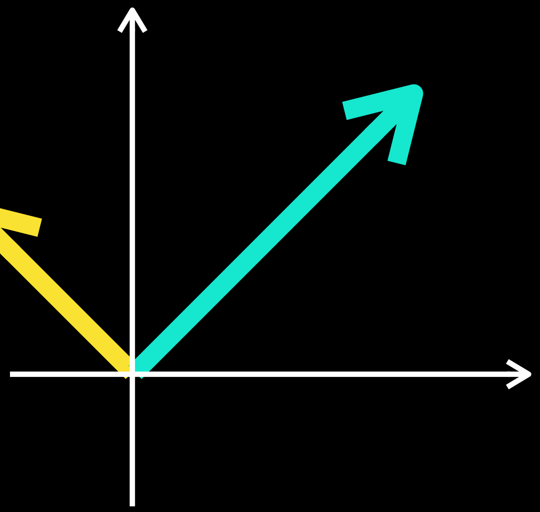
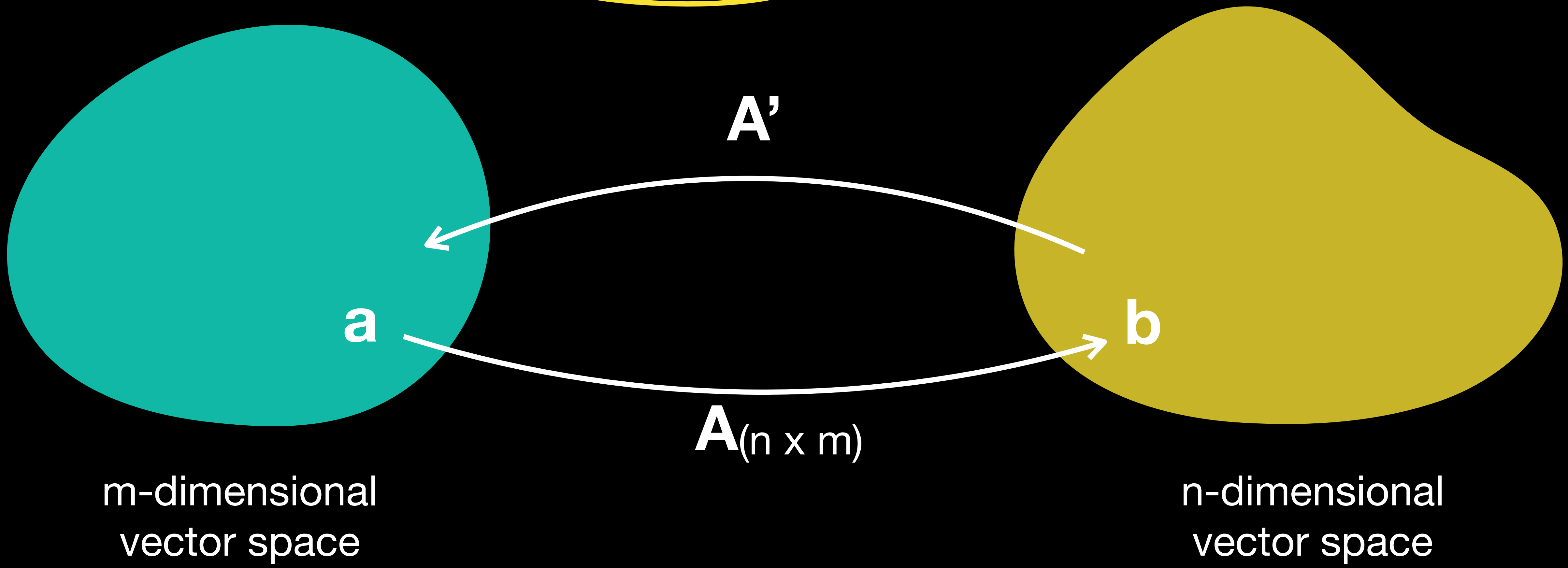
Inverse Matrix

