[6] Dimension

The size of a basis

Key fact for this week: all bases for a vector space have the same size.

We use this as the "basis" for answering many pending questions.

Morphing Lemma

Morphing Lemma: Suppose S is a set of vectors, and B is a linearly independent set of vectors in Span S. Then $|S| \geq |B|$.

Before we prove it—what good is this lemma?

Theorem: Any basis for $\mathcal{V}$ is a smallest generating set for $\mathcal{V}$.

Proof: Let S be a smallest generating set for $\mathcal{V}$. Let B be a basis for $\mathcal{V}$. Then B is a linearly independent set of vectors in Span S. By the Morphing Lemma, B is no bigger than S, so B is also a smallest generating set.

Theorem: All bases for a vector space $\mathcal{V}$ have the same size.

Proof: They are all smallest generating sets.

Proof of the Morphing Lemma

Morphing Lemma: Suppose S is a set of vectors, and B is a linearly independent set of vectors in Span S. Then $|S| \geq |B|$.

Proof outline: modify S step by step, introducing vectors of B one by one, without increasing the size.

How? Using the Exchange Lemma....

Review of Exchange Lemma

Exchange Lemma: Suppose S is a set of vectors and A is a subset of S. Suppose $\mathbf{z}$ is a vector in Span S such that $A \cup \{\mathbf{z}\}$ is linearly independent.
Then there is a vector $\mathbf{w} \in S - A$ such that

$$\text{Span } S = \text{Span } (S \cup \{\mathbf{z}\} - \{\mathbf{w}\})$$

Proof of the Morphing Lemma

Let $B = \{\mathbf{b}_1, \ldots, \mathbf{b}_n\}$. Define $S_0 = S$.
Prove by induction on $k \leq n$ that there is a generating set S_k of Span S that contains $\mathbf{b}_1, \ldots, \mathbf{b}_k$ and has size $|S|$.

Base case: $k = 0$ is trivial.

To go from S_{k-1} to S_k : use the Exchange Lemma.

- $A_k = \{\mathbf{b}_1, \ldots, \mathbf{b}_{k-1}\}$ and $\mathbf{z} = \mathbf{b}_k$

Exchange Lemma $\Rightarrow$ there is a vector $\mathbf{w}$ in S_{k-1} such that

$$\text{Span } (S_{k-1} \cup \{\mathbf{b}_k\} - \{\mathbf{w}\}) = \text{Span } S_{k-1}$$

Set $S_k = S_{k-1} \cup \{\mathbf{b}_k\} - \{\mathbf{w}\}$.

QED

This induction proof is an algorithm.

Dimension

Definition: We define the *dimension* of a vector space to be the size of a basis for that vector space. The dimension of a vector space $\mathcal{V}$ is written $\dim \mathcal{V}$.

Definition: We define the *rank* of a set S of vectors as the dimension of Span S. We write rank S.

Example: The vectors [1, 0, 0], [0, 2, 0], [2, 4, 0] are linearly dependent.
Therefore their rank is less than three.
First two of these vectors form a basis for the span of all three, so the rank is two.

Example: The vector space Span {[0, 0, 0]} is spanned by an empty set of vectors.
Therefore the rank of {[0, 0, 0]} is zero.

Row rank, column rank

Definition: For a matrix M, the *row rank* of M is the rank of its rows, and the *column rank* of M is the rank of its columns.
Equivalently, the row rank of M is the dimension of Row M, and the column rank of M is the dimension of Col M.

Example: Consider the matrix

$$M = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 2 & 4 & 0 \end{bmatrix}$$

whose rows are the vectors we saw before: [1, 0, 0], [0, 2, 0], [2, 4, 0]
The set of these vectors has rank two, so the row rank of M is two.
The columns of M are [1, 0, 2], [0, 2, 4], and [0, 0,0].
Since the third vector is the zero vector, it is not needed for spanning the column space.
Since each of the first two vectors has a nonzero where the other has a zero, these two are linearly independent, so the column rank is two.

Row rank, column rank

Definition: For a matrix M, the *row rank* of M is the rank of its rows, and the *column rank* of M is the rank of its columns.
Equivalently, the row rank of M is the dimension of Row M, and the column rank of M is the dimension of Col M.

