

## matrices2

November 24, 2021

```
[6]: tA="""(1  0 -1 0  0  0)
(0  1  0 0  0 -1)
(0  0  0 1 -1  0)
(0  0  0 1  0 -1)
(1 -1  0 0  0  0)
(0  0  1 0 -1  0)"""
tB="""(1  0  0 0  0  0)
(0  1  0 0  0  0)
(0  0  1 0  0  0)
(0  0  1 1  0  0)
(1 -1  0 0  1  0)
(0  0  0 -1  1)"""
tL12='''(0 1 0 0 0 0)
(1 0 0 0 0 0)
(0 0 0 0 1 0)
(0 0 0 0 0 1)
(0 0 1 0 0 0)
(0 0 0 1 0 0)'''
tL23='''(0 0 1 0 0 0)
(0 0 0 0 0 1)
(1 0 0 0 0 0)
(0 0 0 0 1 0)
(0 0 0 1 0 0)
(0 1 0 0 0 0)'''
tM='''( 0 -2  0 1  2 -1  0  0  0  3  0 -3)
( 1 -1 -2 0  0  2  3 -3  0  0  0  0)
(-2  0  1 2 -1  0  0  0  3  0 -3  0)
( 0  0  0 1  0  1  0 -2  0 -1 -2 -1)
( 1  1  0 0  0  0 -1 -1 -2  0  0 -2)
( 0  0  1 0  1  0 -2  0 -1 -2 -1  0)'''
tM2='''( 0  0 1  0  0  0  0)
( 1  0 0  0  0  0  0)
( 0  0 0  0  1  0  0)
( 0  0 0  1  0  0  0)
( 0  0 0  0  0  1  0)
( 0  1 0  0  0  0  0)'''
tM3='''( 0  0 0 1  0  0  0  0  0  0  0  0  0)
( 0  0 0 0  1  0  0  0  0  0  0  0  0)
( 0  0 0 0  0  1  0  0  0  0  0  0  0)
( 0  0 0 0  0  0  1  0  0  0  0  0  0)
( 0  0 0 0  0  0  0  1  0  0  0  0  0)
( 0  0 0 0  0  0  0  0  1  0  0  0  0)
( 0  0 0 0  0  0  0  0  0  1  0  0  0)
( 0  0 0 0  0  0  0  0  0  0  1  0  0)
( 0  0 0 0  0  0  0  0  0  0  0  1  0)
( 0  0 0 0  0  0  0  0  0  0  0  0  1)
( 0  0 0 0  0  0  0  0  0  0  0  0  0)'''
```

```
( 1  0  0 0  0  0  0  0  0  0  0  0)
(-2  1  0 2 -5  3  6  0  3 -3 -3  0)
( 0  1  0 1 -2  1  0 -1  0 -2 -2 -1)
( 1  1  2 0  0 -1 -4 -1 -2  0  0 -1)
( 0  0  1 0  1  0 -2 -1 -1 -1 -1 -1)'''
```

```
[7]: tg2='(1 -1 -1 -1  1  1)'
```

```
[8]: def parse_matrix(text, field=QQ):
      return matrix([map(field,s.strip('()').split()) for s in text.strip().
      ↪split('\n')])
```

```
[9]: def parse_vector(text, field=QQ, to_strip='()'):
      return vector(map(field,text.strip().strip(to_strip).split()))
```

```
[10]: def matrix_to_latex(M, debut = r'\begin{pmatrix}' + '\n ', fin = '\n' +
      ↪r'\end{pmatrix}',
      separateurLigne = r' \\' + '\n ', separateurColonne = ' &
      ↪'):
      print(debut + \
      separateurLigne.join([separateurColonne.join(map(str, row)) for row in
      ↪M.rows()]) + \
      fin, end='')
```

```
[11]: def ascii_transposed_vector_to_latex(v, field=QQ, to_strip='()'):
      # matrix_to_latex(matrix(parse_vector(v)).T, separateurLigne = r' \\'
      matrix_to_latex(matrix(parse_vector(v, field=field, to_strip=to_strip)).T,
      ↪separateurLigne = r' \\' ,
      fin = '\n' + r'\end{pmatrix}')
```

```
[12]: def ascii_vector_to_latex(v, to_strip='()'):
      # matrix_to_latex(matrix(parse_vector(v)).T, separateurLigne = r' \\'
      matrix_to_latex(matrix(parse_vector(v, to_strip=to_strip)), separateurLigne
      ↪= r' \\' )
```

```
[13]: def ascii_transposed_vector_to_latex2(v, to_strip='()', debut =
      ↪r'\begin{pmatrix}' + '\n ',
      fin = '\n' + r'\end{pmatrix}',
      ↪separateurLigne = r' \\' ):
      print(debut + \
      separateurLigne.join(v.strip().strip(to_strip).split()) + \
      fin, end='')
```

```
[19]: M=matrix([[1,2],[3,4]])
matrix_to_latex(M)
```

```
\begin{pmatrix}
1 & 2 \\\
```

```

3 & 4
\end{pmatrix}

