Universidad de Jaén
Departamento de Matemáticas
(Área de Álgebra)

Núcleo e imagen de una aplicación lineal. Clasificación.

En esta práctica se pretende utilizar el cálculo de la expresión matricial de una aplicación lineal respecto de las bases del dominio y codominio de dicha aplicación, para la obtención del núcleo y la imagen, y, a partir de ellos, clasificar los morfismos de espacios vectoriales en epimorfismos, monomorfismos o isomorfismos de una forma más sencilla.

1.- NÚCLEO E IMAGEN DE UNA APLICACIÓN LINEAL.

Dados V y V' dos espacios vectoriales sobre $\mathbb{K}$ y f:V $\longrightarrow$ V' una aplicación lineal, se definen el **núcleo de f** como el subespacio de V dado por:

$$\text{Ker}(f) = \{x \in V \,/\, f(x) = 0\}$$

y se define la **imagen de f** como el subespacio de V' dado por:

$$\text{Im}\,(f) = \{f(x) \,/\, x \in V\}.$$

Un primer método para calcular el núcleo y la imagen es a partir de la definición. Si conocemos la expresión matricial de f, es decir, f(x) = Ax, para calcular el núcleo basta con resolver el sistema homogéneo f(x) = 0, es decir, Ax = 0. Si el sistema anterior es S.C.D. entonces Ker(f)={0} y si es un S.C.I. entonces Ker(f) = L({u_1, u_2,..., u_r}) siendo {u_1, u_2,..., u_r}una base del subespacio vectorial de soluciones del sistema Ax = 0. Nótese que la orden **NullSpace** nos permite obtener esta base directamente.

Ejemplo 9.1. Calcular la dimensión, la base y las ecuaciones paramétricas e implícitas del núcleo de la aplicación lineal f: $\mathbb{R}^3 \longrightarrow \mathbb{R}^4$ dada por f(x, y, z) = (2x – z, x + y, 3x + y – z, 2y + z).

En primer lugar calculemos la matriz asociada a la aplicación lineal respecto de las bases canónicas:

In[1]:=
```
f[{x_,y_,z_}]:={2x-z, x+y, 3x+y-z, 2y+z}
B= IdentityMatrix[3];
A= Transpose[Table [f[B[[i]]],{i,1,3}]]
MatrixForm[A]
```

Out[1]:=

$$\begin{pmatrix} 2 & 0 & -1 \\ 1 & 1 & 0 \\ 3 & 1 & -1 \\ 0 & 2 & 1 \end{pmatrix}$$

La dimensión del núcleo será:

In[2]:= **Dimensions[A][[2]]-MatrixRank[A]**

Out[2]:= `1`

MÉTODO 1. Resolvemos el sistema de ecuaciones lineales homogéneo,

In[3]:= **Solve[A.{x_1,x_2,x_3}=={0,0,0},{x_1,x_2,x_3}]**

Out[3]:= $\{\{x_2 \to -x_1, x_3 \to 2x_1\}\}$

Calculamos una base (usamos lo aprendido en la sesión 7), ejecutando previamente el comando **BaseImplícitas[]**:

In[4]:= **basenucleo =BaseImplicitas[{x_2 == -x_1,x_3==2x_1},3]**

Out[4]:= $\{\{1,-1,2\}\}$

Y ahora las ecuaciones paramétricas, ejecutando previamente el programa **Paramétricas[]** de la sesión 7:

In[5]:= **Paramétricas[basenucleo]**

Out[5]:= $\{x_1 == \lambda[1], x_2 == -\lambda[1], x_3 == 2\lambda[1]\}$

MÉTODO 2. Con **NullSpace[]**, directamente obtenemos la base,

Ahora primero calculamos la base del núcleo:

In[8]:= **baseNucleo=NullSpace[A]**

Out[8]:= `{{1,-1,2}}`

En este caso deducimos que la dimensión del núcleo es 1 y por tanto necesitamos un parámetro, a, para el cálculo de las paramétricas. Como el núcleo es un subespacio de $\mathbb{R}^3$, escribiremos una lista con tres coordenadas {x,y,z} y formaremos una ecuación de estas coordenadas, con la lista formada por las coordenadas de los vectores de la base del núcleo, multiplicada matricialmente por la lista de los parámetros. Por último, la función

LogicalExpand igualará término a término las listas implicadas, dando lugar a las ecuaciones paramétricas en la forma habitual.

In[9]:= **param={a};**
coord={x,y,z};
paramNucleo=LogicalExpand[coord==
Transpose[baseNucleo].param]

Out[9]:= `x = = a && y = = -a && z = = 2a`

La orden **Eliminate[paramNucleo, param]** hace que se elimine el único parámetro que hay en este caso, obteniendo las ecuaciones implícitas:

In[10]:= **Eliminate[paramNucleo, param]**

Out[10]: = `x = = -y && 2y = = - z`

Recordemos que el número de ecuaciones implícitas de un subespacio U de un espacio vectorial V, es igual a dim(V) - dim(U). En nuestro ejemplo, efectivamente, nos han salido 3-1=2 ecuaciones implícitas del núcleo de f.

■

Para calcular la imagen de la aplicación lineal, buscaremos un sistema de generadores, que según vimos en una proposición, puede obtenerse a partir de las imágenes mediante f de cualquier sistema de generadores del dominio. Teniendo en cuenta esto, sabemos que las columnas de la matriz asociada a f, constituyen un sistema generador de Im (f). Así, una base no será más que el conjunto formado por el mayor número de columnas que sean linealmente independientes y que podemos obtenerlas a partir de la forma normal de Hermite de la matriz asociada a f.

Ejemplo 9.2. Calcular base, dimensión, ecuaciones paramétricas e implícitas de la imagen de la aplicación lineal f del ejercicio anterior.

En el ejercicio anterior hemos definido la aplicación f y hemos calculado su matriz asociada respecto de las bases canónicas:

In[11]:= **generadorImagen=Transpose[A];**
RowReduce[generadorImagen]

Out[11]:= `{{1,0,1,-1},{0,1,1,2},{0,0,0,0}}`

Como vemos, en este caso, las filas linealmente independientes de la traspuesta de A (o lo que es igual, las columnas de A linealmente independientes) son las dos primeras filas, por tanto la base será:

In[12]:= **baseImagen=Table[%[[i]],{i,2}]**

Out[12]:= `{{1,0,1,-1},{0,1,1,2}}`

Ahora tendremos que introducir dos parámetros {a,b} y coordenadas {x,y,z,t} pues la imagen es subespacio vectorial de $\mathbb{R}^4$. Las ecuaciones correspondientes son:

In[13]:= **param={a,b};**
coord={x,y,z,t};
paramImagen=LogicalExpand[coord == Transpose[baseImagen].param]

Out[13]:= `t = = -a+2b && x = = a && y = = b && z = = a+b`

In[14]:= **Eliminate[paramImagen, param]**

Out[14]:= `t = = -x+2y && x = = -y+z`

En este caso hemos obtenido dim(V') – dim(Im(f)) = 4 – 2 = 2 ecuaciones implícitas del subespacio imagen de la aplicación lineal f.

■

Como fácilmente se puede observar el método anterior no es totalmente programable pues es necesario intervenir añadiendo los vectores que amplían la base del núcleo hasta una del espacio V', o bien calculando le rango de la matriz formada por la columnas de la matriz asociada a f. Veamos como la forma normal de Hermite nos facilita el cálculo del núcleo y la imagen de f.

Si al calcular la forma de Hermite por columnas de A realizamos las operaciones elementales sobre $\left(\frac{A}{I}\right)$, obtenemos $\left(\frac{C}{P}\right)$ donde P es la matriz regular de orden n tal que C = A.P, pues bien se tiene que las columnas no nulas de C forma una base de Im (f) y las columnas de P que están bajo las columnas de ceros de C (si las hay) forman una base de Ker (f).

Recordemos que en el Mathematica la orden **RowReduce[A]** nos calcula la forma normal de Hermite por filas, luego al hacer lo anterior con el Mathematica nosotros trabajaremos por filas transponiendo la matriz $\left(\frac{A}{I}\right)$ antes de calcular la forma de Hermite y al final transponiendo el resultado (C | P).

Ejemplo 9.3. Usar la forma normal de Hermite para calcular bases del núcleo y la imagen de la aplicación lineal f del ejercicio 9.1.

In[15]:= **f[{x_,y_,z_}]:={2x-z,x+y,3x+y-z,2y+z}**
B= IdentityMatrix[3];
A= Transpose[Table [f[B[[i]]],{i,1,3}]];
Join[A,B];
AI=Transpose[%];
CP=RowReduce[AI];
MatrixForm[Transpose[CP]]

Out[15]:=

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & -2 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & -1 \\ -1 & 0 & 2 \end{pmatrix}$$

Por tanto, una base de la imagen es {(1,0,1,1),(0,1,1,-2)} y una base del núcleo es {(1,-1,2)}.

2.- TIPOS DE APLICACIONES LINEALES.

Una aplicación lineal pueden ser **monomorfismo, epimorfismo o isomorfismo** si como aplicación es inyectiva, sobreyectiva o biyectiva, respectivamente.

Proposición

Sean V y V' espacios vectoriales de dimensión n y m, respectivamente. Dada una aplicación lineal $f:V \longrightarrow V'$, y A la matriz asociada a f respecto de ciertas bases B y B' de V y V' respectivamente, se verifica:

1. f es inyectiva $\Leftrightarrow$ Ker(f) = {0} $\Leftrightarrow$ rang(A) = n.
2. f es sobreyectiva $\Leftrightarrow$ Im(f) = V' $\Leftrightarrow$ rang(A) = m.
3. f es biyectiva $\Leftrightarrow$ A es cuadrada y regular.

Según la proposición anterior la aplicación f del ejemplo no es ni inyectiva pues Ker (f) =L{(1,-1,2)} $\neq$ {0}, ni sobreyectiva pues Im (f) = L{(1,0,1,1),(0,1,1,-2)} $\neq$ $\mathbb{R}^4$.