Linear Algebra Essentials



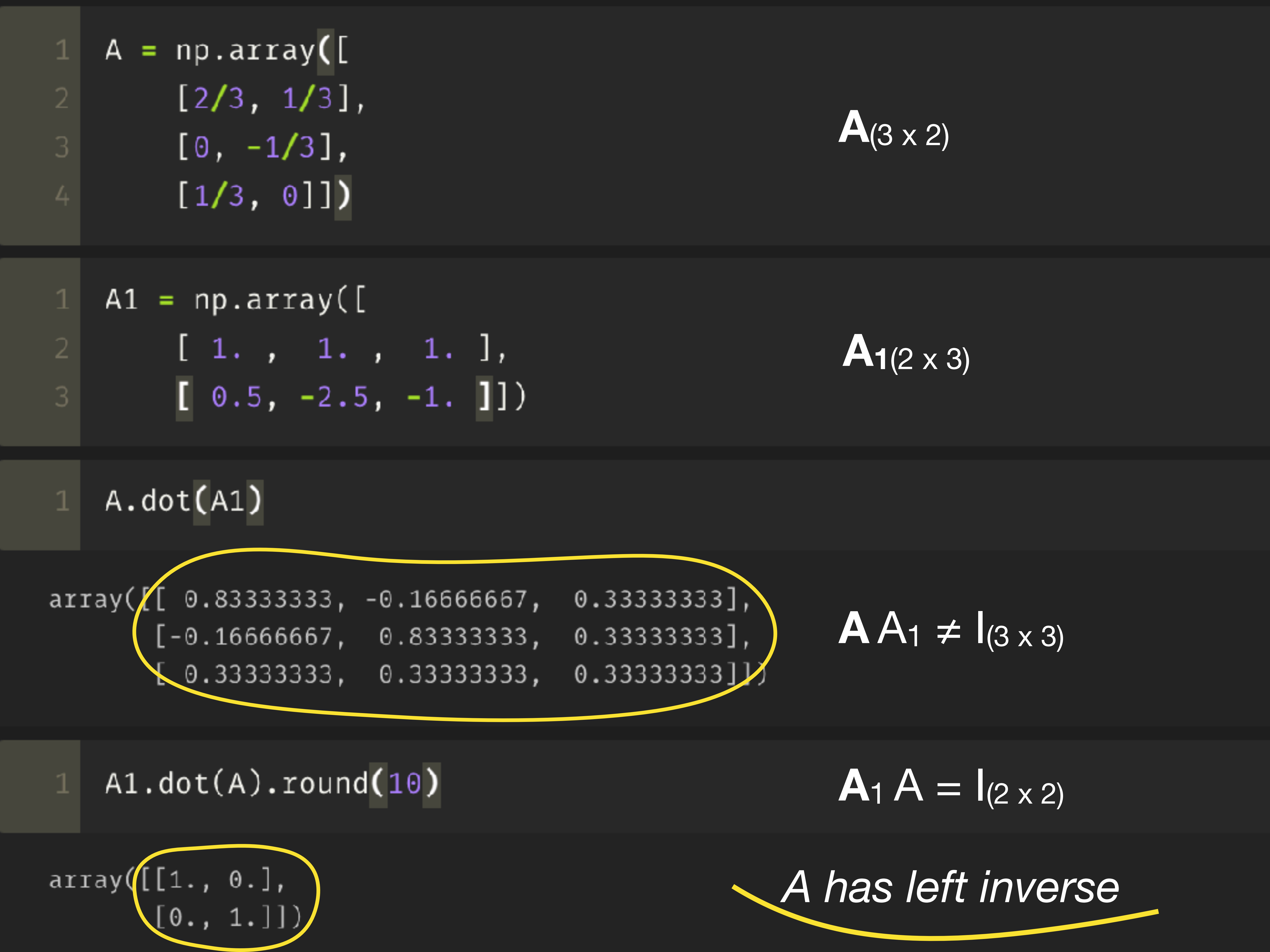
$$\boxed{\mathbf{A}_{(n \times m)}\ \mathbf{a}_{(m \times 1)}} = \mathbf{b}_{(n \times 1)}$$



$$\mathbf{A'}\boxed{\mathbf{b}_{(n \times 1)}} = a_{(m \times 1)}$$

$\mathbf{a} = \mathbf{A'}_{(m \times n)} \boxed{\mathbf{A}_{(n \times m)}\ a} = I_{(m \times m)}\ a$ - A has left inverse A’

$\mathbf{b} = \mathbf{A}_{(n \times m)}\ \mathbf{A'}_{(m \times n)}\ b = I_{(n \times n)}\ b$ - A has right inverse A’



```
A = np.array([
      [2/3, 1/3],
      [0, -1/3],
      [1/3, 0]])
```

$\mathbf{A}_{(3 \text{ x } 2)}$

```
A1 = np.array([
      [ 1. ,  1. ,  1. ],
      [ 0.5, -2.5, -1. ]])
```

$\mathbf{A}_{\mathbf{1}(2 \text{ x } 3)}$

```
A.dot(A1)
```

```
array([[ 0.83333333, -0.16666667,  0.33333333],
       [-0.16666667,  0.83333333,  0.33333333],
       [ 0.33333333,  0.33333333,  0.33333333]])
```

$\mathbf{A}\, A_1 \neq I_{(3 \text{ x } 3)}$

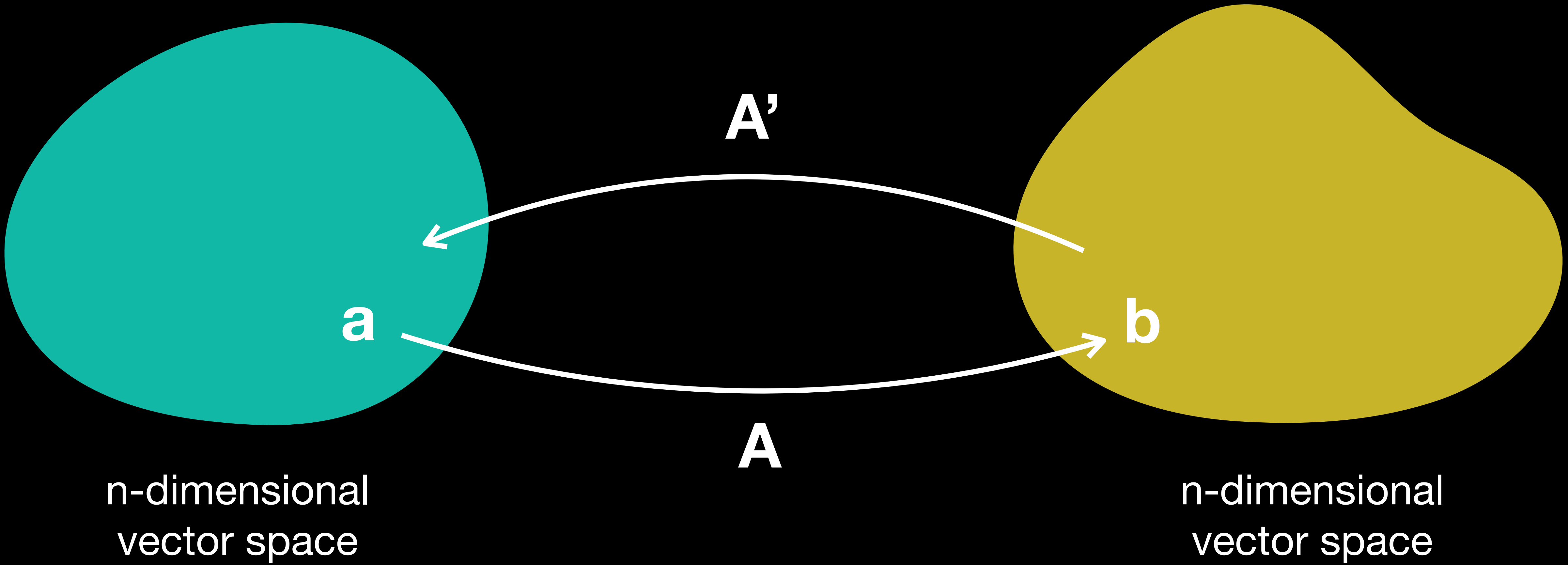
```
A1.dot(A).round(10)
```

$\mathbf{A}_1\, A = I_{(2 \text{ x } 2)}$

```
array([[1., 0.],
       [0., 1.]])
```

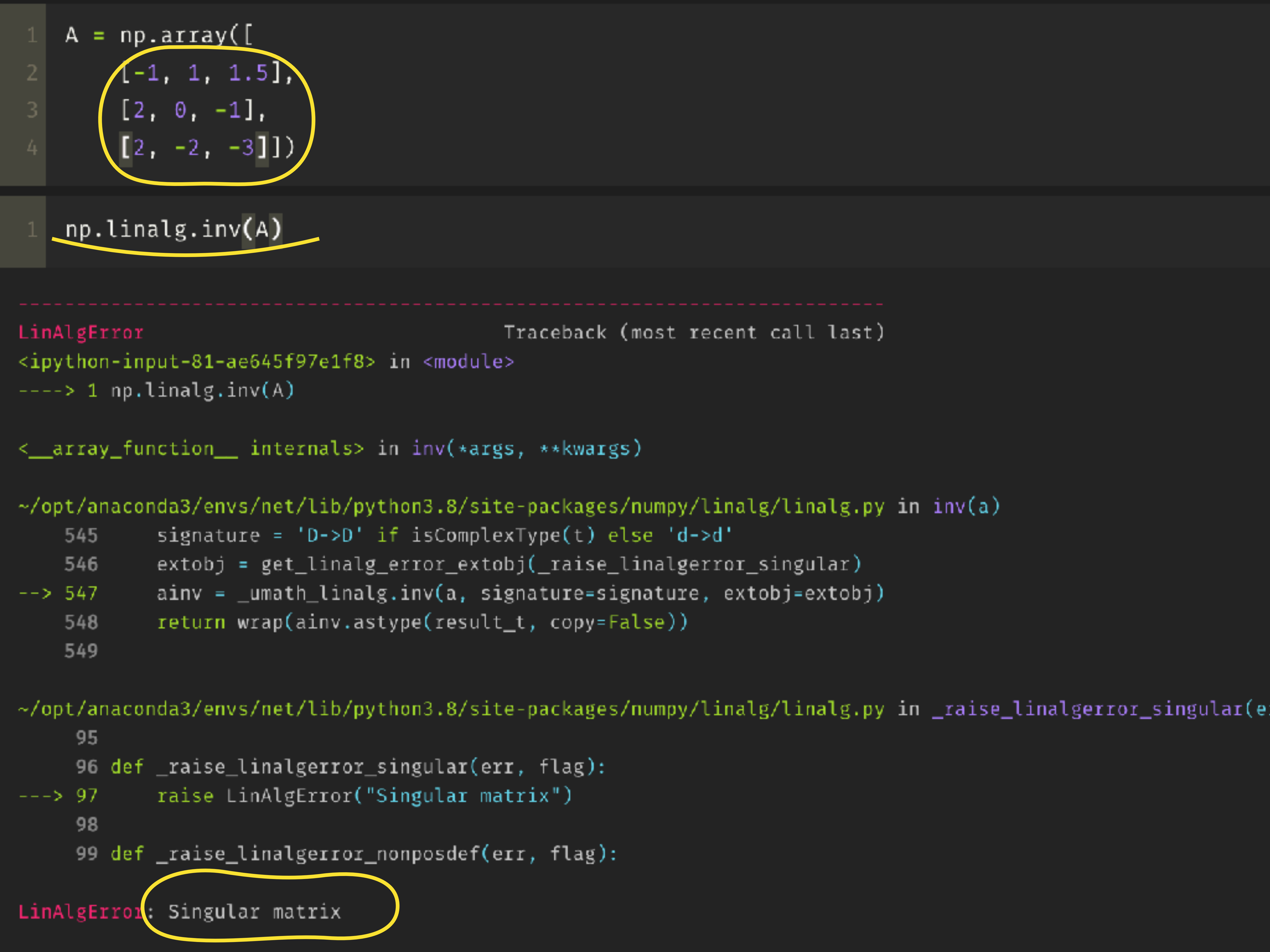
A has left inverse

if $n = m$, $A\,A' = I = A'\,A$



square matrix A is *invertible*,
if A^{-1} exists and $A\,A^{-1} = A^{-1}\,A = I$

if A^{-1} does not exist, A is called a *singular* matrix



```
A = np.array([
    [-1, 1, 1.5],
    [2, 0, -1],
    [2, -2, -3]])
```

```
np.linalg.inv(A)
```

```
---------------------------------------------------------------------------
LinAlgError                               Traceback (most recent call last)
<ipython-input-81-ae645f97e1f8> in <module>
----> 1 np.linalg.inv(A)

<__array_function__ internals> in inv(*args, **kwargs)

~/opt/anaconda3/envs/net/lib/python3.8/site-packages/numpy/linalg/linalg.py in inv(a)
    545     signature = 'D->D' if isComplexType(t) else 'd->d'
    546     extobj = get_linalg_error_extobj(_raise_linalgerror_singular)
--> 547     ainv = _umath_linalg.inv(a, signature=signature, extobj=extobj)
    548     return wrap(ainv.astype(result_t, copy=False))
    549

~/opt/anaconda3/envs/net/lib/python3.8/site-packages/numpy/linalg/linalg.py in _raise_linalgerror_singular(e
     95
     96 def _raise_linalgerror_singular(err, flag):
---> 97     raise LinAlgError("Singular matrix")
     98
     99 def _raise_linalgerror_nonposdef(err, flag):

LinAlgError: Singular matrix
```

Some properties

$(A^{-1})^{-1} = A$

$(A^{-1})^{-1} A^{-1} = (B)^{-1} B = I$

$(A^{-1})^{-1} \boxed{A^{-1} \underline{A}} = I \underline{A}$

$(A^{-1})^{-1} I = A$

$\underline{(A^{-1})^{-1} = A}$

$I^{-1} = I$

$I^{-1} I = I$ *- because I^{-1} is inverse of I*

$I^{-1} I = I^{-1}$ *- because I is identity matrix*

$\underline{I^{-1} = I}$