Ex.7 de la feuille 2 dans un cas concret

F-X. Dehon - 12 oct. 2020 - dehon[@]unice.fr

Enoncé :

7. Une matrice symétrique réelle est dite (définie) positive si toutes ses valeurs propres sont (strictement) positives.

7.a. Montrer que $S \in M_n(\mathbb{R})$ est une matrice symétrique positive si et seulement s'il existe une matrice $A \in M_n(\mathbb{R})$ telle que $S = {}^tAA$.

7.b. Montrer que $S \in M_n(\mathbb{R})$ est une matrice symétrique définie positive si et seulement s'il existe une matrice $A \in Gl_n(\mathbb{R})$ telle que $S = {}^tAA$.

7.c. Montrer que pour toute matrice symétrique positive S, il existe une et une seule matrice symétrique positive R telle que $R^2 = S$.

On spécifie ci-dessous la réponse à l'ex.7 dans le cas $S = \begin{pmatrix} 3 & 2 \\ 2 & 3 \end{pmatrix}$

1ère méthode avec une diagonalisation de S

In [1]:
```
S=matrix([[3,2],[2,3]]);show(S)
```
Out[1]:

$$\begin{pmatrix} 3 & 2 \\ 2 & 3 \end{pmatrix}$$

In [2]:
```
#Valeurs propres ?
print S.charpoly().roots()
```
```
[(5, 1), (1, 1)]
```

In [3]:
```
print (S-identity_matrix(2)).right_kernel_matrix() #base de vecteurs propres pour la val
eur propre 1
```
```
[ 1 -1]
```

In [4]:
```
print (S-5*identity_matrix(2)).right_kernel_matrix()
```
```
[1 1]
```

In [5]:
```
u=vector([1,-1]);print u*u #produit scalaire usuel
v=vector([1,1])
```
```
2
```

In [6]:
```
P=block_matrix(1,2,[matrix(u/sqrt(u*u)).transpose(),matrix(v/sqrt(v*v)).transpose()])
print P
```
```
[ 1/2*sqrt(2)| 1/2*sqrt(2)]
[-1/2*sqrt(2)| 1/2*sqrt(2)]
```

In [7]:
```
D=diagonal_matrix([1,sqrt(5)])
```

In [8]:
```
print S*P==P*D^2 #vérification P est une base de vecteurs propres et les valeurs propres
associées sont les coef. de D^2
```
```
True
```

In [9]:
```
R=P*D*P.transpose();show(R)
```
Out[9]:

$$\begin{pmatrix} \frac{1}{2}\sqrt{5}+\frac{1}{2} & \frac{1}{2}\sqrt{5}-\frac{1}{2} \\ \frac{1}{2}\sqrt{5}-\frac{1}{2} & \frac{1}{2}\sqrt{5}+\frac{1}{2} \end{pmatrix}$$

In [10]:
```
show(R^2)
```

Out[10]:

$$\begin{pmatrix} \frac{1}{4}\left(\sqrt{5}+1\right)^2+\frac{1}{4}\left(\sqrt{5}-1\right)^2 & \frac{1}{2}\left(\sqrt{5}+1\right)\left(\sqrt{5}-1\right) \\ \frac{1}{2}\left(\sqrt{5}+1\right)\left(\sqrt{5}-1\right) & \frac{1}{4}\left(\sqrt{5}+1\right)^2+\frac{1}{4}\left(\sqrt{5}-1\right)^2 \end{pmatrix}$$

R^2 est égal à S ?

In [11]:
```
show((R^2).simplify_full())
print R^2==S
```

Out[11]:

$$\begin{pmatrix} 3 & 2 \\ 2 & 3 \end{pmatrix}$$

```
True
```

2ème méthode avec le calcul d'une base orthonormée pour le produit scalaire de matrice S.

In [12]:
```
def B(u,v):return(u*S*v) #produit scalaire de matrice S
def N(u):return(sqrt(B(u,u))) # norme associée
```

In [13]:
```
#base canonique de R^2
f1=vector([1,0])
f2=vector([0,1])
```

In [14]:
```
e1=f1/N(f1);print e1
```

```
(1/3*sqrt(3), 0)
```

In [15]:
```
f2p=f2-B(f2,e1)*e1
e2=f2p/N(f2p);print e2
```

```
(-2/5*sqrt(5/3), 3/5*sqrt(5/3))
```

In [16]:
```
BON=[e1,e2]
show(matrix(2,2,lambda i,j:B(BON[i],BON[j]))) #vérification (e1,e2) est orthonormée pour B
```

Out[16]:

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

In [17]:
```
P=block_matrix(1,2,[matrix(e1).transpose(),matrix(e2).transpose()]);show(P) #Mat(e1,e2)
```

Out[17]:

$$\left(\begin{array}{c|c} \frac{1}{3}\sqrt{3} & -\frac{2}{5}\sqrt{\frac{5}{3}} \\ 0 & \frac{3}{5}\sqrt{\frac{5}{3}} \end{array}\right)$$

In [18]:
```
show(P.transpose()*S*P) #vérification matricielle
```

Out[18]:

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

In [19]:
```
A=P^(-1);show(A)
print S==A.transpose()*A
```

Out[19]:

$$\begin{pmatrix} \sqrt{3} & \frac{2}{3}\sqrt{3} \\ 0 & \sqrt{\frac{5}{3}} \end{pmatrix}$$

```
True
```