

Apunte Único: Álgebra Lineal Computacional - Práctica 2

Por alumnos de ALC
Facultad de Ciencias Exactas y Naturales
UBA

última actualización 05/04/25 @ 17:54

Choose your destiny:

(doubleclick en el ejercicio para saltar)

☉ [Notas teóricas](#)

☉ Ejercicios de la guía:

1.	5.	9.	13.	17.	21.	25.
2.	6.	10.	14.	18.	22.	
3.	7.	11.	15.	19.	23.	
4.	8.	12.	16.	20.	24.	

☉ Ejercicios de Parciales

🔥??.

Esta Guía 2 que tenés se actualizó por última vez:

05/04/25 @ 17:54

Escaneá el QR para bajarte (quizás) una versión más nueva:

Guía 2



El resto de las guías repo en [github](#) para descargar las guías con los últimos updates.



Si querés mandar un ejercicio o avisar de algún error, lo más fácil es por [Telegram](#).



Notas teóricas:

Transformaciones lineales

✚ Dados V y W dos K -espacios vectoriales, una $f : V \rightarrow W$ es *transformación lineal* si cumple:

- $f(v_1 + v_2) = f(v_1) + f(v_2) \quad \forall v, w \in V$
- $f(\alpha \cdot v_1) = \alpha \cdot f(v_1) \quad \forall \alpha \in K, v \in V$

✚ $f : K^n \rightarrow K^m$ si transformo:

$$f(x_1, \dots, x_n) = f\left(\sum_{k=1}^n x_k \cdot \underbrace{e_k}_{\in K^{n \times 1}}\right) \stackrel{\text{TL}}{=} \sum_{k=1}^n x_k \cdot \underbrace{f(e_k)}_{\in K^{m \times 1}} = \underbrace{\left(f(e_1) \mid \dots \mid f(e_n) \right)}_{A \in K^{m \times n}} \cdot \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \underbrace{A \cdot x}_{\in K^{m \times 1}}$$

✚ *Matriz de una transformación lineal:*

Dados V y W dos K -espacios vectoriales y $f : V \rightarrow W$ una t.l. Sean $B = \{v_1, \dots, v_n\}$ base de V y $B' = \{w_1, \dots, w_m\}$ se llama matriz de la transformación lineal de la base B en la base B' a aquella matriz $[f]_{BB'}$ que satisface:

$$[f]_{BB'}[v]_B = [f(v)]_{B'} \quad \forall v \in V$$

✚ Sea V un K -espacio vectorial y $B = \{v_1, \dots, v_n\}$ base de V . Podemos definir en forma única una t.l. de V en W definiendo cada $f(v_i) \in W$ con $i = 1, \dots, n$.

✚ Sea $A \in K^{m \times n}$, define $f : K^n \rightarrow K^m$. El $\text{Nu}(A) = \{x \in K^n / Ax = 0\}$

✚ Sea $A \in K^{m \times n}$, define $f : K^n \rightarrow K^m$. La $\text{Im}(A) = \{Ax \in K^m \text{ con } x \in K^n\} = \langle c_1(A), \dots, c_n(A) \rangle$. También $\text{rg}(A) = \dim(\text{Im}(A))$

✚ *Propiedades de una transformación lineal:*

Sea $f : V \rightarrow W$ una t.l. y $B = \{v_1, \dots, v_n\}$ un conjunto de generadores de V . Entonces $\{f(v_1), \dots, f(v_n)\}$ es un conjunto generador para la imagen de f .

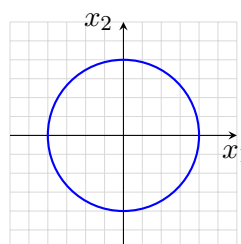
- f se dice *monomorfismo* si es inyectiva. Si f es *mono*, $\dim(\text{Nu}(f)) = 0$
- f se dice *epimorfismo* si es suryectiva. Si f es *epi*, $\dim(\text{Im}(f)) = \dim(W)$
- f se dice *isomorfismo* si es *mono* y *epi*. Si f es iso es inversible.

