Kernel: SageMath 9.7

Finding eigenvalues and eigenvectors of matrices in CoCalc

In [1]:
```
# Define a matrix, one row at a time.
# Here's the one we did in class:
M=matrix([[1,2],[-2,3]])
show(M)
```

Out[1]: $\begin{pmatrix} 1 & 2 \\ -2 & 3 \end{pmatrix}$

In [3]:
```
show(M.eigenvalues())
```

Out[3]: $[2 - 1.732050807568878?\sqrt{-1}, 2 + 1.732050807568878?\sqrt{-1}]$

Oops! we got $2 \pm \sqrt{3} = 2 \pm 1.73205\ldots$. But the eigenvalues of this matrix were actually imaginary. What went wrong??

That was my fault! I wrote down the quadratic formula with the wrong sign of things under the square root: The solutions of $ax^2 + bx + c = 0$ are actually:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

So the argument of the square root was negative! Try another matrix:

In [10]:
```
M=matrix([[4,-1],[2,1]])
show(M)
```

Out[10]: $\begin{pmatrix} 4 & -1 \\ 2 & 1 \end{pmatrix}$

The characteristic polynomial--the determinant of the matrix $\left(\mathbb{M} - \lambda \mathbb{1}\right) = 0$ -- is $(4 - \lambda)(1 - \lambda) - (-1)(2) = 0$. Solve the quadratic to find λ:

In [8]:
```
lda=var('lda')   # 'lda' is short for "lambda"
solve((4-lda)*(1-lda)+2==0, lda)
```

Out[8]:
```
[lda == 3, lda == 2]
```

In [12]:
```
# We can get the *hopefully same* eigenvalues using...
show(M.eigenvalues())
```

Out[12]: $[3, 2]$

In [13]:
```
# This command gives you eigenvalues and eigen vectors:
show( M.eigenvectors_right())
```

Out[13]: $[(3, [(1,\ 1)], 1), (2, [(1,\ 2)], 1)]$

The response is a pair of *triples*. The first triple contains an **eigenvalue** (3) followed by the corresponding **eigenvector** (the column vector (1,1)), followed by the **multiplicity** or degeneracy of the eigenvalue (1).

Let's test the first eigenvector. Multiplying the matrix $\mathbb{M}$ by the first eigenvector should produce a vector which is 3 times the eigenvector...(and also test the second...)

In [15]:
```
ev1=matrix([[1],[1]]); ev2=matrix([[1],[2]])
show(ev1)
show(M*ev1)
print("Ah-ha, ev1 is an eigenvector with eigenvalue 3\n\n")
show(ev2)
show(M*ev2)
print("And M*ev2 gives 2*ev2!")
```

Out[15]:

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

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

```
Ah-ha, ev1 is an eigenvector with eigenvalue 3
```

$$\begin{pmatrix} 1 \\ 2 \end{pmatrix}$$

$$\begin{pmatrix} 2 \\ 4 \end{pmatrix}$$

```
And M*ev2 gives 2*ev2!
```

In [19]:
```
# You can "normalize" a vector to 1 by multiplying by 1/(Norm of the vector)
print("Norm of ev1 is ",ev1.norm(), " that is, the sqrt of (1^2+1^2).")
nev1=(1/ev1.norm())*ev1
show(nev1)
```

Out[19]:
```
Norm of ev1 is  1.4142135623730951  that is, the sqrt of (1^2+1^2).
```

$$\begin{pmatrix} 0.7071067811865475 \\ 0.7071067811865475 \end{pmatrix}$$

Using imaginary numbers

Sagemath defines the symbol `I` to mean $i = \sqrt{-1}$.

E.g. the S_y matrix is this matrix:

In [22]:
```
hbar=var('hbar')
```

```
Sy=hbar/2*matrix([[0,-I],[I,0]])
show(Sy)
```

Out[22]:

$$\begin{pmatrix} 0 & -\frac{1}{2}i\,hbar \\ \frac{1}{2}i\,hbar & 0 \end{pmatrix}$$

In [23]:

```
show(Sy.eigenvectors_right())
```

Out[23]:

$$\left[\left(-\frac{1}{2}\,hbar, \left[(1,\,-i)\right], 1\right), \left(\frac{1}{2}\,hbar, \left[(1,\,i)\right], 1\right)\right]$$

In [0]:

```
```