

Moviendo la primera fila al final ya tenemos la forma escalonada, por tanto $\text{rg}(A) = 2$ y se deduce $\dim \text{Ker}(f) = 1$, $\dim \text{Im}(f) = 2$. Se cumple $A\vec{c} = \vec{0}$ con $\vec{c} = (1, -2, 0)^t$ y estas son las coordenadas de $1 - 2x + 0x^2$, entonces $\{1 - 2x\}$ es una base de $\text{Ker}(f)$. Las columnas pivote son la primera y la tercera, entonces una base de $\text{Im}(f)$ está formada por los vectores (polinomios) cuyas coordenadas son $(0, 6, 0)^t$ y $(0, 2, 2)^t$, es decir, $\{6x, 2x + 2x^2\}$. Evidentemente la podemos “simplificar” a $\{x, x + x^2\}$. También sería válida $\{x, x^2\}$, pues genera lo mismo. Por la Proposición 2.3.3, f no es ni inyectiva ni sobreyectiva.

Exprimiendo el silicio [opcional]. A pesar de que esta sección hay un concepto nuevo importante, el de aplicación lineal, los cálculos son similares a los que ya hemos hecho y por tanto no hay demasiado que decir aquí.

Con `matlab/octave` solo recordaremos que si capturamos la salida de `rref` con dos variables, la segunda será asignada con la lista de los números de las columnas pivote. Así en el ejemplo de la aplicación $f : \mathbb{R}^4 \rightarrow \mathbb{R}^4$ produciríamos una base de $\text{Im}(f)$ con las líneas:

```
1 % Matriz A
2 A = [4,2,-1,3; 2,1,0,1; 1,0,3,-2; 9,2,-1,8];
3 [E, p] = rref(A);
4 % Una base de Im(f) es:
5 A(:,p)
```

y lo mismo funcionaría con cualquier otra matriz. La sintaxis de `matlab/octave` permite escribir `A(:,p)` en la última línea para indicar que las filas son arbitrarias y las columnas las que corresponden a los índices de `p`.

Con `sagemath` ya habíamos usado `right_kernel` y ahora encontramos sentido al nombre (por cierto `left_kernel` es esencialmente el núcleo de la traspuesta). Para la imagen se emplea `column_space` que es coherente con la Proposición 2.3.1. De esta forma, el ejemplo de la aplicación $f : \mathbb{R}^4 \rightarrow \mathbb{R}^4$ se resolvería con:

```
1 # Matriz A
2 A = matrix(QQ,4,4,[4,2,-1,3, 2,1,0,1, 1,0,3,-2, 9,2,-1,8])
3 # Núcleo
4 nuc = A.right_kernel()
5 # Imagen
6 ima = A.column_space()
7 # Dimensiones
8 print('dim_Ker =', nuc.dimension(), 'dim_Im =', ima.dimension())
9 # Base del núcleo
10 print(nuc.basis())
11 # Base de la imagen
12 print(ima.basis())
```

La base que produce de $\text{Im}(f)$ es más bonita que la nuestra.

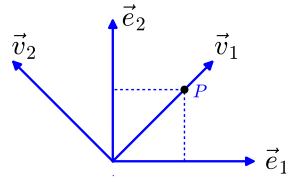
2.4. Cambio de base

Corriendo el riesgo de propiciar el fenómeno psicológico de la profecía autocumplida, diré que el tema de esta sección es uno de los que más confunde a los estudiantes de álgebra lineal. La buena noticia es que, al menos a mi juicio, la confusión desapa-

rece cuando en vez de empeñarnos en memorizar una fórmula nos limitamos a tener en mente una idea sencilla.

La motivación del cambio de base es muy natural. Ya habíamos incidido sobre ello diciendo que una base es un sistema de medidas que especificamos, siendo las medidas los valores de las coordenadas en la base escogida, y a veces es conveniente pasar de un sistema de medidas a otro. Si me sitúo de pie en el origen mirando a la derecha, encontraré lógico expresar los puntos o vectores con $a(1, 0)^t + b(0, 1)^t$, a unidades a la derecha y b hacia arriba, las coordenadas cartesianas habituales. Pero si giro la cabeza 45° en sentido positivo (antihorario) mi nueva derecha es lo que mi yo sin la cabeza girada llamaría $(1/\sqrt{2}, 1/\sqrt{2})^t$ y mi nuevo arriba lo que llamaría $(-1/\sqrt{2}, 1/\sqrt{2})^t$.

$$\begin{aligned} \vec{e}_1 &= \begin{pmatrix} 1 \\ 0 \end{pmatrix} & \vec{e}_2 &= \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ \vec{v}_1 &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} & \vec{v}_2 &= \frac{1}{\sqrt{2}} \begin{pmatrix} -1 \\ 1 \end{pmatrix} \end{aligned}$$