✚ *Norma* Sea $\|\cdot\| : K^n \rightarrow \mathbb{R} \geq 0$. Entonces $\|\cdot\|$ es norma si cumple:

- 1) $\|x\| \geq 0$ y $\|x\| = 0 \Leftrightarrow x = 0, x \in K^n$
- 2) $\|\alpha x\| = |\alpha| \|x\|$ con $\alpha \in K$ y $x \in K^n$
- 3) $\|x + y\| \leq \|x\| + \|y\|$ con $x, y \in K$

✚ Ejemplos:

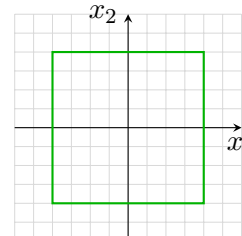
- Norma 2: $\|x\|_2 = \sqrt{\sum_{k=0}^n |x_k|^2} \xrightarrow{\text{por ejemplo}} \|x\|_2 = 1$



• Norma p : $\|x\|_p = \sqrt[p]{\sum_{k=0}^n |x_k|^p}$ $\xrightarrow{\text{por ejemplo}}$ $\|x\|_p = 1$



• Norma ∞ : $\lim_{p \rightarrow \infty} \|x\|_p = \max_{1 \leq i \leq n} |x_i|$ $\xrightarrow{\text{por ejemplo}}$ $\|x\|_\infty = 1$



Aritmética de punto flotante:

✿ Escribir 0.25 en base 10:

Base 10 es obviamente nuestra base favorita:

$$\begin{cases} 0.25 \cdot 10 = 2 + 0.5 \\ 0.5 \cdot 10 = 5 + 0 \\ 0 \cdot 10 = 0 + 0 \end{cases} \rightarrow (0.25)_{10} = (2 \cdot 10^{-1} + 5 \cdot 10^{-2} + 0 \cdot 10^{-3} + 0)_{10} = 0.25$$

Escribir 0.25 en base 2:

$$\begin{cases} 0.25 \cdot 2 = 0 + 0.5 \\ 0.5 \cdot 2 = 1 + 0 \\ 0 \cdot 2 = 0 + 0 \end{cases} \rightarrow (0.25)_2 = (0 \cdot 2^{-1} + 1 \cdot 2^{-2} + 0 \cdot 2^{-3} + 0)_2 = 0.01$$

Escribir 0.3 en base 2:

$$\begin{cases} 0.3 \cdot 2 = 0 + 0.6 \\ 0.6 \cdot 2 = 1 + 0.2 \\ 0.2 \cdot 2 = 0 + 0.4 \\ 0.4 \cdot 2 = 0 + 0.8 \\ 0.8 \cdot 2 = 1 + 0.6 \\ 0.6 \cdot 2 = 1 + 0.2 \\ 0.2 \cdot 2 = 0 + 0.4 \\ 0.4 \cdot 2 = 0 + 0.8 \\ 0.8 \cdot 2 = 1 + 0.6 \\ \vdots = \vdots \end{cases} \rightarrow (0.3)_2 = (0 \cdot 2^{-1} + 1 \cdot 2^{-2} + 0 \cdot 2^{-3} + 0 \cdot 2^{-4} + 1 \cdot 2^{-5} + 1 \cdot 2^{-6} + 0 \cdot 2^{-7} + 0 \cdot 2^{-8} + 1 \cdot 2^{-9} + 1 \cdot 2^{-10} + 0 \cdot 2^{-11} + 0 \cdot 2^{-12} \dots)_2 = 0.01\overline{0011}$$

Para escribir al 0.3 en base 2 voy a necesitar infinitos números en la *mantisa*, la máquina no puede y ahí aparecen los errores de *redondeo* o *truncamiento*.

Errores:

Tengo que un *número de máquina*, número posta que la máquina representa, con la notación *mantisa*, *exponente*:

En base 10 $\rightarrow x = 0, a_1 a_2 a_3 \dots a_m \cdot 10^{exp}$ con $0 \leq a_i \leq 9 (a_1 \neq 0)$

En base 2 $\rightarrow x = 0, a_1 a_2 a_3 \dots a_m \cdot 2^{exp}$ con $0 \leq a_i \leq 1 (a_1 \neq 0)$

Por ejemplo si $m = 3 \Rightarrow x = 0, a_1 a_2 a_3 \cdot 2^{exp}$. Para cada valor de *exp* voy a tener un total de $1 \cdot 2 \cdot 2 = 4$

posibles valores de máquina. La separación entre 2 valores x_1 y x_2 consecutivos es de 2^m , por eso para órdenes grandes la separación entre un número y otro es mayor.

Si el número real, real que quiero es $x = 0.3$, la máquina no puede representarlo de forma exacta. Puedo acotar el error en forma absoluta como:

$$|x - x^*| \leq \frac{1}{2} \frac{1}{2^m} \cdot 2^{exp}$$

Y en forma relativa como:

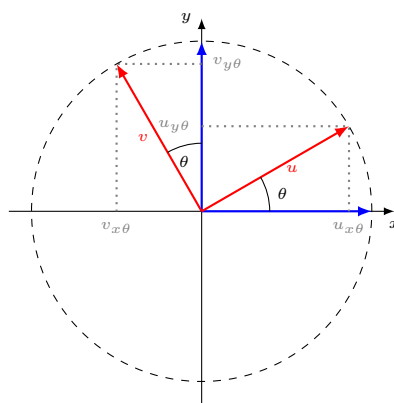
$$\frac{|x - x^*|}{|x|} \leq 5 \cdot 2^{-m}$$

Deducción matriz de rotación 2d (ponele):

Quiero que:

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \cdot \begin{pmatrix} u \\ v \end{pmatrix} = \underbrace{\begin{pmatrix} a \\ c \end{pmatrix}}_{\star^1} \cdot u_0 + \underbrace{\begin{pmatrix} b \\ d \end{pmatrix}}_{\star^2} \cdot v_0 = \begin{pmatrix} u_\theta \\ v_\theta \end{pmatrix}$$

En el gráfico veo lo que quiero lograr.



Entre el gráfico y \star^1 :

$$\begin{pmatrix} a \\ c \end{pmatrix} \cdot u_0 = \begin{pmatrix} u_{x\theta} \\ u_{y\theta} \end{pmatrix} \xrightarrow[\text{sohcatoa}]{!} \begin{pmatrix} u_0 \cdot \cos(\theta) \\ u_0 \cdot \sin(\theta) \end{pmatrix} \Leftrightarrow \begin{pmatrix} a \\ c \end{pmatrix} = \begin{pmatrix} \cos(\theta) \\ \sin(\theta) \end{pmatrix}$$

Entre el gráfico y \star^2 :

$$\begin{pmatrix} b \\ d \end{pmatrix} \cdot v_0 = \begin{pmatrix} v_{x\theta} \\ v_{y\theta} \end{pmatrix} \xrightarrow[\text{sohcatoa}]{!} \begin{pmatrix} -v_0 \cdot \sin(\theta) \\ v_0 \cdot \cos(\theta) \end{pmatrix} \Leftrightarrow \begin{pmatrix} b \\ d \end{pmatrix} = \begin{pmatrix} -\sin(\theta) \\ \cos(\theta) \end{pmatrix}$$

Juntando esos resultados:

$$R_\theta = \begin{pmatrix} a & b \\ c & d \end{pmatrix} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix}$$

Ejercicios de la guía:

Ejercicio 1. 🤖... hay que hacerlo! 🤖

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en LaTeX → [una pull request](#) al 🐙.

Ejercicio 2. 🤖... hay que hacerlo! 🤖

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en LaTeX → [una pull request](#) al 🐙.

Ejercicio 3. 🤖... hay que hacerlo! 🤖

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en LaTeX → [una pull request](#) al 🐙.

Ejercicio 4. 🤖... hay que hacerlo! 🤖

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en LaTeX → [una pull request](#) al 🐙.

Ejercicio 5. 🤖... hay que hacerlo! 🤖

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en LaTeX → [una pull request](#) al 🐙.

Ejercicio 6. 🤖... hay que hacerlo! 🤖

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en LaTeX → [una pull request](#) al 🐙.

