BOOK: Linear Algebra: Theory, Intuition, Code

AUTHOR: Mike X Cohen

WEBSITE: sincxpress.com

CHAPTER: Matrix spaces (chapter 8)

In [1]:
```
## import libraries for the entire chapter
import numpy as np
from scipy.linalg import null_space
```

Section 8.1 Column space of a matrix

Lo spazio delle colonne di una matrice è il sottospazio occupato da tutte le colonne di quella matrice. Lo spazio delle colonne è talvolta chiamato anche intervallo o immagine di una matrice. In altre parole, pensa a una matrice come a un insieme di vettori colonna e al sottospazio compreso da quell'insieme di vettori come allo spazio colonna della matrice. Lo spazio delle colonne è indicato utilizzando la notazione $C(\mathbf{A})$.

Column space of a matrix

$$C(\mathbf{A}) = \{\beta_1 a_1 + ... + \beta_n a_n, \quad \beta \in \mathbb{R}\} \qquad (8.3)$$

$$C(\mathbf{A}) = span(a_1, ..., a_n) \qquad (8.4)$$

Lo spazio colonna di una matrice *M*x*N* è R^M perché la colonna ha *N* elementi

$$\mathbf{B} = \begin{bmatrix} 1 & 2 & 4 \\ 0 & 4 & 4 \\ 4 & 1 & 9 \\ 6 & 0 & 12 \\ 1 & 1 & 3 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 1 & 2 & 3 & 5 \\ 0 & 8 & 13 & 21 \\ 0 & 0 & 34 & 55 \\ 0 & 0 & 0 & 89 \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} 1 & 5 & -3 & 4 \\ 2 & 6 & -2 & 4 \\ 3 & 7 & -1 & 4 \\ 4 & 8 & 0 & 4 \\ 5 & 9 & 1 & 4 \\ 6 & 10 & 2 & 4 \\ 7 & 11 & 3 & 4 \end{bmatrix}$$

Column space = 5 4 7

Section 8.2, The column space of **A** and **AA**^{T}

Esse hanno spazio colonna della stessa dimensione. D'altra parte, **A** e **AA**^{T} generalmente non hanno esattamente le stesse colonne. Pertanto, le colonne di queste due matrici si estendono sullo stesso sottospazio, ma possono avere insiemi di basi diversi.

Section 8.3, Determining weather **v** ∈ C(**A**)

Sì se **v** è ottenibile mediante combinazione lineare delle colonne di **A**

Esempio numerico:

$$\begin{bmatrix} 2 & 1 \\ 4 & 4 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 4 \\ 12 \\ 0 \end{bmatrix} \quad \Rightarrow \quad \mathbf{Ax} = \mathbf{v} \qquad (8.6)$$

Se esiste un **x** che soddisfa la suddetta eqazione matriciale --> **v** ∈ *C*(**A**) --- con la Geometria:

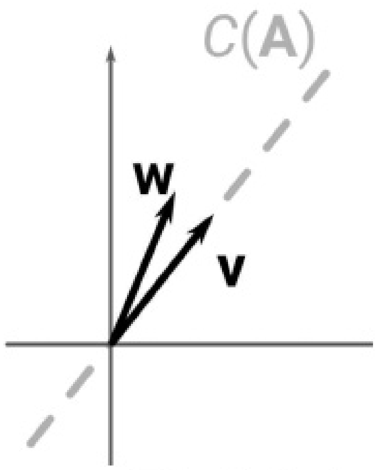


Figure 8.1: The dashed gray line represents $C(\mathbf{A}) \in \mathbb{R}^2$. Then $\mathbf{v} \in C(\mathbf{A})$ and $\mathbf{w} \notin C(\mathbf{A})$

Con l'Algebra: aumento **A** con **v**; se il rango non varia --> **v** ∈ *C*(**A**)

Section 8.4, Row space of a matrix

E' indicato con $R(\mathbf{A})$ --- vale: $R(\mathbf{A}) = C(\mathbf{A}^T)$ --- per una matrice quadrata "full rank" vale $R(\mathbf{A}) = C(\mathbf{A}) \in R^N$

Section 8.5, Row spaces of $\mathbf{A}$ and $\mathbf{A}^T\mathbf{A}$

Esse hanno spazio colonna della stessa dimensione. D'altra parte, $\mathbf{A}$ e $\mathbf{A}^T\mathbf{A}$ generalmente non hanno esattamente le stesse colonne. Pertanto, le colonne di queste due matrici si estendono sullo stesso sottospazio, ma possono avere insiemi di basi diversi.

$\mathbf{A}^T\mathbf{A}$ potrebbe essere una matrice più piccola (ovvero, potrebbe avere meno numeri) e quindi essere computazionalmente più semplice da gestire.

Section 8.6, Null space of a matrix

E' indicato con $N(\mathbf{A})$, è così definito; $\mathbf{Ay} = \mathbf{0}$

Null space of a matrix

$$N(\mathbf{A}) = \{\lambda \mathbf{y} \mid \mathbf{Ay} = \mathbf{0}, \quad \lambda \in \mathbb{R}\} - \{\mathbf{y} = \mathbf{0}\} \qquad (8.10)$$

Section 8.6, code block 8.1

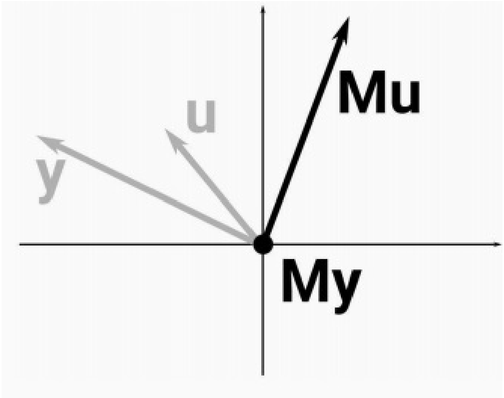
Esempio di calcolo del "null matrix space"

In [2]:
```
A = np.random.randn(3,4)

# the null space
null_space(A)
```

Out[2]:
```
array([[ 0.0926282 ],
       [ 0.06469105],
       [-0.13749914],
       [ 0.98403713]])
```

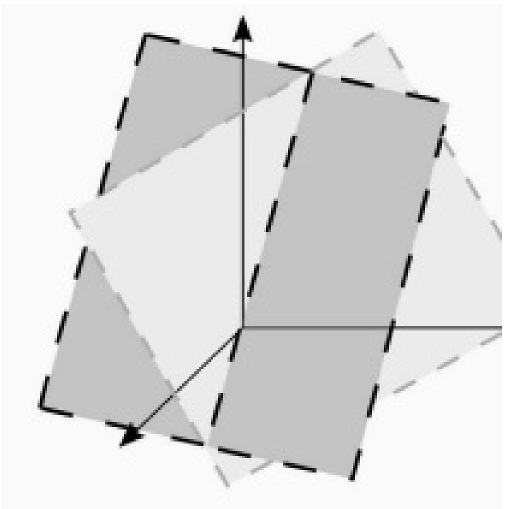
Section 8.7, Geometric interpretation of the null space



y è ortogonale a **M** --> abbiamo "null space", altrimenti no

Section 8.8, Orthogonal subspaces, orthogonal complements 8.8

I quattro sottospazi di una matrice - lo spazio delle colonne, lo spazio delle righe, lo spazio nullo e lo spazio nullo a sinistra - sono ragguppabili in due coppie di complementi ortogonali.



N.B. Nello spazio 3D non possono esistere spazi ortogonali, si potrebbe pensare a due piani perpendicolari, ma essi -- *sempre* -- si intersecano in una retta comune.

Section 8.9, Orthogonalities of the matrix spaces

Orthogonality of the column space and the left-null space, if:

$\mathbf{y} \perp C(\mathbf{A})$ or re-writing in a different form $\mathbf{A}^T\mathbf{y} = 0$

If $C(\mathbf{A})$ spans in R^M --> $N(\mathbf{A}^T)$ must be empty, because only zeros vectors are $\perp$ to the entire ambient space.

In [26]:
```python
from sympy import Matrix
A = [[2, 3, 5], [-4, 2, 3], [0, 0, 0]]
A = Matrix(A)
A * A.nullspace()[0]
```

Out[26]:

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

In [45]:
```python
from sympy import Matrix
B = [[1,1],[2,2]]
B = Matrix(B)
BT = B.T
CB = B.columnspace()
NBT = BT.nullspace()[0]
print(CB,'= C(B)')
print(' ')
print(NBT,'= N(BT)')

D = [[1,1],[2,3]]
D = Matrix(D)
DT = D.T
CD = D.columnspace()
NDT = DT.nullspace()
print(' ')
print(CD,'= C(D)')
print(' ')
print('If more then one column --> zero space must be the null vector')
print(NDT,'= N(DT)')
#print(NBT,'= N(BT)')
```

```
[Matrix([
[1],
[2]])] = C(B)

Matrix([[-2], [1]]) = N(BT)

[Matrix([
[1],
[2]]), Matrix([
[1],
[3]])] = C(D)

If more then one column --> zero space must be the null vector
[] = N(DT)
```

Orthogonality of the row space and the null space

Orthogonality of the row space and the null space, if:

$\mathbf{y} \perp R(\mathbf{A})$ or re-writing in a different form $\mathbf{Ay} = 0$

If $R(\mathbf{A})$ spans in R^M --> $N(\mathbf{A})$ must be empty, because only zeros vectors are $\perp$ to the entire ambient space.

Section 8.10, Dimensionalities of the matrix spaces

The dimensionalities of the column space, the row space, and the two null spaces are all interconnected.

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 3 \\ 0 & 4 & 5 \\ 2 & 0 & 6 \end{bmatrix}, \qquad \begin{gathered} \mathbf{A} \in \mathbb{R}^{3\times 3} \\ rank(\mathbf{A}) = 2 \\ dim(N(\mathbf{A})) = 1 \\ dim(C(\mathbf{A})) = 2 \end{gathered}$$

$$\underbrace{C(\mathbf{A}) \cup N(\mathbf{A}^{\mathrm{T}})}_{\mathbb{R}^{\mathrm{M}}} \qquad \underbrace{R(\mathbf{A}) \cup N(\mathbf{A})}_{\mathbb{R}^{\mathrm{N}}}$$

Subspace dimensionalities

$$dim(C(\mathbf{A})) + dim(N(\mathbf{A}^{\mathrm{T}})) = M \tag{8.25}$$

$$dim(R(\mathbf{A})) + dim(N(\mathbf{A})) = N \tag{8.26}$$

Section 8.11, More on **Ax** = **b** and **Ay** = **0**

Ax = **b** -- si conosce a priori la matrice **A e il vettore b, e l'obiettivo dell'analisi è trovare il vettore x.**

Es. **Xβ** = **y**, dove **X** è chiamata "matrice di progetto", **β** è chiamato "coefficienti di regressione" e **y** è chiamato "dati osservati".

1) l'equazione ha una souzione? Ha una soluzione esatta se **b** = spazio colonna di **A**

2) se **b** non è lo spazio colonna di **A**, quale la soluzione più approssimata rispetto alla soluzione esatta? L'equazione diventa **Ax** = **b**ˆ, dove **x** e **b**ˆ sono selezionati in modo tale che:

(1) **b**ˆ sia nello spazio delle colonne di **A** e **x** siano i coefficienti,

(2) **b**ˆ sia il più vicino possibile all'originale **b**.

Ciò si ottiene attraverso la “soluzione dei minimi quadrati”, che è la spina dorsale della statistica.

Ay = **0**. Siamo interessati a una versione spostata di questa matrice, espressa come (**A** − **λI**)y = **0**. La soluzione di questa equazione (vettore **y**) è chiamata autovettore della matrice e **λ** è il suo autovalore associato.

Gli autovettori rivelano direzioni nella matrice che hanno proprietà speciali, come la robustezza alle trasformazioni geometriche o la massimizzazione della covarianza in un set di dati. In diversi contesti.

La soluzione di **Ay** = **0** è chiamata "Analisi delle Componenti Principali", autocomposizione generalizzata, decomposizione ai valori singolari, analisi discriminante lineare di Fisher,

Queste analisi svolgono un ruolo centrale nelle applicazioni di apprendimento automatico e nell'elaborazione del segnale multivariata.

Section 8.14 CODE CHALLENGES

1. Crea due matrici di numeri casuali 4×4, ciascuna con rango=3 (consulta la sfida del codice del capitolo 7 per come farlo). Chiama quelle matrici **A** e **B**. Quindi trova un vettore nello spazio nullo di **A** (vettore **n**). Infine, mostra che **BAn** è il vettore degli zeri mentre **ABn** non lo è.

In [3]:
```
# reduced-rank matrices
A = np.random.randn(4,3)@np.random.randn(3,4)
B = np.random.randn(4,3)@np.random.randn(3,4)

# null space of A
n = null_space(A)

# zeros vector
print(B@A@n)

# not zeros vector
print(A@B@n)
```

```
[[-4.44089210e-16]
 [ 2.22044605e-16]
 [-5.55111512e-17]
 [ 8.32667268e-17]]
[[-0.96735913]
 [ 0.21520275]
 [-3.70047294]
 [ 2.14360596]]
```

Section 8.14 CODE CHALLENGES

2. Confermare le dimensionalità del subspazio espresse a

pagina 233. Creare una matrice 16×11 con rango=9. Quindi identificare le basi per gli spazi nulli di sinistra e di destra e determinarne le dimensionalità. Confermare che la dimensionalità dello spazio nullo a sinistra più la dimensionalità dello spazio delle colonne è 16 (la dimensionalità dell'ambiente dello spazio delle colonne) e che la dimensionalità dello spazio nullo più la dimensionalità dello spazio delle righe è 11 (la dimensionalità dell'ambiente dello spazio riga).

In [2]:
```
# a reduced-rank matrix
A = np.random.randn(16,9) @ np.random.randn(9,11)

# null spaces
rn = null_space(A)
ln = null_space(A.T)
r  = np.linalg.matrix_rank(A)
r1  = np.linalg.matrix_rank(ln)
print(rn)

# dimensionalities
print(' ')
print('Rango di rn = ',r)
print('Shape di rn = ',rn.shape[1],' da cui 9+2 = 11')
print('Shape di ln = ',ln.shape[1],' da cui 9+7 = 16')
```

```
[[ 0.13931114 -0.08999635]
 [ 0.33726933 -0.14699845]
 [-0.72052699 -0.42983549]
 [ 0.01915707 -0.31366692]
 [ 0.04137513  0.16779997]
 [ 0.42453447 -0.41831105]
 [-0.008615    0.50161473]
 [-0.13907693 -0.0975584 ]
 [ 0.24662175 -0.40794992]
 [ 0.27431119  0.14318513]
 [ 0.09944204 -0.18959315]]

Rango di rn =  9
Shape di rn =  2  da cui 9+2 = 11
Shape di ln =  7  da cui 9+7 = 16
```

In [ ]: