s71

March 1, 2022

1 25 reikningar með numpy

1.1 setup

[ ]:
```
import numpy as np
import numpy.linalg as la
from math import pow

a = np.array([ 3, 2, 8])
b = np.array([ 2, 0, 1])
c = np.array([-2, 1, 5])

A = np.array([
    [2,4,7],
    [3,4,8],
    [4,6,9]
])

B = np.array([
    [5,5,5],
    [6,6,6],
    [7,8,9]
])
```

1.2 1

1.2.1 a)

[ ]:
```
aOut = (a * b) - ((c * (B * b)) / (3 * la.norm(c)))
print(aOut)
```

```
[[7.21716124 0.         6.47854845]
 [7.46059349 0.         6.17425814]
 [7.70402573 0.         5.26138721]]
```

1.2.2 b)

[ ]:
```
bOut = 2 * ((A.T) + (B ** 2)) * (a - (2 * b))
print(bOut)
```

```
[[ -54  112  348]
 [ -80  160  504]
 [-112  288 1080]]
```

1.2.3 c)

[ ]:
```
cOut1 = la.det(A)
cOut2 = 1 / A # x ¹ == 1/x

print(cOut1)
print(cOut2)
```

```
10.000000000000002
[[0.5        0.25       0.14285714]
 [0.33333333 0.25       0.125     ]
 [0.25       0.16666667 0.11111111]]
```

1.2.4 d)

[ ]:
```
# hneppi = np.array([A,1])
reiknad = la.solve(A,b)
print(reiknad)
```

```
[-2.   1.5  0. ]
```

1.2.5 e)

[ ]:
```
# 10x + 5y = 170
# x + y = 23

hneppi = np.array([[10,5],[1,1]])
utkoma = np.array([170,23])
reiknad = la.solve(hneppi,utkoma)
print(reiknad)
```

```
[11. 12.]
```

1.3 2

[ ]:
```
def jafna(n):
    A = np.zeros((n,n))
    b = np.zeros(n)
    for i in range(n-1):
```

```
        A[,(j,i)] = [1,(i-j)]

out = jafna(4)
print(out)
```

```
---------------------------------------------------------------------------
LinAlgError                               Traceback (most recent call last)
/home/tumi/Documents/reikni/s7/s7.ipynb Cell 16' in <module>
      <a href='vscode-notebook-cell:/home/tumi/Documents/reikni/s7/s7.
 ↪ipynb#ch0000019?line=6'>7</a>              utkoma = np.array([2])
      <a href='vscode-notebook-cell:/home/tumi/Documents/reikni/s7/s7.
 ↪ipynb#ch0000019?line=7'>8</a>              print(la.solve(hneppi,utkoma))
----> <a href='vscode-notebook-cell:/home/tumi/Documents/reikni/s7/s7.
 ↪ipynb#ch0000019?line=8'>9</a> out = jafna(4)
     <a href='vscode-notebook-cell:/home/tumi/Documents/reikni/s7/s7.
 ↪ipynb#ch0000019?line=9'>10</a> print(out)

/home/tumi/Documents/reikni/s7/s7.ipynb Cell 16' in jafna(n)
      <a href='vscode-notebook-cell:/home/tumi/Documents/reikni/s7/s7.
 ↪ipynb#ch0000019?line=5'>6</a> hneppi = np.array([i-j,1])
      <a href='vscode-notebook-cell:/home/tumi/Documents/reikni/s7/s7.
 ↪ipynb#ch0000019?line=6'>7</a> utkoma = np.array([2])
----> <a href='vscode-notebook-cell:/home/tumi/Documents/reikni/s7/s7.
 ↪ipynb#ch0000019?line=7'>8</a> print(la.solve(hneppi,utkoma))

File <__array_function__ internals>:180, in solve(*args, **kwargs)

File ~/.local/lib/python3.10/site-packages/numpy/linalg/linalg.py:379, in␣
 ↪solve(a, b)
    <a href='file:///~/.local/lib/python3.10/site-packages/numpy/linalg/linalg.
 ↪py?line=314'>315</a> """
    <a href='file:///~/.local/lib/python3.10/site-packages/numpy/linalg/linalg.
 ↪py?line=315'>316</a> Solve a linear matrix equation, or system of linear␣
 ↪scalar equations.
    <a href='file:///~/.local/lib/python3.10/site-packages/numpy/linalg/linalg.
 ↪py?line=316'>317</a>
   (…)
    <a href='file:///~/.local/lib/python3.10/site-packages/numpy/linalg/linalg.
 ↪py?line=375'>376</a>
    <a href='file:///~/.local/lib/python3.10/site-packages/numpy/linalg/linalg.
 ↪py?line=376'>377</a> """
    <a href='file:///~/.local/lib/python3.10/site-packages/numpy/linalg/linalg.
 ↪py?line=377'>378</a> a, _ = _makearray(a)
--> <a href='file:///~/.local/lib/python3.10/site-packages/numpy/linalg/linalg.
 ↪py?line=378'>379</a> _assert_stacked_2d(a)
    <a href='file:///~/.local/lib/python3.10/site-packages/numpy/linalg/linalg.
 ↪py?line=379'>380</a> _assert_stacked_square(a)
```

```
    <a href='file:///~/.local/lib/python3.10/site-packages/numpy/linalg/linalg.
 ↪py?line=380'>381</a> b, wrap = _makearray(b)

File ~/.local/lib/python3.10/site-packages/numpy/linalg/linalg.py:196, in␣
 ↪_assert_stacked_2d(*arrays)
    <a href='file:///~/.local/lib/python3.10/site-packages/numpy/linalg/linalg.
 ↪py?line=193'>194</a> for a in arrays:
    <a href='file:///~/.local/lib/python3.10/site-packages/numpy/linalg/linalg.
 ↪py?line=194'>195</a>     if a.ndim < 2:
--> <a href='file:///~/.local/lib/python3.10/site-packages/numpy/linalg/linalg.
 ↪py?line=195'>196</a>         raise LinAlgError('%d-dimensional array given.␣
 ↪Array must be '
    <a href='file:///~/.local/lib/python3.10/site-packages/numpy/linalg/linalg.
 ↪py?line=196'>197</a>                 'at least two-dimensional' % a.ndim)

LinAlgError: 1-dimensional array given. Array must be at least two-dimensional
```