Example: Consider the matrix

$$M = \begin{bmatrix} 1 & 0 & 0 & 5 \\ 0 & 2 & 0 & 7 \\ 0 & 0 & 3 & 9 \end{bmatrix}$$

Each of the rows has a nonzero where the others have zeroes, so the three rows are linearly independent. Thus the row rank of M is three.
The columns of M are [1, 0, 0], [0, 2, 0], [0, 0, 3], and [5, 7, 9].
The first three columns are linearly independent, and the fourth can be written as a linear combination of the first three, so the column rank is three.

Row rank, column rank

Definition: For a matrix M, the *row rank* of M is the rank of its rows, and the *column rank* of M is the rank of its columns.

Equivalently, the row rank of M is the dimension of Row M, and the column rank of M is the dimension of Col M.

Does column rank always equal row rank? Q

Cardinality of a vector space over *GF* (2)

Recall *checksum problem*
Checksum function $\mathbf{x} \mapsto [\mathbf{a}_1 \cdot \mathbf{x}, \ldots, \mathbf{a}_{64} \cdot \mathbf{x}]$

Original "file" $\mathbf{p}$, transmission error $\mathbf{e}$ so corrupted file is $\mathbf{p} + \mathbf{e}$.

What is probability that corrupted file has the same checksum as original?

If error is chosen according to uniform distribution,

$$
\begin{aligned}
&\text{Probability } (\mathbf{p} + \mathbf{e} \text{ has same checksum as } \mathbf{p}) \\
&\quad = \text{Probability } (\mathbf{e} \text{ is a solution to homogeneous linear system}) \\
&\quad = \frac{\text{number of solutions to homogeneous linear system}}{\text{number of } n\text{-vectors}} \\
&\quad = \frac{\text{number of solutions to homogeneous linear system}}{2^n}
\end{aligned}
$$

raising Question

How to find number of solutions to a homogeneous linear system over *GF* (2)?

Cardinality of a vector space over $GF(2)$

How to find number of solutions to a homogeneous linear system over $GF(2)$?

Solution set of a homogeneous linear system is a vector space.
Question becomes

How to find out cardinality of a vector space $\mathcal{V}$ over $GF(2)$?

- Suppose basis for $\mathcal{V}$ is $\mathbf{b}_1, \ldots, \mathbf{b}_n$.
- Then $\mathcal{V}$ is set of linear combinations
$$\beta_1 \mathbf{b}_1 + \cdots + \beta_n \mathbf{b}_n$$
- Number of linear combinations is 2^n.
- By Unique-Representation Lemma, every linear combination gives a different vector of $\mathcal{V}$.
- Thus cardinality is $2^{\dim \mathcal{V}}$.

Cardinality of a vector space over $GF(2)$

Cardinality of a vector space $\mathcal{V}$ over $GF(2)$ is $2^{\dim \mathcal{V}}$.

$\boxed{\text{How to find dimension of solution set of a homogeneous linear system?}}$

Write linear system as $A\mathbf{x} = \mathbf{0}$.

$\boxed{\text{How to find dimension of the null space of } A\text{?}}$

Answers will come later.

Subset-Basis Lemma

Lemma: Every finite set T of vectors contains a subset S that is a basis for Span T.

Proof: The Grow algorithm finds a basis for $\mathcal{V}$ if it terminates.

> Initialize $S = \emptyset$.
> Repeat while possible: select a vector **v** in $\mathcal{V}$ that is not in Span S, and put it in S.

Revised version:

> Initialize $S = \emptyset$
> Repeat while possible: select a vector **v** in T that is not in Span S, and put it in S.

Differs from original:

- This algorithm stops when Span S contains every vector in T.
- The original Grow algorithm stops only once Span S contains every vector in $\mathcal{V}$.

However, that's okay: when Span S contains all the vectors in T, Span S also contains all linear combinations of vectors in T, so at this point Span $S = \mathcal{V}$.
Shows that original Grow algorithm can be guided to make same choices as this algorithm, so result is a basis.

QED

Termination of Grow algorithm

```
def Grow(V)
  B = ∅
  repeat while possible:
        find a vector v in V that is not in Span B, and put it in S.
```

Grow-Algorithm-Termination Lemma: If $\mathcal{V}$ is a subspace of $\mathbb{F}^D$ where D is finite then $\text{Grow}(\mathcal{V})$ terminates.

