

T2.0 Recordatorio de algebra lineal (opcional)

Índice

1 Notación

- 1.1 Vectores
- 1.2 Matrices
- 1.3 Tensores

2 Espacios vectoriales

- 2.1 Suma y escalado de vectores
- 2.2 Independencia lineal, span y base
- 2.3 Transformaciones lineales y matrices
- 2.4 Rango y espacio nulo de una matriz
- 2.5 Proyección lineal

3 Normas vectoriales y matriciales

- 3.1 Normas vectoriales
- 3.2 Normas matriciales

4 Propiedades de una matriz

- 4.1 Traza de una matriz cuadrada
- 4.2 Determinante de una matriz cuadrada
- 4.3 Rango de una matriz
- 4.4 Números de condición

5 Tipos de matrices especiales

- 5.1 Matriz diagonal
- 5.2 Matrices triangulares
- 5.3 Matrices definidas positivas
- 5.4 Matrices ortogonales

6 Multiplicación de matrices

- 6.1 Definición y propiedades
- 6.2 Productos vector-vector
 - 6.2.1 Producto escalar, interno o punto
 - 6.2.2 Producto vectorial, externo o cruz
- 6.3 Productos matriz-vector
 - 6.3.1 Interpretación por filas
 - 6.3.2 Interpretación por columnas
- 6.4 Productos matriz-matriz
 - 6.4.1 Interpretaciones
 - 6.4.2 Notación

7 Inversa de una matriz cuadrada

8 Conceptos básicos de cálculo matricial

- 8.1 Derivadas
- 8.2 Gradientes
- 8.3 La Jacobiana
 - 8.3.1 Multiplicación de Jacobianas y vectores
 - 8.3.2 Jacobiana de una composición
- 8.4 La Hessiana

9 Gradientes de funciones comunes

- 9.1 Gradientes de funciones de escalar a escalar
- 9.2 Gradientes de funciones de vector a escalar
- 9.3 Gradientes de funciones de matriz a escalar

1 Notación

1.1 Vectores

Vector $\mathbf{x} \in \mathbb{R}^n$: suponemos que es vector columna

$$\mathbf{x} = (x_1, x_2, \dots, x_n)^t$$

Vector de todo unos y todo ceros: $\mathbf{1}$ $\mathbf{0}$

Vector unitario \mathbf{e}_i o **one-hot**: todo ceros salvo un uno en la posición i

$$\mathbf{e}_i = (0, \dots, 0, 1, 0, \dots, 0)^t$$

1.2 Matrices

Matriz $\mathbf{A} \in \mathbb{R}^{m \times n}$: **cuadrada** si $m = n$

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$$

Entrada de la i -ésima fila y j -ésima columna: A_{ij} o $A_{i,j}$

Fila i -ésima y columna j -ésima: $\mathbf{A}_{i,:}$ y $\mathbf{A}_{:,j}$

Matriz vista por filas: $\mathbf{A} = [\mathbf{A}_{1,:}; \mathbf{A}_{2,:}; \dots; \mathbf{A}_{n,:}]$

Matriz vista por columnas: $\mathbf{A} = [\mathbf{A}_{:,1}, \mathbf{A}_{:,2}, \dots, \mathbf{A}_{:,n}]$

Transpuesta de $\mathbf{A} \in \mathbb{R}^{m \times n}$: $\mathbf{A}^t \in \mathbb{R}^{n \times m}$ tal que $(\mathbf{A}^t)_{ij} = A_{ji}$, cumple

$$\begin{aligned} (\mathbf{A}^t)^t &= \mathbf{A} \\ (\mathbf{AB})^t &= \mathbf{B}^t \mathbf{A}^t \\ (\mathbf{A} + \mathbf{B})^t &= \mathbf{A}^t + \mathbf{B}^t \end{aligned}$$

Matriz simétrica: matriz cuadrada \mathbf{A} tal que $\mathbf{A} = \mathbf{A}^t$; \mathbb{S}^n denota todas las matrices simétricas de talla n

Reformato de matriz a vector: $\text{vec}(\mathbf{A}) = [\mathbf{A}_{:,1}; \mathbf{A}_{:,2}; \dots; \mathbf{A}_{:,n}] \in \mathbb{R}^{mn \times 1}$

Reformato de vector a matriz: puede ser por filas (python, C++) o por columnas (julia, octave, R, fortran)

1.3 Tensores

Tensor: generalización de matriz a tres o más dimensiones

Orden o rango: dimensiones (concepto distinto en matrices)

Ejemplo: vector 8-dimensional reformatado como matriz y tensor

```
In [1]: import numpy as np
a = np.arange(8)
print(a, a.shape)
a2d = a.reshape([2, -1])
print(a2d, a2d.shape)
a3d = a.reshape([2, 2, -1])
print(a3d, a3d.shape)
```

```
[0 1 2 3 4 5 6 7] (8,)
```

```
[[0 1 2 3]
```

```
 [4 5 6 7]] (2, 4)
```

```
[[[0 1]
```

```
 [2 3]]
```

```
[[[4 5]
```

```
 [6 7]]] (2, 2, 2)
```

2 Espacios vectoriales

2.1 Suma y escalado de vectores

Espacio vectorial: conjunto de **vectores** que pueden sumarse entre ellos y escalarse (multiplicarse) por **escalares**

2.2 Independencia lineal, span y base

Independencia lineal: $\{x_1, \dots, x_n\}$ es **linealmente independiente** si ninguno puede representarse como combinación lineal del resto

Span de $\{x_1, \dots, x_n\}$: $\text{span}(\{x_1, \dots, x_n\}) = \{v : v = \sum_{i=1}^n \alpha_i x_i, \alpha_i \in \mathbb{R}\}$

Base \mathcal{B} : $\text{span}(\mathcal{B}) = \mathbb{R}^n$

Base estándar o canónica: $\mathcal{B} = \{e_1 = (1, 0, \dots, 0)^t, \dots, e_n = (0, 0, \dots, 1)^t\}$

2.3 Transformaciones lineales y matrices

Transformación lineal: $f : \mathcal{V} \rightarrow \mathcal{W}$ tal que $f(\mathbf{v} + \mathbf{w}) = f(\mathbf{v}) + f(\mathbf{w})$ y $f(a\mathbf{v}) = af(\mathbf{v})$ para todo $\mathbf{v}, \mathbf{w} \in \mathcal{V}$

Representación matricial: $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ con $\mathbf{A} \in \mathbb{R}^{m \times n}$ tal que $\mathbf{y} = \mathbf{A}\mathbf{x}$

2.4 Rango y espacio nulo de una matriz

Rango, espacio columna o imagen de $\mathbf{A}^{m \times n}$: $\text{range}(\mathbf{A}) = \{\mathbf{v} \in \mathbb{R}^m : \mathbf{v} = \mathbf{A}\mathbf{x}, \mathbf{x} \in \mathbb{R}^n\}$

Espacio nulo o núcleo de $\mathbf{A}^{m \times n}$: $\text{nullspace}(\mathbf{A}) = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{A}\mathbf{x} = \mathbf{0}\}$

2.5 Proyección lineal

Proyección de $\mathbf{y} \in \mathbb{R}^m$ en el span de $\{\mathbf{x}_1, \dots, \mathbf{x}_n \in \mathbb{R}^m\}$:

$$\text{Proj}(\mathbf{y}; \{\mathbf{x}_1, \dots, \mathbf{x}_n\}) = \underset{\mathbf{v} \in \text{span}(\{\mathbf{x}_1, \dots, \mathbf{x}_n\})}{\text{argmin}} \|\mathbf{y} - \mathbf{v}\|_2$$

Proyección de $\mathbf{y} \in \mathbb{R}^m$ en el span de $\mathbf{A} \in \mathbb{R}^{m \times n}$:

$$\text{Proj}(\mathbf{y}; \mathbf{A}) = \underset{\mathbf{v} \in \text{range}(\mathbf{A})}{\text{argmin}} \|\mathbf{y} - \mathbf{v}\|_2 = \text{Proj}(\mathbf{A}) \mathbf{y} \quad \text{con} \quad \text{Proj}(\mathbf{A}) = \mathbf{A}(\mathbf{A}^t \mathbf{A})^{-1} \mathbf{A}^t$$

3 Normas vectoriales y matriciales

3.1 Normas vectoriales

Norma de un vector \mathbf{x} : $\|\mathbf{x}\|$ es cualquier $f : \mathbb{R}^n \rightarrow \mathbb{R}$ tal que

$$f(\mathbf{x}) \geq 0 \quad f(\mathbf{x}) = 0 \Leftrightarrow \mathbf{x} = \mathbf{0} \quad f(t\mathbf{x}) = |t|f(\mathbf{x}) \quad f(\mathbf{x} + \mathbf{y}) \leq f(\mathbf{x}) + f(\mathbf{y})$$

p-norma o L_p : $\|\mathbf{x}\|_p = \left(\sum_{i=1}^n |x_i|^p\right)^{1/p} \quad p \in \mathbb{R}^{\geq 1}$

2-norma, Euclídea o L_2 : $\|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^n x_i^2} = \sqrt{\mathbf{x}^t \mathbf{x}}$

1-norma, taxicab, Manhattan o L_1 : $\|\mathbf{x}\|_1 = \sum_{i=1}^n |x_i|$

Max-norma, infinito o L_∞ : $\|\mathbf{x}\|_\infty = \max_i |x_i|$

0-pseudo-norma: $\|\mathbf{x}\|_0 = \sum_{i=1}^n \mathbb{I}(|x_i| > 0) \stackrel{0^0=0}{=} \sum_{i=1}^n x_i^0$

```
In [1]: import numpy as np
a = np.array([-3, 4])
for p in (2, 1, np.inf, 0):
    print(f'L{p}({a}) = {np.linalg.norm(a, p)}')
```

```
L2([-3  4]) = 5.0
L1([-3  4]) = 7.0
Linf([-3  4]) = 4.0
L0([-3  4]) = 2.0
```

3.2 Normas matriciales

Suponemos $\mathbf{A} \in \mathbb{R}^{m \times n}$ y la interpretamos como función lineal $f(\mathbf{x}) = \mathbf{A}\mathbf{x}$

Norma inducida por L_p : $\|\mathbf{A}\|_p = \max_{\mathbf{x} \neq \mathbf{0}} \frac{\|\mathbf{A}\mathbf{x}\|_p}{\|\mathbf{x}\|_p} = \max_{\|\mathbf{x}\|_p=1} \|\mathbf{A}\mathbf{x}\|_p$ (máxima ganancia en la dirección de \mathbf{x})

Norma inducida por L_2 : $\|\mathbf{A}\|_2 = \sqrt{\lambda_{\max}(\mathbf{A}^t \mathbf{A})} = \sigma_{\max}(\mathbf{A})$ (máximo valor singular de \mathbf{A})

Norma nuclear o traza: $\|\mathbf{A}\|_* = \text{tr}(\sqrt{\mathbf{A}^t \mathbf{A}}) = \sum_i \sigma_i = \sum_i |\sigma_i| = \|\boldsymbol{\sigma}\|_1$

p-norma de Schatten: $\|\mathbf{A}\|_p = (\sum_i \sigma_i^p(\mathbf{A}))^{1/p}$ $p \in \mathbb{R}^{\geq 1}$

Norma vectorial: $\|\mathbf{A}\| = \|\text{vec}(\mathbf{A})\|$

Norma de Frobenius: $\|\mathbf{A}\|_F = \|\text{vec}(\mathbf{A})\|_2 = \sqrt{\sum_{i=1}^m \sum_{j=1}^n a_{ij}^2} = \sqrt{\text{tr}(\mathbf{A}^t \mathbf{A})}$

```
In [ ]: import numpy as np
A = np.array([[1, 0], [0, 1]])
for p in (2, 1, np.inf, 'fro'):
    print(f'p = {p}: {np.linalg.norm(A, p)}')
```

```
p = 2: 1.0
p = 1: 1.0
p = inf: 1.0
p = fro: 1.4142135623730951
```

4 Propiedades de una matriz

4.1 Traza de una matriz cuadrada

Traza de $\mathbf{A} \in \mathbb{R}^{n \times n}$: $\text{tr}(\mathbf{A}) = \sum_{i=1}^n A_{ii}$

Propiedades: $c \in \mathbb{R}$ $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n \times n}$

$$\text{tr}(\mathbf{A}) = \text{tr}(\mathbf{A}^t)$$

$$\text{tr}(\mathbf{A} + \mathbf{B}) = \text{tr}(\mathbf{A}) + \text{tr}(\mathbf{B})$$

$$\text{tr}(c\mathbf{A}) = c \text{tr}(\mathbf{A})$$

$$\text{tr}(\mathbf{AB}) = \text{tr}(\mathbf{BA})$$

$$\text{tr}(\mathbf{A}) = \sum_{i=1}^n \lambda_i \quad (\text{suma de valores propios})$$

Propiedad de la permutación cíclica: $\text{tr}(\mathbf{ABC}) = \text{tr}(\mathbf{BCA}) = \text{tr}(\mathbf{CAB})$ \mathbf{ABC} cuadrada

Truco de la traza: $\mathbf{x}^t \mathbf{A} \mathbf{x} = \text{tr}(\mathbf{x}^t \mathbf{A} \mathbf{x}) = \text{tr}(\mathbf{x} \mathbf{x}^t \mathbf{A})$

Estimador traza de Hutchinson: aproximación Monte Carlo a $\text{tr}(\mathbf{A})$ con vectores aleatorios \mathbf{v} tal que $\mathbb{E}[\mathbf{v}\mathbf{v}^t] = \mathbf{I}$

$$\text{tr}(\mathbf{A}) = \text{tr}(\mathbf{A} \mathbb{E}[\mathbf{v}\mathbf{v}^t]) = \mathbb{E}[\text{tr}(\mathbf{A} \mathbf{v}\mathbf{v}^t)] = \mathbb{E}[\text{tr}(\mathbf{v}^t \mathbf{A} \mathbf{v})]$$

```
In [1]: import numpy as np
A = np.array([[2, 1], [1, 2]])
print(np.trace(A))
```

4

4.2 Determinante de una matriz cuadrada

Determinante de $\mathbf{A} \in \mathbb{R}^{n \times n}$: $|\mathbf{A}|$ mide cuánto cambia un volumen unitario viendo \mathbf{A} como una transformación lineal

Propiedades: $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n \times n}$

$$\begin{aligned} |\mathbf{A}| &= |\mathbf{A}^t| \\ |c\mathbf{A}| &= c^n |\mathbf{A}| \\ |\mathbf{AB}| &= |\mathbf{A}||\mathbf{B}| \\ |\mathbf{A}| &= 0 \quad \text{sii} \quad \mathbf{A} \text{ singular} \\ |\mathbf{A}^{-1}| &= \frac{1}{|\mathbf{A}|} \quad \text{si} \quad \mathbf{A} \text{ es no singular} \\ |\mathbf{A}| &= \prod_{i=1}^n \lambda_i \quad \{\lambda_i\} \text{ valores propios de } \mathbf{A} \end{aligned}$$

Si \mathbf{A} es definida positiva, su **descomposición de Cholesky** es $\mathbf{A} = \mathbf{L}\mathbf{L}^t$ con \mathbf{L} triangular inferior y:

$$\begin{aligned} |\mathbf{A}| &= |\mathbf{L}||\mathbf{L}^t| = |\mathbf{L}|^2 = \left(\prod_i L_{ii} \right)^2 \\ \log|\mathbf{A}| &= 2 \log|\mathbf{L}| = 2 \log \prod_i L_{ii} = 2 \operatorname{tr}(\log(\operatorname{diag}(\mathbf{L}))) \end{aligned}$$

```
In [2]: import numpy as np
A = np.array([[2, 1], [1, 2]])
print(round(np.linalg.det(A)))
```

3

4.3 Rango de una matriz

Rango de $\mathbf{A} \in \mathbb{R}^{m \times n}$: $\text{rank}(\mathbf{A})$ es la dimensión del espacio generado por las columnas o filas de \mathbf{A} (pues coincide para toda \mathbf{A})

Propiedades: $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{m \times n}$, $\mathbf{C} \in \mathbb{R}^{n \times p}$

$$\text{rank}(\mathbf{A}) \leq \min(m, n)$$

$$\text{rank}(\mathbf{A}) = \text{rank}(\mathbf{A}^t) = \text{rank}(\mathbf{A}^t \mathbf{A}) = \text{rank}(\mathbf{A} \mathbf{A}^t)$$

$$\text{rank}(\mathbf{A} \mathbf{C}) \leq \min(\text{rank}(\mathbf{A}), \text{rank}(\mathbf{C}))$$

$$\text{rank}(\mathbf{A} + \mathbf{B}) \leq \text{rank}(\mathbf{A}) + \text{rank}(\mathbf{B})$$

\mathbf{A} es de rango completo: si $\text{rank}(\mathbf{A}) = \min(m, n)$; si no, es de **rango deficiente**

Invertibilidad: \mathbf{A} es **invertible** sii es de rango completo

```
In [3]: import numpy as np
A = np.array([[1, 0, 1], [-2, -3, 1], [3, 3, 0]])
print(np.linalg.matrix_rank(A))
```

2

4.4 Números de condición

Número de condición de \mathbf{A} : $\kappa(\mathbf{A})$ mide la inestabilidad numérica de los cálculos con \mathbf{A} (mejor que el determinante)

$$\kappa(\mathbf{A}) = \|\mathbf{A}\| \|\mathbf{A}^{-1}\| \quad \kappa(\mathbf{A}) \geq 1$$

\mathbf{A} bien condicionada: si $\kappa(\mathbf{A}) \approx 1$; si no, \mathbf{A} está **mal condicionada** y, si $\kappa(\mathbf{A})$ es grande, \mathbf{A} es **casi-singular**

Número de condición con L2: $\kappa(\mathbf{A}) = \frac{\sigma_{\max}}{\sigma_{\min}} = \sqrt{\frac{\lambda_{\max}}{\lambda_{\min}}}$ σ valor singular, λ valor propio

```
In [4]: import numpy as np
A = 0.1 * np.eye(100)
print(np.linalg.cond(A), np.linalg.det(A))
```

```
1.0 1.0000000000000033e-100
```

5 Tipos de matrices especiales

5.1 Matriz diagonal

Matriz diagonal: $\mathbf{D} = \text{diag}(d_1, \dots, d_n) \in \mathbb{R}^{n \times n}$ con ceros fuera de la diagonal

Construcción y extracción de una matriz diagonal: $\mathbf{D} = \text{diag}(\mathbf{d})$ y $\mathbf{d} = \text{diag}(\mathbf{D})$

Matriz identidad: $\mathbf{I} = \text{diag}(1, \dots, 1) \in \mathbb{R}^{n \times n}$ con unos en diagonal y ceros fuera

Matriz diagonal por bloques: $\begin{pmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{B} \end{pmatrix}$

Matriz banda-diagonal: con elementos no nulos en una banda de ancho k centrada en la diagonal;
tridiagonal si $k = 3$

```
In [1]: import numpy as np
print("Construcción:\n", np.diag([1, 2]))
print("Extracción: ", np.diag(np.array([[1, 0], [0, 2]])))
print("Identidad:\n", np.eye(2))
```

Construcción:

```
[[1 0]
 [0 2]]
```

Extracción: [1 2]

Identidad:

```
[[1. 0.]
 [0. 1.]]
```

5.2 Matrices triangulares

Matriz triangular superior: solo tiene elementos no nulos en la diagonal y por encima

Matriz triangular inferior: solo tiene elementos no nulos en la diagonal y por debajo

Propiedades: valores propios en la diagonal, por lo que $|\mathbf{A}| = \prod_i A_{ii}$

```
In [2]: import numpy as np
A = np.arange(1, 5).reshape((2, 2))
print(A, "\n", np.tril(A), "\n", np.triu(A))
```

```
[[1 2]
 [3 4]]
[[1 0]
 [3 4]]
[[1 2]
 [0 4]]
```


5.3 Matrices definidas positivas

Forma cuadrática asociada a una matriz real y simétrica, $\mathbf{A} \in \mathbb{R}^{n \times n}$:

$$f(\mathbf{x}) = \mathbf{x}^t \mathbf{A} \mathbf{x} = \sum_{i=1}^n \sum_{j=1}^n A_{ij} x_i x_j$$

A definida positiva ($\mathbf{A} \succ 0$ o $\mathbf{A} > 0$) sii $f(\mathbf{x}) > 0$ para todo \mathbf{x} no nulo

A semi-definida positiva ($\mathbf{A} \succeq 0$ o $\mathbf{A} \geq 0$) sii $f(\mathbf{x}) \geq 0$ para todo \mathbf{x} no nulo

A definida negativa ($\mathbf{A} \prec 0$ o $\mathbf{A} < 0$) sii $f(\mathbf{x}) < 0$ para todo \mathbf{x} no nulo

A semi-definida negativa ($\mathbf{A} \preceq 0$ o $\mathbf{A} \leq 0$) sii $f(\mathbf{x}) \leq 0$ para todo \mathbf{x} no nulo

A indefinida si no es semi-definida positiva ni semi-definida negativa

Relación entre A y -A: $\mathbf{A} \succ 0$ sii $-\mathbf{A} \prec 0$ $\mathbf{A} \succeq 0$ sii $-\mathbf{A} \preceq 0$ \mathbf{A} indefinida sii $-\mathbf{A}$ indefinida

Condición suficiente para $\mathbf{A} \succ 0$: \mathbf{A} diagonalmente dominante, $|a_{ii}| > \sum_{j \neq i} |a_{ij}|$ para todo i , y diagonal positiva

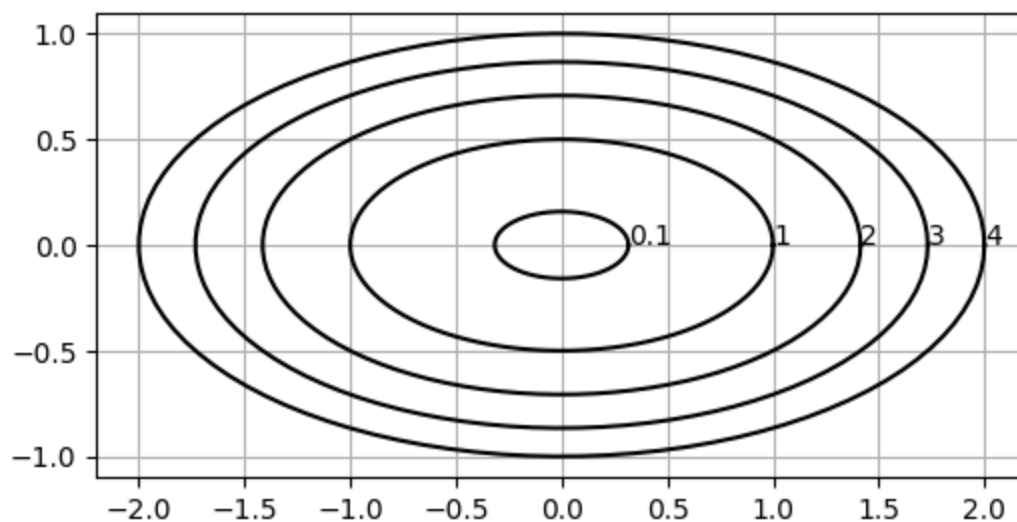
Condición necesaria y suficiente para $\mathbf{A} \succ 0$: valores propios positivos; por ejemplo, si $\mathbf{A} = \text{diag}(\lambda_1, \dots, \lambda_n)$ con $\{\lambda_i > 0\}$

$$f(\mathbf{x}) = \mathbf{x}^t \mathbf{A} \mathbf{x} = \sum_{i=1}^n \lambda_i x_i^2 > 0$$

Matriz de Gram de $\mathbf{A} \in \mathbb{R}^{m \times n}$: $\mathbf{G} = \mathbf{A}^t \mathbf{A}$ en general semi-definida positiva; definida positiva si $m \geq n$ y de rango completo

Ejemplo: si $\mathbf{A} = \text{diag}(\lambda_1, \lambda_2) = \text{diag}(1, 4)$, el conjunto de nivel C_k para f , $C_k = \{\mathbf{x} : f(\mathbf{x}) = k\}$, es una elipse centrada en el origen de semiejes alineados con los ejes canónicos y longitudes $\{\sqrt{k/\lambda_i}\}$

```
In [5]: import numpy as np; import matplotlib.pyplot as plt
A = np.diag([1, 4]); t = np.linspace(0, 2.0 * np.pi, 100)
circ = np.array([np.cos(t), np.sin(t)]).T
fig, ax = plt.subplots(figsize=(6, 6))
ax.set(aspect='equal'); ax.grid()
for k in (.1, 1, 2, 3, 4):
    semiaxes = np.sqrt(k * np.linalg.inv(A)); C = circ @ semiaxes # C @ A @ C.T = k
    plt.plot(C[:, 0], C[:, 1], color='black'); plt.annotate(k, xy=(C[0]))
```



5.4 Matrices ortogonales

Vectores ortogonales: $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ son ortogonales si $\mathbf{x}^t \mathbf{y} = 0$

Vector normalizado: $\mathbf{x} \in \mathbb{R}^n$ está normalizado si $\|\mathbf{x}\|_2 = 1$

Conjunto de vectores ortonormal: si son ortogonales dos a dos y normalizados

Matriz ortogonal $\mathbf{U} \in \mathbb{R}^{n \times n}$: si el conjunto de sus columnas (o filas) es ortonormal

Caracterización de matriz ortogonal: \mathbf{U} ortogonal sii $\mathbf{U}^t \mathbf{U} = \mathbf{I} = \mathbf{U} \mathbf{U}^t$ sii $\mathbf{U}^t = \mathbf{U}^{-1}$

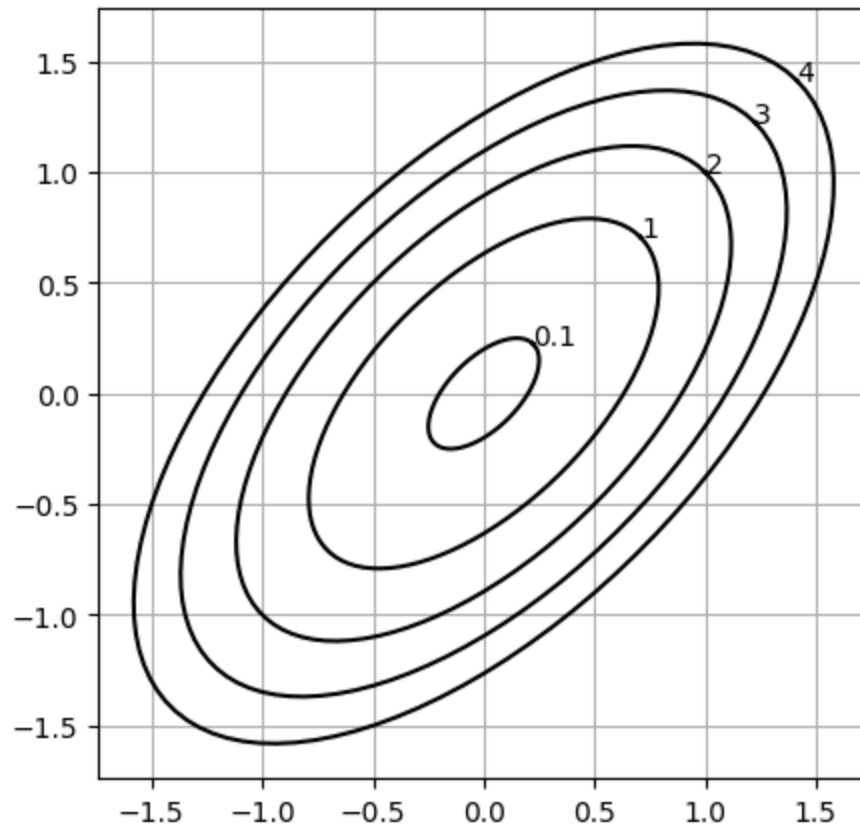
Propiedades de las transformaciones ortogonales: preservan longitudes y ángulos

$$\begin{aligned} \|\mathbf{U}\mathbf{x}\|_2 &= \|\mathbf{x}\|_2 \\ \cos(\alpha(\mathbf{U}\mathbf{x}, \mathbf{U}\mathbf{y})) &= \frac{(\mathbf{U}\mathbf{x})^t (\mathbf{U}\mathbf{y})}{\|\mathbf{U}\mathbf{x}\| \|\mathbf{U}\mathbf{y}\|} = \frac{\mathbf{x}^t \mathbf{y}}{\|\mathbf{x}\| \|\mathbf{y}\|} = \cos(\alpha(\mathbf{x}, \mathbf{y})) \end{aligned}$$

Ejemplo (cont.): matriz ortogonal de rotación 2d de α radianes en sentido antihorario,

$$\mathbf{R}(\alpha) = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix}$$

```
In [6]: import numpy as np; import matplotlib.pyplot as plt
A = np.diag([1, 4]); t = np.linspace(0, 2.0 * np.pi, 100)
circ = np.array([np.cos(t), np.sin(t)]).T; alpha = np.pi / 4
R = np.array([ [np.cos(alpha), -np.sin(alpha)], [np.sin(alpha), np.cos(alpha)] ])
fig, ax = plt.subplots(figsize=(5, 5))
ax.set(aspect='equal'); ax.grid()
for k in (.1, 1, 2, 3, 4):
    semiaxes = np.sqrt(k * np.linalg.inv(A)); C = circ @ semiaxes # C @ A @ C.T = k
    C = C @ R.T; plt.plot(C[:, 0], C[:, 1], color='black'); plt.annotate(k, xy=(C[0]))
```



6 Multiplicación de matrices

6.1 Definición y propiedades

Producto de dos matrices: $\mathbf{A} \in \mathbb{R}^{m \times n}$ y $\mathbf{B} \in \mathbb{R}^{n \times p}$

$$\mathbf{C} = \mathbf{AB} \in \mathbb{R}^{m \times p} \quad \text{con} \quad C_{ij} = \sum_{k=1}^n A_{ik} B_{kj}$$

Coste: $O(mnp)$, aunque mucho menor con GPUs

Propiedad asociativa: $(\mathbf{AB})\mathbf{C} = \mathbf{A}(\mathbf{BC})$

Propiedad distributiva: $\mathbf{A}(\mathbf{B} + \mathbf{C}) = \mathbf{AB} + \mathbf{AC}$

No propiedad conmutativa: $\mathbf{AB} \neq \mathbf{BA}$

```
In [1]: import numpy as np
A = np.array([[0, 1], [1, 2]]); B = A; A @ B
```

```
Out[1]: array([[1, 2],
               [2, 5]])
```

6.2 Productos vector-vector

6.2.1 Producto escalar, interno o punto

$$\mathbf{x}, \mathbf{y} \in \mathbb{R}^n : \quad \langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^t \mathbf{y} = \sum_{i=1}^n x_i y_i = \mathbf{y}^t \mathbf{x}$$

```
In [1]: import numpy as np
x = np.array([1, 2]); y = np.array([2, 1])
print(np.dot(x, y))
```

4

6.2.2 Producto vectorial, externo o cruz

$$\mathbf{x} \in \mathbb{R}^m, \mathbf{y} \in \mathbb{R}^n : \quad \mathbf{x} \mathbf{y}^t = \begin{bmatrix} x_1 y_1 & x_1 y_2 & \cdots & x_1 y_n \\ x_2 y_1 & x_2 y_2 & \cdots & x_2 y_n \\ \vdots & \vdots & \ddots & \vdots \\ x_m y_1 & x_m y_2 & \cdots & x_m y_n \end{bmatrix} \in \mathbb{R}^{m \times n}$$

```
In [2]: import numpy as np
x = np.array([1, 2]); y = np.array([2, 1])
print(np.outer(x, y));
```

```
[[2 1]
 [4 2]]
```

6.3 Productos matriz-vector

Producto matriz-vector:

$$\mathbf{A} \in \mathbb{R}^{m \times n}, \mathbf{x} \in \mathbb{R}^n : \quad \mathbf{y} = \mathbf{A}\mathbf{x} \in \mathbb{R}^m$$

6.3.1 Interpretación por filas

La i -ésima entrada de \mathbf{y} es el producto escalar de la i -ésima fila de \mathbf{A} y \mathbf{x} :

$$\mathbf{y} = \mathbf{A}\mathbf{x} = \begin{bmatrix} \mathbf{a}_1^t \\ \mathbf{a}_2^t \\ \vdots \\ \mathbf{a}_m^t \end{bmatrix} \mathbf{x} = \begin{bmatrix} \mathbf{a}_1^t \mathbf{x} \\ \mathbf{a}_2^t \mathbf{x} \\ \vdots \\ \mathbf{a}_m^t \mathbf{x} \end{bmatrix}$$

6.3.2 Interpretación por columnas

\mathbf{y} es una combinación lineal de las columnas de \mathbf{A} , las cuales pueden verse como una **base** de un **subespacio lineal**:

$$\mathbf{y} = \mathbf{A}\mathbf{x} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n] \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \mathbf{a}_1 x_1 + \mathbf{a}_2 x_2 + \dots + \mathbf{a}_n x_n$$

6.4 Productos matriz-matriz

6.4.1 Interpretaciones

Dadas $\mathbf{A} \in \mathbb{R}^{m \times n}$ y $\mathbf{B} \in \mathbb{R}^{n \times p}$, existen cuatro maneras diferentes de interpretar $\mathbf{C} = \mathbf{AB} \in \mathbb{R}^{m \times p}$:

$$\mathbf{C} = \begin{bmatrix} \mathbf{a}_1^t \\ \mathbf{a}_2^t \\ \vdots \\ \mathbf{a}_m^t \end{bmatrix} [\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_p] = \begin{bmatrix} \mathbf{a}_1^t \mathbf{b}_1 & \mathbf{a}_1^t \mathbf{b}_2 & \cdots & \mathbf{a}_1^t \mathbf{b}_p \\ \mathbf{a}_2^t \mathbf{b}_1 & \mathbf{a}_2^t \mathbf{b}_2 & \cdots & \mathbf{a}_2^t \mathbf{b}_p \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{a}_m^t \mathbf{b}_1 & \mathbf{a}_m^t \mathbf{b}_2 & \cdots & \mathbf{a}_m^t \mathbf{b}_p \end{bmatrix}$$

$$\mathbf{C} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n] \begin{bmatrix} \mathbf{b}_1^t \\ \mathbf{b}_2^t \\ \vdots \\ \mathbf{b}_n^t \end{bmatrix} = \sum_{i=1}^n \mathbf{a}_i \mathbf{b}_i^t$$

$$\mathbf{C} = \mathbf{A} [\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_p] = [\mathbf{Ab}_1, \mathbf{Ab}_2, \dots, \mathbf{Ab}_p]$$

$$\mathbf{C} = \begin{bmatrix} \mathbf{a}_1^t \\ \mathbf{a}_2^t \\ \vdots \\ \mathbf{a}_m^t \end{bmatrix} \mathbf{B} = \begin{bmatrix} \mathbf{a}_1^t \mathbf{B} \\ \mathbf{a}_2^t \mathbf{B} \\ \vdots \\ \mathbf{a}_m^t \mathbf{B} \end{bmatrix}$$

6.4.2 Notación

Cuadrado matricial: $\mathbf{A}^2 = \mathbf{A}\mathbf{A}$

Cuadrado elemental: $\mathbf{A}^{\odot 2} = [A_{ij}^2]$

Raíz cuadrada matricial: $\mathbf{A} = \sqrt{\mathbf{M}}$ si $\mathbf{A}^2 = \mathbf{M}$

Raíz cuadrada elemental: $[\sqrt{M_{ij}}]$

7 Inversa de una matriz cuadrada

Inversa de $\mathbf{A} \in \mathbb{R}^{n \times n}$: matriz \mathbf{A}^{-1} tal que $\mathbf{A}^{-1}\mathbf{A} = \mathbf{I} = \mathbf{A}\mathbf{A}^{-1}$

Existencia: \mathbf{A}^{-1} existe sii $\det \mathbf{A} \neq 0$; esto es, si \mathbf{A} es no singular

Propiedades: $(\mathbf{A}^{-1})^{-1} = \mathbf{A}$ $(\mathbf{AB})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}$ $(\mathbf{A}^{-1})^t = (\mathbf{A}^t)^{-1} = \mathbf{A}^{-t}$

Inversa de $\mathbf{A} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$: $\mathbf{A}^{-1} = \frac{1}{|\mathbf{A}|} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$

Inversa de una diagonal por bloques: $\begin{pmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{B} \end{pmatrix}^{-1} = \begin{pmatrix} \mathbf{A}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{B}^{-1} \end{pmatrix}$

8 Conceptos básicos de cálculo matricial

8.1 Derivadas

Derivada: de una función $f : \mathbb{R} \rightarrow \mathbb{R}$ en un punto

$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}$$

Interpretación geométrica: $f'(x)$ puede verse como la pendiente de la línea tangente en $f(x)$

$$f(x+h) \approx f(x) + f'(x)h$$

Aproximaciones a la derivada de diferencia finita: dado un $h > 0$ pequeño

$$f'(x) \approx \frac{f(x+h) - f(x)}{h} \quad (\text{diferencia forward})$$

$$f'(x) \approx \frac{f(x+h/2) - f(x-h/2)}{h} \quad (\text{diferencia central})$$

$$f'(x) \approx \frac{f(x) - f(x-h)}{h} \quad (\text{diferencia backward})$$

La diferenciación es un operador funcional: $D(f) = f'$

Notación de Lagrange: f' denota la derivada; f'' la segunda; $f^{(n)}$ la n -ésima

Notación de Leibinz: si $y = f(x)$ es una función, $\frac{dy}{dx}$ o $\frac{d}{dx}f(x)$ es su derivada y $\left. \frac{df}{dx} \right|_{x=a}$ su derivada en un punto a

8.2 Gradientes

Derivada parcial: de $f : \mathbb{R}^n \rightarrow \mathbb{R}$ respecto a x_i

$$\frac{\partial f}{\partial x_i} = \lim_{h \rightarrow 0} \frac{f(\mathbf{x} + h\mathbf{e}_i) - f(\mathbf{x})}{h} \quad (\mathbf{e}_i \text{ es el } i\text{-ésimo vector unitario})$$

Gradiente: de f en un punto \mathbf{x}

$$\mathbf{g} = \frac{\partial f}{\partial \mathbf{x}} = \nabla f = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{pmatrix} \quad \text{o} \quad \mathbf{g}(\mathbf{x}^*) = \left. \frac{\partial f}{\partial \mathbf{x}} \right|_{\mathbf{x}^*} \quad (\text{enfatisa que se evalúa en } \mathbf{x}^*)$$

8.3 La Jacobiana

Matriz Jacobiana: de $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$

$$\mathbf{J}_f(\mathbf{x}) = \frac{\partial \mathbf{f}}{\partial \mathbf{x}^t} = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{pmatrix} = \begin{pmatrix} \nabla f_1(\mathbf{x})^t \\ \vdots \\ \nabla f_m(\mathbf{x})^t \end{pmatrix}$$

8.3.1 Multiplicación de Jacobianas y vectores

Producto Jacobiana-vector: $\mathbf{J} \in \mathbb{R}^{m \times n}$, $\mathbf{v} \in \mathbb{R}^n$

$$\mathbf{J}_f(\mathbf{x})\mathbf{v} = \begin{pmatrix} \nabla f_1(\mathbf{x})^t \\ \vdots \\ \nabla f_m(\mathbf{x})^t \end{pmatrix} \mathbf{v} = \begin{pmatrix} \nabla f_1(\mathbf{x})^t \mathbf{v} \\ \vdots \\ \nabla f_m(\mathbf{x})^t \mathbf{v} \end{pmatrix}$$

Producto vector-Jacobiana: $\mathbf{J} \in \mathbb{R}^{m \times n}$, $\mathbf{u} \in \mathbb{R}^m$

$$\mathbf{u}^t \mathbf{J}_f(\mathbf{x}) = \mathbf{u}^t \left(\frac{\partial \mathbf{f}}{\partial x_1}, \dots, \frac{\partial \mathbf{f}}{\partial x_n} \right) = \left(\mathbf{u} \frac{\partial \mathbf{f}}{\partial x_1}, \dots, \mathbf{u} \frac{\partial \mathbf{f}}{\partial x_n} \right)$$

8.3.2 Jacobiana de una composición

Jacobiana de una composición: $h(\mathbf{x}) = g(f(\mathbf{x}))$

$$\mathbf{J}_h(\mathbf{x}) = \mathbf{J}_g(f(\mathbf{x}))\mathbf{J}_f(\mathbf{x})$$

Ejemplo: $f : \mathbb{R} \rightarrow \mathbb{R}^2, g : \mathbb{R}^2 \rightarrow \mathbb{R}^2$

$$\frac{\partial \mathbf{g}}{\partial x} = \begin{pmatrix} \frac{\partial}{\partial x} g_1(f_1(x), f_2(x)) \\ \frac{\partial}{\partial x} g_2(f_1(x), f_2(x)) \end{pmatrix} = \begin{pmatrix} \frac{\partial g_1}{\partial f_1} \frac{\partial f_1}{\partial x} + \frac{\partial g_1}{\partial f_2} \frac{\partial f_2}{\partial x} \\ \frac{\partial g_2}{\partial f_1} \frac{\partial f_1}{\partial x} + \frac{\partial g_2}{\partial f_2} \frac{\partial f_2}{\partial x} \end{pmatrix} = \begin{pmatrix} \frac{\partial g_1}{\partial f_1} & \frac{\partial g_1}{\partial f_2} \\ \frac{\partial g_2}{\partial f_1} & \frac{\partial g_2}{\partial f_2} \end{pmatrix} \begin{pmatrix} \frac{\partial f_1}{\partial x} \\ \frac{\partial f_2}{\partial x} \end{pmatrix} = \frac{\partial \mathbf{g}}{\partial \mathbf{f}} \frac{\partial \mathbf{f}}{\partial x}$$

8.4 La Hessiana

Matriz Hessiana: de $f : \mathbb{R}^n \rightarrow \mathbb{R}$ es la Jacobiana del gradiente

$$\mathbf{H}_f = \frac{\partial^2 f}{\partial \mathbf{x}^2} = \nabla^2 f = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{pmatrix}$$

9 Gradientes de funciones comunes

9.1 Gradientes de funciones de escalar a escalar

Función diferenciable: $f : \mathbb{R} \rightarrow \mathbb{R}$

$$\frac{d}{dx} cx^n = cnx^{n-1}$$

$$\frac{d}{dx} \log(x) = 1/x$$

$$\frac{d}{dx} e^x = e^x$$

$$\frac{d}{dx} [f(x) + g(x)] = \frac{df(x)}{dx} + \frac{dg(x)}{dx}$$

$$\frac{d}{dx} [f(x)g(x)] = f(x) \frac{dg(x)}{dx} + g(x) \frac{df(x)}{dx}$$

$$\frac{d}{dx} f(u(x)) = \frac{df(u)}{du} \frac{du}{dx}$$

9.2 Gradientes de funciones de vector a escalar

Función diferenciable: $f : \mathbb{R}^n \rightarrow \mathbb{R}$

$$\frac{\partial}{\partial \mathbf{x}} (\mathbf{a}^t \mathbf{x}) = \mathbf{a}$$

$$\frac{\partial}{\partial \mathbf{x}} (\mathbf{b}^t \mathbf{A} \mathbf{x}) = \mathbf{A}^t \mathbf{b}$$

$$\frac{\partial}{\partial \mathbf{x}} (\mathbf{x}^t \mathbf{A} \mathbf{x}) = (\mathbf{A} + \mathbf{A}^t) \mathbf{x}$$

9.3 Gradientes de funciones de matriz a escalar

Función diferenciable: $f : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}$

$$\frac{\partial f}{\partial \mathbf{X}} = \begin{pmatrix} \frac{\partial f}{\partial x_{11}} & \cdots & \frac{\partial f}{\partial x_{1n}} \\ \vdots & \ddots & \vdots \\ \frac{\partial f}{\partial x_{m1}} & \cdots & \frac{\partial f}{\partial x_{mn}} \end{pmatrix}$$

Identidades para formas cuadráticas:

$$\frac{\partial}{\partial \mathbf{X}} (\mathbf{a}^t \mathbf{X} \mathbf{b}) = \mathbf{a} \mathbf{b}^t \quad \frac{\partial}{\partial \mathbf{X}} (\mathbf{a}^t \mathbf{X}^t \mathbf{b}) = \mathbf{b} \mathbf{a}^t$$

Identidades con la traza:

$$\begin{aligned} \frac{\partial}{\partial \mathbf{X}} \text{tr}(\mathbf{A} \mathbf{X} \mathbf{B}) &= \mathbf{A}^t \mathbf{B}^t \\ \frac{\partial}{\partial \mathbf{X}} \text{tr}(\mathbf{X}^t \mathbf{A}) &= \mathbf{A} \\ \frac{\partial}{\partial \mathbf{X}} \text{tr}(\mathbf{X}^{-1} \mathbf{A}) &= -\mathbf{X}^{-t} \mathbf{A}^t \mathbf{X}^{-t} \\ \frac{\partial}{\partial \mathbf{X}} \text{tr}(\mathbf{X}^t \mathbf{A} \mathbf{X}) &= (\mathbf{A} + \mathbf{A}^t) \mathbf{X} \end{aligned}$$

Identidades con el determinante:

$$\begin{aligned} \frac{\partial}{\partial \mathbf{X}} |\mathbf{A} \mathbf{X} \mathbf{B}| &= |\mathbf{A} \mathbf{X} \mathbf{B}| \mathbf{X}^{-t} \\ \frac{\partial}{\partial \mathbf{X}} \ln(|\mathbf{X}|) &= \mathbf{X}^{-t} \end{aligned}$$