Aritmética de punto flotante

Ejercicio 7. Algunos experimentos: Realizar las siguientes operaciones en Python 🐍. En todos los casos, pensar: ¿Cuál es el resultado esperado? ¿Coincide con el obtenido? ¿A qué se debe el problema (si lo hay)? (Notamos ϵ al épsilon de la máquina. Puede obtenerse importando la librería numpy como np y ejecutando el comando `np.finfo(float).eps`).

- | | | |
|--|--|---|
| a) Tomando $p = 1e34, q = 1$, calcular $p + q - p$. | | |
| b) Tomando $p = 100, q = 1e - 15$, calcular $(p + q) + q$ y $((p + q) + q) + q$. Comparar con $p + 2q$ y con $p + 3q$ respectivamente. | | |
| c) $0.1 + 0.2 == 0.3$ | g) $\frac{\epsilon}{2}$ | k) $(1 + (\frac{\epsilon}{2} + \frac{\epsilon}{2})) - 1$ |
| d) $0.1 + 0.3 == 0.4$ | h) $(1 + \frac{\epsilon}{2}) + \frac{\epsilon}{2}$ | l) $\sin(10^j \pi)$ para $1 \leq j \leq 25$. |
| e) $1e - 323$ | i) $1 + (\frac{\epsilon}{2} + \frac{\epsilon}{2})$ | |
| f) $1e - 324$ | j) $((1 + \frac{\epsilon}{2}) + \frac{\epsilon}{2}) - 1$ | m) $\sin(\frac{\pi}{2} + \pi 10^j)$ para $1 \leq j \leq 25$. |
-

- a) El epsilon sería el número más chico tal que:

$$1 + \epsilon \neq 1$$

Comparando:

$$\begin{aligned}
 p &= 100 = 0.1 \cdot 10^3 \\
 q &= 0.000\,000\,000\,000\,000\,001 \cdot 10^3 \\
 2q &= 0.000\,000\,000\,000\,000\,002 \cdot 10^3 \\
 3q &= 0.000\,000\,000\,000\,000\,003 \cdot 10^3 \\
 p + 2q &= 0.100\,000\,000\,000\,000\,000\,02 \cdot 10^3 = 0.100\,000\,000\,000\,000\,000\,0 \cdot 10^3 = 100 \\
 &\quad \begin{array}{c} \text{16}^{\text{to}} \text{ decimal} \\ \uparrow \\ \text{fue un placer} \end{array} \\
 p + 3q &= 0.100\,000\,000\,000\,000\,000\,03 \cdot 10^3 = 0.100\,000\,000\,000\,000\,000\,0 \cdot 10^3 = 100 \\
 &\quad \begin{array}{c} \text{16}^{\text{to}} \text{ decimal} \\ \uparrow \\ \text{fue un placer} \end{array}
 \end{aligned}$$

```
import numpy as np

epsilon = np.finfo(float).eps

print(f"epsilon = {epsilon}")    # epsilon = 2.220446049250313e-16

p = 100
q = 1e-15

calculo1 = (p + q) + q
calculo2 = ((p + q) + q) + q
calculo3 = p + 2*q
calculo4 = p + 3*q

print(f"p = {p} \n q = {q}")
print(f"(p + q) + q = {calculo1}")
print(f"((p + q) + q) + q = {calculo2}")
print(f"p + 2q = {calculo3}")
print(f"p + 3q = {calculo4}")
```

c) ... hay que hacerlo! 🚫

Si querés mandá la solución → [al grupo de Telegram](#) 🗨️, o mejor aún si querés subirlo en L^AT_EX → [una pull request](#) al 🐙.

d) ... hay que hacerlo! 🚫

Si querés mandá la solución → [al grupo de Telegram](#) 🗨️, o mejor aún si querés subirlo en L^AT_EX → [una pull request](#) al 🐙.

e) ... hay que hacerlo! 🚫

Si querés mandá la solución → [al grupo de Telegram](#) 🗨️, o mejor aún si querés subirlo en L^AT_EX → [una pull request](#) al 🐙.

f) ... hay que hacerlo! 🚫

Si querés mandá la solución → [al grupo de Telegram](#) 🗨️, o mejor aún si querés subirlo en L^AT_EX → [una pull request](#) al 🐙.

g) ... hay que hacerlo! 🚫

Si querés mandá la solución → [al grupo de Telegram](#) 🗨️, o mejor aún si querés subirlo en L^AT_EX → [una pull request](#) al 🐙.

h) ... hay que hacerlo! 🚫

Si querés mandá la solución → [al grupo de Telegram](#) 🗨️, o mejor aún si querés subirlo en L^AT_EX → [una pull request](#) al 🐙.

i) 🐞... hay que hacerlo! 🐞

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en \LaTeX → *una pull request* al 🐙.

j) 🐞... hay que hacerlo! 🐞

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en \LaTeX → *una pull request* al 🐙.

k) 🐞... hay que hacerlo! 🐞

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en \LaTeX → *una pull request* al 🐙.

l) 🐞... hay que hacerlo! 🐞

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en \LaTeX → *una pull request* al 🐙.

m) 🐞... hay que hacerlo! 🐞

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en \LaTeX → *una pull request* al 🐙.

Dale las gracias y un poco de amor 🧡 a los que contribuyeron! Gracias por tu aporte:
👉 naD GarRaz 🐙

Ejercicio 8. 🐞... hay que hacerlo! 🐞

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en \LaTeX → *una pull request* al 🐙.

Ejercicio 9. 🐞... hay que hacerlo! 🐞

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en \LaTeX → *una pull request* al 🐙.

Ejercicio 10. 🐞... hay que hacerlo! 🐞

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en \LaTeX → *una pull request* al 🐙.

Ejercicio 11. 🐞... hay que hacerlo! 🐞

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en \LaTeX → *una pull request* al 🐙.

Ejercicio 12. 🐞... hay que hacerlo! 🐞

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en \LaTeX → *una pull request* al 🐙.

Ejercicio 13. 🐞... hay que hacerlo! 🐞

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en \LaTeX → *una pull request* al 🐙.

Ejercicio 14. 🐞... hay que hacerlo! 🐞

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en \LaTeX → *una pull request* al 🐙.

Ejercicio 15. 🐞... hay que hacerlo! 🐞

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en \LaTeX → *una pull request* al 🐙.

Ejercicio 16. 🤖... hay que hacerlo! 🤖

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en $\text{IAT}_{\text{E}}\text{X}$ → [una pull request](#) al 🐙.

Ejercicio 17. 🤖... hay que hacerlo! 🤖

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en $\text{IAT}_{\text{E}}\text{X}$ → [una pull request](#) al 🐙.

Ejercicio 18. 🤖... hay que hacerlo! 🤖

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en $\text{IAT}_{\text{E}}\text{X}$ → [una pull request](#) al 🐙.

Ejercicio 19. 🤖... hay que hacerlo! 🤖

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en $\text{IAT}_{\text{E}}\text{X}$ → [una pull request](#) al 🐙.

Ejercicio 20. 🤖... hay que hacerlo! 🤖

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en $\text{IAT}_{\text{E}}\text{X}$ → [una pull request](#) al 🐙.

Ejercicio 21. 🤖... hay que hacerlo! 🤖

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en $\text{IAT}_{\text{E}}\text{X}$ → [una pull request](#) al 🐙.

Ejercicio 22. 🤖... hay que hacerlo! 🤖

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en $\text{IAT}_{\text{E}}\text{X}$ → [una pull request](#) al 🐙.

Ejercicio 23. 🤖... hay que hacerlo! 🤖

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en $\text{IAT}_{\text{E}}\text{X}$ → [una pull request](#) al 🐙.

Ejercicio 24. 🤖... hay que hacerlo! 🤖

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en $\text{IAT}_{\text{E}}\text{X}$ → [una pull request](#) al 🐙.

Ejercicio 25. 🤖... hay que hacerlo! 🤖

Si querés mandá la solución → [al grupo de Telegram](#) 🗉, o mejor aún si querés subirlo en $\text{IAT}_{\text{E}}\text{X}$ → [una pull request](#) al 🐙.

Ejercicios de parciales: