ..., v_k)$

(3). Continue and after k steps we will arrive at RE form.

2.21. Algorithm (Reduced Row Echelon Form): "Backward pass"
Let $(v_1, ..., v_k)$ be in RE form, we start at the end of the list

(1). Let e be the echelon position for v_k, use v_k to shear $v_1, ..., v_{k-1}$ so that these rows all have 0 in eth position.

(2). We do the same with v_{k-1} and shear $v_1, ..., v_{k-2}$

Repeat this for k steps to arrive at RRE form.

2.22. Remark (Result of GE): Nonzero rows are obviously linearly independent. Zero rows indicate redundencies in original list. **The nonzero rows is a basis for the original list of vectors.**

2.23. Example: Suppose we have

$$v_1 = ae_1 + be_2$$
$$v_2 = ce_1 + de_2$$

Under what condition is (v_1, v_2) lineraly independent?
We can put this list of vectors as a matrix. $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$
Putting it into RE form, the possibilities are:

$$\begin{pmatrix} 1 & * \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & * \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$

Now it is clear that the last 3 possibilities are not linearly independent. So we focus on the first RE form.

Case 1: If $a \neq 0$

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \xrightarrow{R_1 \to a^{-1}R_1} \begin{pmatrix} 1 & a^{-1}b \\ c & d \end{pmatrix} \xrightarrow{R_2 \to R_2 - c \cdot R_1} \begin{pmatrix} 1 & a^{-1}b \\ 0 & d - ca^{-1}b \end{pmatrix}$$

We also need $d - ca^{-1}b \neq 0$ so we can divide by it
$$\begin{pmatrix} 1 & a^{-1}b \\ 0 & d - ca^{-1}b \end{pmatrix} \xrightarrow{R_2 \to (d-ca^{-1}b)^{-1}R_2} \begin{pmatrix} 1 & a^{-1}b \\ 0 & 1 \end{pmatrix}$$
Since $a \neq 0$ we need $d - ca^{-1}b \neq 0 \implies ad - bc \neq 0$.

Case 2: $a = 0$, then we need $c \neq 0$
$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \xrightarrow{R_1 \Leftrightarrow R_2} \begin{pmatrix} c & d \\ a & b \end{pmatrix} \xrightarrow{R_1 \to c^{-1}R_1} \begin{pmatrix} 1 & c^{-1}d & b - ac^{-1}d \end{pmatrix}$$
Similarly we require $b - ac^{-1}d \neq 0$ so
$$\begin{pmatrix} 1 & c^{-1}d \\ 0 & b - ac^{-1}d \end{pmatrix} \xrightarrow{R_2 \to (b-ac^{-1}d)^{-1}R_2} \begin{pmatrix} 1 & c^{-1}d \\ 0 & 1 \end{pmatrix}$$
So we need $a \neq 0$ and $ad - bc \neq 0$, or $a = 0$ and $c \neq 0$ and $ad - bc \neq 0$.
However, notice that $a = 0$ and $ad - bc \neq 0 \implies c \neq 0$, so we can omit that condition. Now the requirement becomes $ad - bc \neq 0$ in both cases, in other words, we just need $ad - bc \neq 0$.

2.24. Definition (Determinant): From the example above, we see $ad - bc$ determines whether $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$ represents a linearly independent list or not. Because this expression determines the linear independence, we call this the **determinant** of $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$. Usually written as $\det(\begin{pmatrix} a & b \\ c & d \end{pmatrix})$, or $|\begin{pmatrix} a & b \\ c & d \end{pmatrix}|$

2.25. Remark: With GE, we can solve systems of linear equations.
When dealing with a homogenous case, $x = 0$ is always a solution, and the set of solutions is always a linear subspace, since $f_i(x) = 0$ and $f_i(y) = 0 \implies f_i(x + y) = f_i(x) + f_i(y) = 0 + 0 = 0$
So ideally, we should provide a *basis* for the space of solutions. Then a ny solution is a linear combination of the basis.

Duality

2.26. Definition (Linear Functional): A linear functional on V is a linear map from V to F, aka. an element of $\mathcal{L}(V, F)$

2.27. Definition (Dual Space): The **dual space** of V, denoted V^*, is the vector space of all linear functional on V, aka. $V^* = \mathcal{L}(V, F)$

2.28. Remark: Besides linear functions, we also have:

- Constant functions $f : V \to \mathbb{F}$. It is not linear unless $f = 0$ but all constant functions together is a vector space, which is just the field $\mathbb{F}$.
- Affine-linear functions: the space of affine-linear functions is $V^* \oplus \mathbb{F}$

Result: given a list of functions to solve, we can view them as a list of vectors and apply GE to this list.

2.29. Remark: If we know values of $f \in V^*$ on a basis β, we know f completely.
If $(f(e_1) = b_1, ..., f(e_n) = b_n)$, then $f(v = a_1e_1 + ... + a_ne_n) = a_1f(e_1) + ... + a_nf(e_n) = a_1b_1 + ... + a_nv_n$.

2.30. Definition (Dual basis): If $\beta = (e_1, ..., e_n)$ is a basis for V, we can use it to produce a **dual basis** for V^*. We use the same strategy as Lagrange Interpolation and define $\beta^* = (e_1^*, ..., e_n^*)$ to be: e_i^* = the linear function taking value 1 on e_i, 0 on $e_{j \neq i}$.
Thus for $f \in V^*$, $f = f(e_1)e_1^* + ... + f(e_n)e_n^*$