```

```

[ ]: M=matrix(eval('[[1,2],[3,4]]'))
      matrix_to_latex(M)

```

```

[8]: A=parse_matrix(tA)
      A.base_ring(), A

```

```

[8]: (
      [ 1  0 -1  0  0  0]
      [ 0  1  0  0  0 -1]
      [ 0  0  0  1 -1  0]
      [ 0  0  0  1  0 -1]
      [ 1 -1  0  0  0  0]
      Rational Field, [ 0  0  1  0 -1  0]
      )

```

```

[9]: L12=parse_matrix(tL12)
      L23=parse_matrix(tL23)
      L12,L23

```

```

[9]: (
      [0 1 0 0 0 0] [0 0 1 0 0 0]
      [1 0 0 0 0 0] [0 0 0 0 0 1]
      [0 0 0 0 1 0] [1 0 0 0 0 0]
      [0 0 0 0 0 1] [0 0 0 0 1 0]
      [0 0 1 0 0 0] [0 0 0 1 0 0]
      [0 0 0 1 0 0], [0 1 0 0 0 0]
      )

```

```

[10]: g2=parse_vector(tg2)
      g2

```

```

[10]: (1, -1, -1, -1, 1, 1)

```

```

[11]: v=vector([1,-1,1,-1,-1,1])

```

```

[12]: v*A

```

```

[12]: (0, 0, 0, 0, -2, 2)

```

```

[13]: v=vector([1,-1,-1,1,-1,1])
      v*A

```

```

[13]: (0, 0, 0, 0, 0, 0)

```

```

[14]: (L12+1)*g2

```

```
[14]: (0, 0, 0, 0, 0, 0)
```

```
[15]: (L23+1)*g2
```

```
[15]: (0, 0, 0, 0, 0, 0)
```

```
[16]: M5=parse_matrix('''
( 0  2  0  1 -2 -1  0  0  0  3  0 -3)
( 1 -1 -2  0  0  2  3 -3  0  0  0  0)
(-2  0  1  2 -1  0  0  0  3  0 -3  0)
( 0  0  0  1  0  1  0 -2  0 -1 -2 -1)
( 1  1  0  0  0  0 -1 -1 -2  0  0 -2)
( 0  0  1  0  1  0 -2  0 -1 -2 -1  0)
''')
M5,M5.left_kernel()
```

```
[16]: (
[ 0  2  0  1 -2 -1  0  0  0  3  0 -3]
[ 1 -1 -2  0  0  2  3 -3  0  0  0  0]
[-2  0  1  2 -1  0  0  0  3  0 -3  0]
[ 0  0  0  1  0  1  0 -2  0 -1 -2 -1]
[ 1  1  0  0  0  0 -1 -1 -2  0  0 -2]
[ 0  0  1  0  1  0 -2  0 -1 -2 -1  0],
```