Así tenemos la base antigua, la canónica $\mathcal{B} = \{\vec{e}_1, \vec{e}_2\}$ y la base nueva $\mathcal{B}' = \{\vec{v}_1, \vec{v}_2\}$ con los vectores indicados. Si antes veía un pájaro en $(1/2, 1/2)^t$, ahora estará en $(1/\sqrt{2}, 0)^t$ como es fácil de comprobar en un dibujo o algebraicamente con la identidad $\frac{1}{2}\vec{e}_1 + \frac{1}{2}\vec{e}_2 = \frac{1}{\sqrt{2}}\vec{v}_1 + 0\vec{v}_2$. En realidad todo esto, de manera más o menos implícita lo sabíamos ya, la novedad es que vamos a examinar la fórmula general de un cambio de base y sobre todo vamos a ver cómo afecta a la matriz de una aplicación lineal $f : V \rightarrow V$. Por ejemplo, si el pájaro cae hacia el eje X reduciendo su altura a la mitad en cada unidad de tiempo, la primera base nos llevaría a considerar la aplicación lineal $(x_1, x_2)^t \mapsto (x_1, x_2/2)^t$ que representa cada una de las reducciones de altura pero ¿cómo las veríamos si hemos girado la cabeza?

Respecto a la fórmula general del cambio de base, si queremos hallar las coordenadas en la base \mathcal{B}' de un vector genérico $\vec{x} = (x_1, x_2)^t = x_1\vec{e}_1 + x_2\vec{e}_2$, debemos resolver $(x_1, x_2)^t = x'_1\vec{v}_1 + x'_2\vec{v}_2$ en x'_1 y x'_2 . Esta ecuación se escribe como

$$(2.3) \quad \vec{x} = C\vec{x}' \quad \text{con} \quad C = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} \quad \text{donde} \quad \begin{cases} \vec{x} = \text{coordenadas en } \mathcal{B}, \\ \vec{x}' = \text{coordenadas en } \mathcal{B}'. \end{cases}$$

En general, sean cuales sean \mathcal{B} y \mathcal{B}' , en dimensión finita siempre tendremos el esquema (2.4)

$$\vec{x} = C\vec{x}' \quad \text{con} \quad \begin{cases} \vec{x} = \text{coord. en } \mathcal{B} \\ \vec{x}' = \text{coord. en } \mathcal{B}' \end{cases} \quad \text{colum. de } C = \begin{cases} \text{coord. de los vectores} \\ \text{de } \mathcal{B}' \text{ en la base } \mathcal{B}. \end{cases}$$

La idea que no debes olvidar es que el cambio de base se lleva a cabo multiplicando por una matriz, la cual tiene que ser cuadrada por la Proposición 2.2.2 y de hecho invertible ya que cambiando de \mathcal{B} a \mathcal{B}' y después de \mathcal{B}' a \mathcal{B} recuperaremos las coordenadas iniciales.

Las aplicaciones lineales $f : V \rightarrow W$ con $W = V$ tienen una mayor relevancia y reciben el nombre de *endomorfismos*, son las que actúan en los vectores de un espacio sin salirse de él. Por ejemplo, para $V = \mathbb{R}^2$, un giro alrededor del origen es un endomorfismo y también lo es la aplicación que envía $(x_1, x_2)^t$ a su sombra $(x_1, 0)^t$ en el eje X . Si $\dim V = n < \infty$, una vez fijada una base \mathcal{B} , las coordenadas conforman un vector de K^n y de acuerdo con la Proposición 2.3.2, f queda representada mediante la multiplicación por una matriz: en términos de las coordenadas la función es $\vec{y} = \vec{y}(\vec{x})$ dada por $\vec{y} = A\vec{x}$. ¿Cómo hallar la matriz A' que da la relación $\vec{y}' = A'\vec{x}'$ correspondiente a la base \mathcal{B}' ? Simplemente sustituyendo la relación (2.4) entre las coordenadas en \mathcal{B} y \mathcal{B}' :

$$\vec{y} = A\vec{x} \quad \begin{array}{l} \vec{x} = C\vec{x}' \\ \vec{y} = C\vec{y}' \end{array} \quad \vec{y}' = C^{-1}AC\vec{x}'.$$

Es decir, $A' = C^{-1}AC$. El cambio de base de la matriz de un endomorfismo es por tanto un sencillo cambio de variable, ¿dónde está la dificultad? Si miras tu libro favorito de álgebra lineal quizá leas allí $A' = CAC^{-1}$ porque parece más natural cambiar en (2.4) de lo antiguo a lo nuevo (¿evolución?) o habrá una fórmula que involucra memorizar notaciones más o menos arbitrarias. Mi consejo es que en vez de tratar de recordar la fórmula de un libro o de estas notas, pienses en términos de cambios de variable.

Como primer ejemplo, resolvamos el problema antes mencionado de averiguar en qué se transformaría $(x_1, x_2)^t \mapsto (x_1, x_2/2)^t$ cuando usamos la base $\mathcal{B}' = \{\vec{v}_1, \vec{v}_2\}$ con $\vec{v}_1 = \frac{1}{\sqrt{2}}(1, 1)^t$, $\vec{v}_2 = \frac{1}{\sqrt{2}}(-1, 1)^t$. Tenemos que la aplicación lineal en la base canónica \mathcal{B} actúa como

$$\vec{x} \in \mathbb{R}^2 \mapsto \vec{y} = A\vec{x} \in \mathbb{R}^2 \quad \text{con} \quad A = \begin{pmatrix} 1 & 0 \\ 0 & 1/2 \end{pmatrix}.$$

Esta A es la matriz correspondiente a $y_1 = x_1$, $y_2 = x_2/2$. Por (2.3) tenemos

$$\frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} \vec{y}' = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 \\ 0 & 1/2 \end{pmatrix} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} \vec{x}' = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 \\ 1/2 & 1/2 \end{pmatrix} \vec{x}'.$$

Así pues

$$\vec{y}' = \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}^{-1} \begin{pmatrix} 1 & -1 \\ 1/2 & 1/2 \end{pmatrix} \vec{x}' = \frac{1}{2} \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} 1 & -1 \\ 1/2 & 1/2 \end{pmatrix} \vec{x}' = \begin{pmatrix} 3/4 & -1/4 \\ -1/4 & 3/4 \end{pmatrix} \vec{x}'.$$

Esta última matriz es la que buscábamos del endomorfismo cuando consideramos la base \mathcal{B}' . Es decir, al girar la cabeza 45° el endomorfismo lo observamos como $(x'_1, x'_2)^t \mapsto \frac{1}{4}(3x'_1 - x'_2, -x'_1 + 3x'_2)^t$.

En un problema como el del cambio de base que involucra el producto de tres matrices vamos ahora con un ejemplo en el que el propio endomorfismo está definido mediante un producto de tres matrices. Todo un reto para comprobar que uno tiene las ideas claras.

Consideramos en $V = \{X \in \mathcal{M}_2(\mathbb{R}) : X = X^t\}$, el endomorfismo definido por

$$f(X) = TXT \quad \text{con} \quad T = \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix}$$

y las bases de V

$$\mathcal{B} = \left\{ \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \right\} \quad \text{y} \quad \mathcal{B}' = \left\{ \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}, \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} -1 & 0 \\ 0 & 0 \end{pmatrix} \right\}.$$

Queremos calcular la matriz de f en la base \mathcal{B} y después cambiarla a la base \mathcal{B}' . Según la Proposición 2.3.2 para la primera tarea hay que poner en columna las coordenadas de las imágenes de los elementos de \mathcal{B} , estas son:

$$(2.5) \quad \begin{cases} f(B_1) = \begin{pmatrix} 4 & 2 \\ 2 & 1 \end{pmatrix} = 4B_1 + 2B_2 + B_3 \longrightarrow (4, 2, 1)^t, \\ f(B_2) = \begin{pmatrix} 4 & 3 \\ 3 & 2 \end{pmatrix} = 4B_1 + 3B_2 + 2B_3 \longrightarrow (4, 3, 2)^t, \\ f(B_3) = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} = B_1 + B_2 + B_3 \longrightarrow (1, 1, 1)^t, \end{cases}$$

donde B_j son los elementos de \mathcal{B} . Además los de \mathcal{B}' son $B_1 + B_2$, $2B_1 + B_3$, $-B_1$. Por tanto la matriz A de f en la base \mathcal{B} y la matriz C que pasa coordenadas en \mathcal{B}' a coordenadas en \mathcal{B} son:

$$A = \begin{pmatrix} 4 & 4 & 1 \\ 2 & 3 & 1 \\ 1 & 2 & 1 \end{pmatrix} \quad \text{y} \quad C = \begin{pmatrix} 1 & 2 & -1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}.$$

Seguramente ayuda a comprender que C lleva a cabo el cambio $\vec{x} = C\vec{x}'$ pensar en que la primera matriz de \mathcal{B}' es $(1, 0, 0)^t$ en su propia base y multiplicado por C debe dar $(1, 1, 0)^t$ que son las coordenadas en \mathcal{B} y entonces este vector conformará la primera columna. Lo mismo se aplica a las otras dos columnas.

La inversa de C es muy sencilla de calcular por reducción de Gauss-Jordan. Partiendo de $(C|I_3)$ poniendo la primera fila al final

$$\xrightarrow{f_3 \rightarrow f_3 - f_1} \left(\begin{array}{ccc|ccc} 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 2 & -1 & 1 & -1 & 0 \end{array} \right) \xrightarrow{f_3 \rightarrow f_3 - 2f_2} \left(\begin{array}{ccc|ccc} 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 1 & -1 & -2 \end{array} \right)$$

y basta cambiar de signo la última fila para obtener la inversa en la mitad derecha. La relación $\vec{y} = A\vec{x}$ equivale a $C\vec{y}' = AC\vec{x}'$ y así la matriz en la base \mathcal{B}' es

$$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & 1 & 2 \end{pmatrix} \begin{pmatrix} 4 & 4 & 1 \\ 2 & 3 & 1 \\ 1 & 2 & 1 \end{pmatrix} \begin{pmatrix} 1 & 2 & -1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 5 & 5 & -2 \\ 3 & 3 & -1 \\ 3 & 2 & 0 \end{pmatrix}.$$

En problemas que conlleven cierto número de cálculos siempre es conveniente disponer de una comprobación, aunque sea parcial, del resultado. Si nuestros cálculos son correctos, según la Proposición 2.3.2 aplicada a la primera columna se debe tener

$$T \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} T = 5 \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} + 3 \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} + 3 \begin{pmatrix} -1 & 0 \\ 0 & 0 \end{pmatrix}$$

y unas sencillas cuentas muestran que es así.

Un comentario final es que resulta asequible (y esto está relacionado con que C sea fácil de invertir) llevar a cabo cálculos como los de (2.5) pero con los elementos de la base \mathcal{B}' lo que nos conduciría a la matriz final sin necesidad de pasar por la matriz en la base \mathcal{B} pero justamente el propósito del ejemplo era practicar con el cambio de base.

Exprimiendo el silicio [opcional]. El siguiente código para matlab/octave lleva a cabo los cálculos del cambio de base en el último ejemplo.

```

1 % Matriz T
2 T = [2,1; 1,1];
3 % Matrices de la base B
4 B1 = [1,0; 0,0];
5 B2 = [0,1; 1,0];
6 B3 = [0,0; 0,1];
7 % Imágenes concatenadas en una matriz
8 fB = [T*B1*T, T*B2*T, T*B3*T];
9 % Matriz de zeros (vacía)
10 A = zeros(3);
11 for j = 1:3
12     % Cada submatriz empieza en la columna 2j-1
13     % y las coordenadas en la base B son los
14     % valores de a_11, a_12 y a_22
15     coord = [fB(1,2*j-1); fB(1,2*j)]; fB(2,2*j)];
16     % Rellena la columna j de A
17     A(:,j) = coord;
18 end
19 % Matriz de cambio de base
20 C = [1,2,-1; 1,0,0; 0,1,0];
21 % Matriz del endomorfismo cambiada de base
22 C^-1*A*C

```

Lo único reseñable es el bucle para extraer las coordenadas como en (2.5). Estas se almacenan en el vector `coord` y se asignan a la columna j con `A(:,j) = coord`.

Aunque no tenga que ver mucho con el álgebra lineal, para el que tenga curiosidad aquí está el código en `sagemath` que utilicé para hacer la figura de esta sección:

```

1 # Punto inicial fantasma (invisible, hace espacio)
2 P = point([(0,-0.1)], size=0)
3
4 # Bases
5 B = [(1,0), (0,1)]
6 Bp = [(1/sqrt(2),1/sqrt(2)), (-1/sqrt(2),1/sqrt(2))]
7 for k in range(2):
8     P += arrow( (0,0), B[k], arrowsize=8, width=5)
9     P += text('␣$\\vec{e}_'+str(k+1)+'$', B[k], fontsize=60,
10             ↪ horizontal_alignment='left', color='black')
11     P += arrow( (0,0), Bp[k], arrowsize=8, width=5)
12     P += text('␣$\\vec{v}_'+str(k+1)+'$', Bp[k],
13             ↪ vertical_alignment='bottom', fontsize=60,
14             ↪ color='black')
15
16 # Punto

```

```
14 P += point([(1/2,1/2)], size=300, color='black', zorder=100)
15 P += line([(1/2,0),(1/2,1/2),(0,1/2)], thickness=3, linestyle='--')
16 P += text('$P$',(0.6,0.48), fontsize=40)
17
18 P.set_aspect_ratio(1)
19 P.axes(False)
20 show(P)
```

Cualquiera puede obtener algo similar en un instante con *software* de diseño gráfico. La única ventaja de tenerlo en un programa es la posibilidad de hacer mil y una variaciones ajustando números.