Proof: By Grow-Algorithm Corollary, B is linearly independent throughout.

Apply the Morphing Lemma with $S = \{\text{standard generators for } \mathbb{F}^D\}$ $\Rightarrow$

$|B| \leq |S| = |D|$.

Since B grows in each iteration, there are at most $|D|$ iterations. QED

Every subspace of $\mathbb{F}^D$ contains a basis

Grow-Algorithm-Termination Lemma: If $\mathcal{V}$ is a subspace of $\mathbb{F}^D$ where D is finite then $\text{Grow}(\mathcal{V})$ terminates.

Theorem: For finite D, every subspace of $\mathbb{F}^D$ contains a basis.

Proof: Let $\mathcal{V}$ be a subspace of $\mathbb{F}^D$.

```
def Grow(V)
  B = ∅
  repeat while possible:
        find a vector v in V that is not in Span B,   and put it in B.
```

Grow-Algorithm-Termination Lemma ensures algorithm terminates.
Upon termination, every vector in $\mathcal{V}$ is in Span B, so B is a set of generators for $\mathcal{V}$. By Grow-Algorithm Corollary, B is linearly independent. Therefore B is a basis for $\mathcal{V}$. QED

Superset-Basis Lemma

Grow-Algorithm-Termination Lemma: If $\mathcal{V}$ is a subspace of $\mathbb{F}^D$ where D is finite then $\texttt{Grow}(\mathcal{V})$ terminates.

Superset-Basis Lemma: Let $\mathcal{V}$ be a vector space consisting of D-vectors where D is finite. Let C be a linearly independent set of vectors belonging to $\mathcal{V}$. Then $\mathcal{V}$ has a basis B containing all vectors in C.

Proof: Use version of Grow algorithm:

Initialize B to the empty set. Repeat while possible: select a vector $\mathbf{v}$ in $\mathcal{V}$ (preferably in C) that is not in Span B, and put it in B.

At first, B will consist of vectors in C until B contains all of C. Then more vectors will be added to B until Span $B = \mathcal{V}$ By Grow-Algorithm Corollary, B is linearly independent throughout. Therefore, once algorithm terminates, B contains C and is a basis for $\mathcal{U}$.

Termination is implied by Grow Algorithm Termination Lemma. QED

Estimating dimension

$T = \{[-0.6, -2.1, -3.5, -2.2], [-1.3, 1.5, -0.9, -0.5], [4.9, -3.7, 0.5, -0.3],$
$[2.6, -3.5, -1.2, -2.0], [-1.5, -2.5, -3.5, 0.94]\}$.
What is the rank of T ?

By Subset-Basis Lemma, T contains a basis.

Therefore dim Span $T \leq |T|$.

Therefore rank $T \leq |T|$.

Proposition: A set T of vectors has rank $\leq |T|$.

Dimension Lemma

Dimension Lemma: If $\mathcal{U}$ is a subspace of $\mathcal{W}$ then

- **D1:** $\dim \mathcal{U} \leq \dim \mathcal{W}$, and
- **D2:** if $\dim \mathcal{U} = \dim \mathcal{W}$ then $\mathcal{U} = \mathcal{W}$

Proof: Let $\mathbf{u}_1, \ldots, \mathbf{u}_k$ be a basis for $\mathcal{U}$.
By Superset-Basis Lemma, there is a basis B for $\mathcal{W}$ that contains $\mathbf{u}_1, \ldots, \mathbf{u}_k$.

- $B = \{\mathbf{u}_1, \ldots, \mathbf{u}_k, \mathbf{b}_1, \ldots, \mathbf{b}_r\}$
- Thus $k \leq |B|$, and
- If $k = |B|$ then $\{\mathbf{u}_1, \ldots, \mathbf{u}_k\} = B$ QED

Example: Suppose $\mathcal{V} = \text{Span } \{[1, 2], [2, 1]\}$.
Clearly $\mathcal{V}$ is a subspace of $\mathbb{R}^2$.
However, the set $\{[1, 2], [2, 1]\}$ is linearly independent, so $\dim \mathcal{V} = 2$.
Since $\dim \mathbb{R}^2 = 2$, D2 shows that $\mathcal{V} = \mathbb{R}^2$.

Example: $S = \{[-0.6, -2.1, -3.5, -2.2], [-1.3, 1.5, -0.9, -0.5], [4.9, -3.7, 0.5, -0.3], [2.6, -3.5, -1.2, -2.0], [-1.5, -2.5, -3.5, 0.94]\}$
Since every vector in S is a 4-vector, Span S is a subspace of $\mathbb{R}^4$.
Since $\dim \mathbb{R}^4 = 4$, D1 shows $\dim \text{Span } S \leq 4$.

Proposition: Any set of D-vectors has rank at most $|D|$.

Rank Theorem

Rank Theorem: For every matrix M, row rank equals column rank.

Lemma: For any matrix A, row rank of $A \leq$ column rank of A

To show theorem:

- Apply lemma to $M \Rightarrow$ row rank of $M \leq$ column rank of M
- Apply lemma to $M^T \Rightarrow$ row rank of $M^T \leq$ column rank of $M^T \Rightarrow$ column rank of $M \leq$ row rank of M

Combine $\Rightarrow$ row rank of $M =$ column rank of M

Direct Sum

Let $\mathcal{U}$ and $\mathcal{V}$ be two vector spaces consisting of *D*-vectors over a field $\mathbb{F}$.

Definition: If $\mathcal{U}$ and $\mathcal{V}$ share only the zero vector then we define the *direct sum* of $\mathcal{U}$ and $\mathcal{V}$ to be the set

$$\{\mathbf{u} + \mathbf{v} : \mathbf{u} \in \mathcal{U}, \mathbf{v} \in \mathcal{V}\}$$

written $\mathcal{U} \oplus \mathcal{V}$

That is, $\mathcal{U} \oplus \mathcal{V}$ is the set of all sums of a vector in $\mathcal{U}$ and a vector in $\mathcal{V}$.

In Python, `[u+v for u in U for v in V]`

(But generally $\mathcal{U}$ and $\mathcal{V}$ are infinite so the Python is just suggestive.)

Direct Sum: Example

Vectors over *GF* (2):

Example: Let $\mathrm{U} =$ Span $\{1000,\ 0100\}$ and let $\mathrm{V} =$ Span $\{0010\}$.

- Every nonzero vector in U has a one in the first or second position (or both) and nowhere else.

- Every nonzero vector in V has a one in the third position and nowhere else.

Therefore the only vector in both U and V is the zero vector.

Therefore $\mathrm{U} \oplus \mathrm{V}$ is defined.

$\mathrm{U} \oplus \mathrm{V} = \{0000 + 0000, 1000 + 0000, 0100 + 0000, 1100 + 0000, 0000 + 0010, 1000 + 0010, 0100 + 0010, 1100 + 0010\}$

which is equal to $\{0000, 1000, 0100, 1100, 0010, 1010, 0110, 1110\}$.

Direct Sum: Example

Vectors over $\mathbb{R}$:

Example: Let $\mathrm{U} =$ Span $\{[1, 2, 1, 2], [3, 0, 0, 4]\}$ and let V be the null space of $\begin{bmatrix} 0 & 1 & -1 & 0 \\ 1 & 0 & 0 & -1 \end{bmatrix}$.

- The vector $[2, -2, -1, 2]$ is in U because it is $[3, 0, 0, 4] - [1, 2, 1, 2]$
- It is also in V because

$$\begin{bmatrix} 0 & 1 & -1 & 0 \\ 1 & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} 2 \\ -2 \\ -1 \\ 2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Therefore we cannot form $\mathrm{U} \oplus \mathrm{V}$.

Matrix invertibility

Rank-Nullity Theorem: For any n-column matrix A,

$$\text{nullity } A + \text{rank } A = n$$

Corollary: Let A be an $R \times C$ matrix. Then A is invertible if and only if $|R| = |C|$ and the columns of A are linearly independent.

Proof: Let F be the field. Define $f : \mathbb{F}^C \longrightarrow \mathbb{F}^R$ by $f(\mathbf{x}) = A\mathbf{x}$.
Then A is an invertible matrix if and only if f is an invertible function.

The function f is invertible iff dim Ker $f = 0$ and dim $\mathbb{F}^C =$ dim $\mathbb{F}^R$
iff nullity $A = 0$ and $|C| = |R|$.

nullity $A = 0$ iff dim Null $A = 0$
iff Null $A = \{\mathbf{0}\}$
iff the only vector $\mathbf{x}$ such that $A\mathbf{x} = \mathbf{0}$ is $\mathbf{x} = \mathbf{0}$
iff the columns of A are linearly independent. QED

Matrix invertibility examples

$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$ is not square so cannot be invertible.

$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ is square and its columns are linearly independent so it is invertible.

$\begin{bmatrix} 1 & 1 & 2 \\ 2 & 1 & 3 \\ 3 & 1 & 4 \end{bmatrix}$ is square but columns not linearly independent so it is not invertible.

Transpose of invertible matrix is invertible

Theorem: The transpose of an invertible matrix is invertible.

$$A = [v_1 | \ldots | v_n] = \begin{bmatrix} \underline{a_1} \\ \vdots \\ \overline{a_n} \end{bmatrix} \qquad A^T = [a_1 | \ldots | a_n]]$$

Proof: Suppose A is invertible. Then A is square and its columns are linearly independent. Let n be the number of columns. Then rank $A = n$.

Because A is square, it has n rows. By Rank Theorem, rows are linearly independent.

Columns of transpose A^T are rows of A, so columns of A^T are linearly independent.

Since A^T is square and columns are linearly independent, A^T is invertible. QED

[7] Gaussian Elimination

Gaussian elimination: Uses

- *Finding a basis for the span of given vectors*. This additionally gives us an algorithm for rank and therefore for testing linear dependence.
- *Solving a matrix equation*,which is the same as *expressing a given vector as a linear combination of other given vectors*, which is the same as *solving a system of linear equations*
- *Finding a basis for the null space of a matrix*, which is the same as *finding a basis for the solution set of a homogeneous linear system*, which is also relevant to representing the solution set of a general linear system.

Echelon form

Echelon form a generalization of triangular matrices

Example: $$\begin{bmatrix} 0 & 2 & 3 & 0 & 5 & 6 \\ 0 & 0 & 1 & 0 & 3 & 4 \\ 0 & 0 & 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 & 0 & 9 \end{bmatrix}$$

Note that

- the first nonzero entry in row 0 is in column 1,
- the first nonzero entry in row 1 is in column 2,
- the first nonzero entry in row 2 is in column 4, and
- the first nonzero entry in row 4 is in column 5.

Definition: An $m \times n$ matrix A is in *echelon form* if it satisfies the following condition: for any row, if that row's first nonzero entry is in position k then every previous row's first nonzero entry is in some position less than k.

Echelon form

Definition: An $m \times n$ matrix A is in *echelon form* if it satisfies the following condition: for any row, if that row's first nonzero entry is in position k then every previous row's first nonzero entry is in some position less than k.

This definition implies that, as you iterate through the rows of A, the first nonzero entries per row move strictly right, forming a sort of staircase that descends to the right.

$$\begin{bmatrix} 2 & 1 & 0 & 4 & 1 & 3 & 9 & 7 \\ 0 & 6 & 0 & 1 & 3 & 0 & 4 & 1 \\ 0 & 0 & 0 & 0 & 2 & 1 & 3 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 2 & 3 & 0 & 5 & 6 \\ 0 & 0 & 1 & 0 & 3 & 4 \\ 0 & 0 & 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 & 0 & 9 \end{bmatrix} \qquad \begin{bmatrix} 4 & 1 & 3 & 0 \\ 0 & 3 & 0 & 1 \\ 0 & 0 & 1 & 7 \\ 0 & 0 & 0 & 9 \end{bmatrix}$$

Echelon form

Definition: An $m \times n$ matrix A is in *echelon form* if it satisfies the following condition: for any row, if that row's first nonzero entry is in position k then any previous row's first nonzero entry is in some position less than k.

If a row of a matrix in echelon form is all zero then every subsequent row must also be all zero, e.g.

$$\begin{bmatrix} 0 & 2 & 3 & 0 & 5 & 6 \\ 0 & 0 & 1 & 0 & 3 & 4 \\ 0 & 0 & 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 & 0 & 9 \end{bmatrix}$$

Uses of echelon form

What good is it having a matrix in echeleon form?

Lemma: If a matrix is in echelon form, the nonzero rows form a basis for the row space.

For example, a basis for the row space of

$$\begin{bmatrix} 0 & 2 & 3 & 0 & 5 & 6 \\ 0 & 0 & 1 & 0 & 3 & 4 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

is {[0, 2, 3, 0, 5, 6], [0, 1, 0, 3, 4]}.

In particular, if every row is nonzero, as in each of the matrices

$$\begin{bmatrix} 0 & 2 & 3 & 0 & 5 & 6 \\ 0 & 0 & 1 & 0 & 3 & 4 \\ 0 & 0 & 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 & 0 & 9 \end{bmatrix}, \begin{bmatrix} 2 & 1 & 0 & 4 & 1 & 3 & 9 & 7 \\ 0 & 6 & 0 & 1 & 3 & 0 & 4 & 1 \\ 0 & 0 & 0 & 0 & 2 & 1 & 3 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 4 & 1 & 3 & 0 \\ 0 & 3 & 0 & 1 \\ 0 & 0 & 1 & 7 \\ 0 & 0 & 0 & 9 \end{bmatrix}$$

then the rows form a basis of the rowspace.

Uses of echelon form

Lemma: If matrix is in echelon form, the nonzero rows form a basis for row space.

It is obvious that the nonzero rows span the row space. We need only show that these vectors are linearly independent. We prove it using the Grow algorithm:

```
def Grow(V)
  S = ;
  repeat while possible:
      find a vector v in V that is not in Span S, and put it in S
```

$$\begin{bmatrix} 4 & 1 & 3 & 0 \\ 0 & 3 & 0 & 1 \\ 0 & 0 & 1 & 7 \\ 0 & 0 & 0 & 9 \end{bmatrix}$$

We run the Grow algorithm, adding rows of matrix in reverse order to S:

- Since Span ; does not include [0, 0, 0, 9], the algorithm adds this vector to S.
- Now $S = \{[0, 0, 0, 9]\}$. Every vector in Span S has zeroes in positions 0, 1, 2, so Span S does not contain [0, 0, 1, 7], so the algorithm adds this vector to S.
- Now $S = \{[0, 0, 0, 9], [0, 0, 1, 7]\}$. Every vector in Span S has zeroes in positions 0, 1, so Span S does not contain [0, 3, 0, 1], so the algorithm adds it.
- Now $S = \{[0, 0, 0, 9], [0, 0, 1, 7], [0, 3, 0, 1]\}$. Every vector in Span S has a zero in position 0, so Span S does not contain [4, 1, 3, 0], so the algorithm adds it, and we are done.

Elementary row-addition operations

$$\begin{bmatrix} 0 & 2 & 3 & 4 & 5 \\ 0 & 0 & 0 & 3 & 2 \\ 1 & 2 & 3 & 4 & 5 \\ 0 & 0 & 0 & 6 & 7 \end{bmatrix} \Longrightarrow \begin{bmatrix} 0 & 2 & 3 & 4 & 5 \\ 0 & 0 & 0 & 3 & 2 \\ 1 & 2 & 3 & 4 & 5 \\ 0 & 0 & 0 & 0 & 3 \end{bmatrix}$$

You can keep the row space by adding multiples of the rows to other rows:

Subtract twice the secondrow

$$2\,[0, 0, 0, 3, 2]$$

from the fourth

$$[0, 0, 0, 6, 7]$$

gettting new fourth row

$$[0, 0, 0, 6, 7] - 2\,[0, 0, 0, 3, 2] = [0, 0, 0, 6 - 6, 7 - 4] = [0, 0, 0, 0, 3]$$

The 3 in the second row is called the *pivot element*.
That element is used to zero out another element in same column.

Elementary row-addition operations

Transformation is multiplication by a *elementary row-addition matrix*:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -2 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 2 & 3 & 4 & 5 \\ 0 & 0 & 0 & 3 & 2 \\ 1 & 2 & 3 & 4 & 5 \\ 0 & 0 & 0 & 6 & 7 \end{bmatrix} = \begin{bmatrix} 0 & 2 & 3 & 4 & 5 \\ 0 & 0 & 0 & 3 & 2 \\ 1 & 2 & 3 & 4 & 5 \\ 0 & 0 & 0 & 6 & 7 \end{bmatrix}$$

Such a matrix is invertible:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -2 & 0 & 1 \end{bmatrix} \text{ and } \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 \end{bmatrix} \text{ are inverses}$$

We will show:

Proposition: If $MA = B$ where M is invertible then Row A = Row B.

Therefore change to row causes no change in row space.

Therefore basis for changed rowlist is also a basis for original rowlist.

Preserving row space

Lemma: $Row\ NA \subseteq Row\ A$.

Proof: Let **v** be any vector in Row NA.
That is, **v** is a linear combination of the rows of NA.
By the linear-combinations definition of vector-matrix multiplication, there is a vector **u** such that

$$\boldsymbol{v} = [u^T]([N][A]) = ([u^T][N])[A] \text{ by associativity}$$

which shows that **v** can be written as a linear combination of the rows of A.
QED

Gaussian elimination

Applying elementary row-addition operations does not change the row space.

Incorporate into the algorithm

```
for c in col_label_list:
rows_with_nonzero = [ r for r in rows left if rowlist[r][c] ! = 0]
 if rows with nonzero ! = [] :
    pivot = rows_with_nonzero[0]
    rows_left.remove(pivot )
    new_rowlist.append(rowlist[pivot])
    for r in rows_with_nonzero[1 :]:
    multiplier = rowlist[r][c]/ rowlist[ pivot][c]
    rowlist[r] -= multiplier * rowlist[pivot]
```

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 5 \\ 3 & 4 & 5 & 6 \\ 4 & 5 & 6 & 7 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 2 & 3 & 4 \\ 0 & -1 & -2 & -3 \\ 0 & -2 & -4 & -6 \\ 0 & -3 & -6 & -9 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 2 & 3 & 4 \\ 0 & -1 & -2 & -3 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

This algorithm is mathematically correct...

Failure of Gaussian elimination

But we compute using floating-point numbers!

$$\begin{bmatrix} 10^{-20} & 0 & 1 \\ 1 & 10^{20} & 1 \\ 0 & 1 & -1 \end{bmatrix} \Longrightarrow \begin{bmatrix} 10^{-20} & 0 & 1 \\ 0 & 10^{20} & 1-10^{20} \\ 0 & 1 & -1 \end{bmatrix} \Longrightarrow \begin{bmatrix} 10^{-20} & 0 & 1 \\ 0 & 10^{20} & -10^{20} \\ 0 & 1 & -1 \end{bmatrix}$$

$$\Longrightarrow \begin{bmatrix} 10^{-20} & 0 & 1 \\ 0 & 10^{20} & -10^{20} \\ 0 & 0 & 0 \end{bmatrix}$$

Gaussian elimination got the wrong answer due to round-off error.

These problems can be mitigated by choosing the pivot element carefully:

- *Partial pivoting:* Among rows with nonzero entries in column c, choose row with entry having *largest* absolute value.
- *Complete pivoting*: Instead of selecting order of columns beforehand, in each iteration choose column to maximize absolute value of pivot element.

In this course, we won't study these techniques in detail.
Instead, we will use Gaussian elimination only for $GF(2)$.

Gaussian elimination for *GF* (2)

		A	B	C	D
	0	0	0	1	1
✓	1	1	0	1	1
	2	1	0	0	1
	3	1	1	1	1

A: Select row 1 as pivot. Put it in new_rowlist Since rows 2 and 3 have nonzeroes, we must add row 1 to rows 2 and 3.

new_row_list

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

		A	B	C	D
	0	0	0	1	1
✓	1	1	0	1	1
	2	0	0	1	0
✓	3	0	1	0	0

B: Select row 3 as pivot. Put it in new_rowlist Other remaining rows have zeroes in column B, so no row additions needed.

new_row_list

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

		A	B	C	D
✓	0	0	0	1	1
✓	1	1	0	1	1
	2	0	0	1	0
✓	3	0	1	0	0

C: Select row 0 as pivot . Put it in new_rowlist. Only other remaining row is row 2, and we add row 0 to row 2.

new_row_list

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

Gaussian elimination for *GF* (2)

new_row_list

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

		A	B	C	D
✓	0	0	0	1	1
✓	1	1	0	1	1
✓	2	0	0	1	0
✓	3	0	1	0	0

D: Only remaining row is row 2, so select it as pivot row.
Put it in new_rowlist
No other rows, so no row additions.

new_row_list

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