Vector space of degree 6 and dimension 2 over Rational Field

Basis matrix:

```
[ 1  0 -1  1 -2  1]
[ 0  1  1 -2  1  1]
)
```

```
[17]: g1=parse_vector('(1 1 1 1 1 1)')
g1
```

```
[17]: (1, 1, 1, 1, 1, 1)
```

```
[18]: g2=parse_vector('(1 -1 -1 -1 1 1)')
g2
```

```
[18]: (1, -1, -1, -1, 1, 1)
```

```
[19]: g3=parse_vector('(1 -1 1 0 -1 0)')
g3
```

```
[19]: (1, -1, 1, 0, -1, 0)
```

```
[20]: g4=parse_vector('(1 1 1 -2 1 -2)')
g4
```

[20]: (1, 1, 1, -2, 1, -2)

[21]:  $(L_{12}+1)*g_3$

[21]: (0, 0, 0, 0, 0, 0)

[22]:  $(L_{12}-1)*g_4$

[22]: (0, 0, 0, 0, 0, 0)

[23]:  $L_{23}*g_3$

[23]: (1, 0, 1, -1, 0, -1)

[24]:  $(g_3+g_4)/2$

[24]: (1, 0, 1, -1, 0, -1)

[25]:  $L_{23}*g_4$

[25]: (1, -2, 1, 1, -2, 1)

[26]:  $(3*g_3-g_4)/2$

[26]: (1, -2, 1, 1, -2, 1)

[27]:  $Q_6 = \text{VectorSpace}(\mathbb{Q}\mathbb{Q}, 6)$

[28]:  $B=[g_1, g_2, g_3, g_4]$

```
[29]: for v in Q6.basis():  
        if v not in span(B):  
            B.append(v)
```

[30]: B

[30]: [(1, 1, 1, 1, 1, 1),  
(1, -1, -1, -1, 1, 1),  
(1, -1, 1, 0, -1, 0),  
(1, 1, 1, -2, 1, -2),  
(1, 0, 0, 0, 0, 0),  
(0, 1, 0, 0, 0, 0)]

[31]:  $P=\text{matrix}(B).T$   
P

[31]: [ 1 1 1 1 1 0]  
[ 1 -1 -1 1 0 1]  
[ 1 -1 1 1 0 0]

```

[ 1 -1  0 -2  0  0]
[ 1  1 -1  1  0  0]
[ 1  1  0 -2  0  0]

```

```
[32]: P^-1*L12*P
```

```

[32]: [ 1  0  0  0  0  0]
      [ 0 -1  0  0  0  0]
      [ 0  0 -1  0  0  0]
      [ 0  0  0  1  0  0]
      [ 0  0  0  0  0  1]
      [ 0  0  0  0  1  0]

```

```
[33]: P^-1*L23*P
```

```

[33]: [ 1  0  0  0  0  1/3  1/6]
      [ 0 -1  0  0  0  0  1/2]
      [ 0  0  1/2  3/2  1/2  1/2]
      [ 0  0  1/2 -1/2  1/6 -1/6]
      [ 0  0  0  0  0 -1 -1]
      [ 0  0  0  0  0  0  1]

```

```
[34]: M=L12
```

```
M
```

```

[34]: [0 1 0 0 0 0]
      [1 0 0 0 0 0]
      [0 0 0 0 1 0]
      [0 0 0 0 0 1]
      [0 0 1 0 0 0]
      [0 0 0 1 0 0]

```

```
[ ]:
```

```
[35]: matrix_to_latex(L12)
```

```

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

```

```

[36]: L123=parse_matrix(''0 0 0 0 0 1
0 0 1 0 0 0
0 0 0 1 0 0

```

```

0 1 0 0 0 0
1 0 0 0 0 0
0 0 0 0 1 0
''' )

```

[37]: L123

```

[37]: [0 0 0 0 0 1]
      [0 0 1 0 0 0]
      [0 0 0 1 0 0]
      [0 1 0 0 0 0]
      [1 0 0 0 0 0]
      [0 0 0 0 1 0]

```

[38]: matrix\_to\_latex(L123)

```

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

```

[39]: L23

```

[39]: [0 0 1 0 0 0]
      [0 0 0 0 0 1]
      [1 0 0 0 0 0]
      [0 0 0 0 1 0]
      [0 0 0 1 0 0]
      [0 1 0 0 0 0]

```

[40]: matrix\_to\_latex(L23)

```

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

```

[41]: ascii\_transposed\_vector\_to\_latex('0 0 0 1 0 1')

```

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

```

```
\end{pmatrix}
```

```
[42]: s='''          f4=(0 0 0 1 0 1)
          f5=(1 1 0 0 0 0)
          f6=(0 0 1 0 1 0)
          f1=(0  0 0 1  0 -1)
          f2=(1 -1 0 0  0  0)
          f3=(0  0 1 0 -1  0)
          e1=(1 0 -1 0  0  0)
          e2=(0 1  0 0  0 -1)
          e3=(0 0  0 1 -1  0)
          e4=(1 0 1 0 0 0)
          e5=(0 1 0 0 0 1)
          e6=(0 0 0 1 1 0)'''
```

```
[43]: for l in s.split('\n'):
        ascii_transposed_vector_to_latex(l.strip().lstrip('fe123456='))
```

```
\begin{pmatrix}
0 \\0 \\0 \\1 \\0 \\1
\end{pmatrix}\begin{pmatrix}
1 \\1 \\0 \\0 \\0 \\0
\end{pmatrix}\begin{pmatrix}
0 \\0 \\1 \\0 \\1 \\0
\end{pmatrix}\begin{pmatrix}
0 \\0 \\0 \\1 \\0 \\-1
\end{pmatrix}\begin{pmatrix}
1 \\-1 \\0 \\0 \\0 \\0
\end{pmatrix}\begin{pmatrix}
0 \\0 \\1 \\0 \\-1 \\0
\end{pmatrix}\begin{pmatrix}
1 \\0 \\-1 \\0 \\0 \\0
\end{pmatrix}\begin{pmatrix}
0 \\1 \\0 \\0 \\0 -1
\end{pmatrix}\begin{pmatrix}
0 \\0 \\0 1 -1  0
\end{pmatrix}\begin{pmatrix}
1 \\0 1 0 0 0
\end{pmatrix}\begin{pmatrix}
0 \\1 0 0 0 1
\end{pmatrix}\begin{pmatrix}
0 \\0 0 1 1 0
\end{pmatrix}
```

```
[54]: M1324=parse_matrix(''0 0 0 1 0 0
0 0 1 0 0 0
1 0 0 0 0 0
0 1 0 0 0 0
```



```

0 0 0 0 1 0
0 0 0 0 0 1
'''
matrix_to_latex(M1324)

```

```

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

```

```

[51]: for l in s2.split('\n\n'):
        ascii_transposed_vector_to_latex(l)

```

```

\begin{pmatrix}
1 & \backslash 1 & \backslash -1 & \backslash -1 & \backslash 0 & \backslash 0
\end{pmatrix} \begin{pmatrix}
1 & \backslash 1 & \backslash 1 & \backslash 1 & \backslash 0 & \backslash 0
\end{pmatrix} \begin{pmatrix}
0 & \backslash 0 & \backslash 0 & \backslash 0 & \backslash 1 & \backslash 0
\end{pmatrix} \begin{pmatrix}
0 & \backslash 0 & \backslash 0 & \backslash 0 & \backslash 0 & \backslash 1
\end{pmatrix}

```

```

[63]: M1324.eigenvectors_right()

```

```

[63]: [(-1,
        [
          (1, 1, -1, -1, 0, 0)
        ],
        1),
        (1,
         [
          (1, 1, 1, 1, 0, 0),
          (0, 0, 0, 0, 1, 0),
          (0, 0, 0, 0, 0, 1)
        ],
        3),
        (-1*I, [(1, -1, 1*I, -1*I, 0, 0)], 1),
        (1*I, [(1, -1, -1*I, 1*I, 0, 0)], 1)]

```

```

[92]: s3='''1 0 0 0 j^2 j
0 1 j j^2 0 0
1 0 0 0 j j^2
0 1 j^2 j 0 0

```

```
1 -1 -i i 0 0
1 -1 i -i 0 0'''
```

```
[97]: for l in s3.split('\n'):
        ascii_transposed_vector_to_latex2(l)
```

```
\begin{pmatrix}
  1 \\ 0 \\ 0 \\ 0 \\ j^2 \\ j
\end{pmatrix} \begin{pmatrix}
  0 \\ 1 \\ j \\ j^2 \\ 0 \\ 0
\end{pmatrix} \begin{pmatrix}
  1 \\ 0 \\ 0 \\ 0 \\ j \\ j^2
\end{pmatrix} \begin{pmatrix}
  0 \\ 1 \\ j^2 \\ j \\ 0 \\ 0
\end{pmatrix} \begin{pmatrix}
  1 \\ -1 \\ -i \\ i \\ 0 \\ 0
\end{pmatrix} \begin{pmatrix}
  1 \\ -1 \\ i \\ -i \\ 0 \\ 0
\end{pmatrix}
```

[73]:  $j = e^{(i \cdot 2 \cdot \pi / 3)}$

```
[77]: expand(j^3)
```

[77]: 1

```
[93]: s3.split('\n')
```

```
[93]: ['1 0 0 0 j^2 j',
      '0 1 j j^2 0 0',
      '1 0 0 0 j j^2',
      '0 1 j^2 j 0 0',
      '1 -1 -i i 0 0',
      '1 -1 i -i 0 0']
```

```
[99]: L123.eigenvectors_right()
```

```
[99]: [(1,
        [
            (1, 0, 0, 0, 1, 1),
            (0, 1, 1, 1, 0, 0)
        ],
        2),
        (-0.5000000000000000? - 0.866025403784439?*I,
         [(1, 0, 0, 0, -0.5000000000000000? + 0.866025403784439?*I,
          -0.5000000000000000? - 0.866025403784439?*I),
          (0, 1, -0.5000000000000000? - 0.866025403784439?*I, -0.5000000000000000? +
          0.866025403784439?*I, 0, 0)],
```

```

2),
(-0.5000000000000000? + 0.866025403784439?*I,
 [(1, 0, 0, 0, -0.5000000000000000? - 0.866025403784439?*I,
-0.5000000000000000? + 0.866025403784439?*I),
 (0, 1, -0.5000000000000000? + 0.866025403784439?*I, -0.5000000000000000? -
0.866025403784439?*I, 0, 0)],
2)]

```

```
[101]: n(j)
```

```
[101]: -0.5000000000000000 + 0.866025403784439*I
```

```
[102]: n(j^2)
```

```
[102]: -0.5000000000000000 - 0.866025403784439*I
```

```
[104]: L12.eigenvectors_right()
```

```
[104]: [(1,
 [
 (1, 1, 0, 0, 0, 0),
 (0, 0, 1, 0, 1, 0),
 (0, 0, 0, 1, 0, 1)
 ],
 3),
 (-1,
 [
 (1, -1, 0, 0, 0, 0),
 (0, 0, 1, 0, -1, 0),
 (0, 0, 0, 1, 0, -1)
 ],
 3)]
```

```
[105]: L23.eigenvectors_right()
```

```
[105]: [(1,
 [
 (1, 0, 1, 0, 0, 0),
 (0, 1, 0, 0, 0, 1),
 (0, 0, 0, 1, 1, 0)
 ],
 3),
 (-1,
 [
 (1, 0, -1, 0, 0, 0),
 (0, 1, 0, 0, 0, -1),
 (0, 0, 0, 1, -1, 0)
 ],
 3)]
```

3)]

[ ]: