

# Kurs i NumPy med NumFys

Thorvald Ballestad

January 25, 2020

- 1 Intro og recap
- 2 To-dimensjonale arrays
- 3 Plotting i 2D
- 4 Lese fra fil
- 5 Ressurser

# Hvorfor NumPy

- Gjør det enklere å jobbe med numerikk i Python

```
1 my_numpy_list*2 // [1,2,4] -> [2,4,8]
2 my_python_list*2 // [1,2,4] ->
  [1,2,4,1,2,4]
```

- Går *mye* raskere

## Litt recap

Dette burde (i noen grad) være kjent fra før:  
NumPy-array > Python-liste.

- `np.linspace ( start ,end,num_points)`
- `np.arange( start , stop , step)`

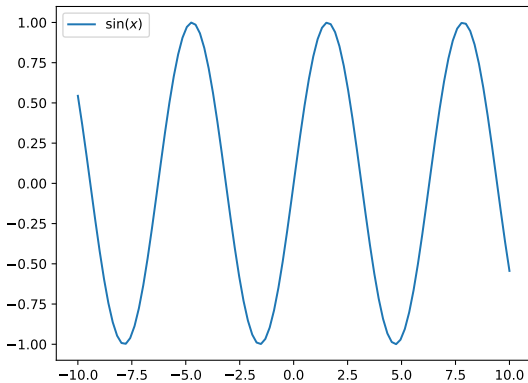
Operasjoner fungerer slik de gjør i matte:

```
1 x = np.array([1,2,3])
2 my_func(x) # Returnerer en array der
  my_func har blitt evaluert for hvert
  element i x
```

## Oppgave 1 – The basics

Plotting: Vi plotter ved å gi matplotlib to lister, en med  $x$ -verdier og en med  $y$ -verdier.

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 x = np.linspace(-10, 10, 100) # 100 points
   from -10 to 10
5 y = np.sin(x)
6
7 plt.plot(x, y, label='sin(x)')
8 plt.legend()
9 plt.show()
```



## Oppgave 2 – Plotting



# To-dimensjonale arrays i NumPy (matriser)

NumPy arrays kan ha flere dimensjoner.

```
1 data = np.array([[1, 2], [3, 4]])  
2 # [[1, 2],  
3 #   [3, 4]]
```

Funker ellers likt som vanlige arrays.

# To-dimensjonale arrays i NumPy (matriser)

Slicing – å få ut dataen du vil ha

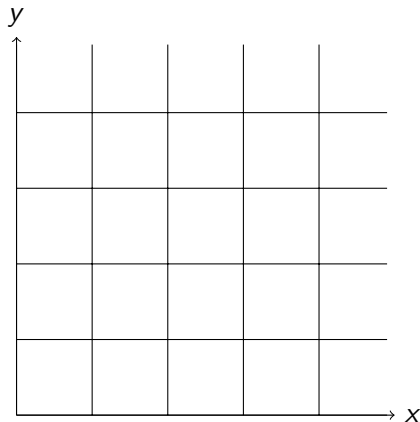
```
1 data = np.array([[1, 2, 3],  
2                 [4, 5, 6]])  
3 data[0]      # First row  
4 data[0, :]   # Also first row  
5 data[:, 0]   # First column
```

## Oppgave 3 – Matriser

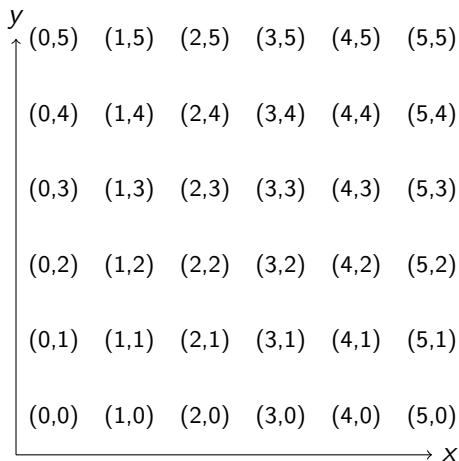
# Meshgrid

## Meshgrid

```
1 x = np.linspace(-4, 4,  
2   100);  
3 y = np.linspace(-4, 4,  
4   100);  
5 xx, yy = np.meshgrid(x, y);
```

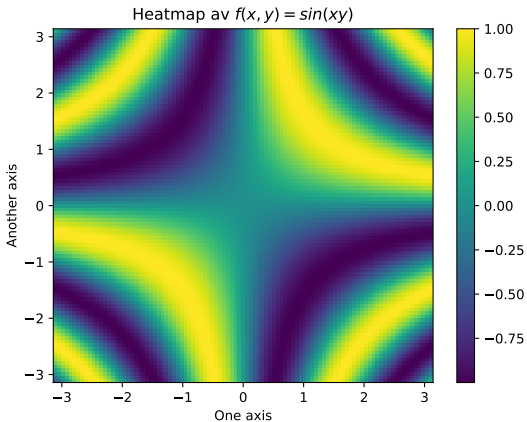


# Meshgrid



## Oppgave 4 – Stepping it up

# Plotting i 2D



# Plotting i 2D

## 2D

- pcolormesh
- quiver
- contour
- streamplot

## 3D

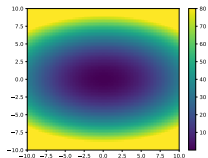
- plot\_surface
- contour
- plot (med x,y,z)
- [matplotlib.org/mpl\\_toolkits/mplot3d/tutorial.html](http://matplotlib.org/mpl_toolkits/mplot3d/tutorial.html) for liste.



# pcolormesh

Heatmap som indikerer verdien til en tovariabel skalar funksjon.

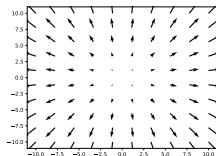
```
1 x = np.linspace(-10, 10, 100)
2 y = np.linspace(-10, 10, 100)
3 xx,yy = np.meshgrid(x,y)
4
5 plt.pcolormesh(xx,yy,xx**2/3 + yy**2)
6 plt.colorbar()
```



# quiver

Indikerer størrelse og retning på en tovariabel vektorfunksjon.

```
1 x = np.linspace(-10, 10, 10)
2 y = np.linspace(-10, 10, 10)
3 xx, yy = np.meshgrid(x, y)
4
5 plt.quiver(xx, yy, xx, yy)
```



## Oppgave 5 – Plotting i 2D

## Lese fra fil

Å lese data fra fil krever litt jobb for å legge inn dataen på riktig måte i python-variabler. NumPy har en nyttig funksjon, `loadtxt`, som gjør dette enkelt.

```
1 # Assume my_data.txt has format
2 # second temperature wind_speed
3
4 time, temp, wind = np.loadtxt('my_data.txt',
5                               unpack=True)
6 # Notice the unpack=True option
```

## Nyttige ressurser

- [www.numfys.net](http://www.numfys.net)  
Inneholder mange notebooks på alle nivåer.
  - <http://numfys.net/kurs>
  - [http://nbviewer.jupyter.org/urls/www.numfys.net/media/notebooks/intermediate\\_plotting\\_final.ipynb](http://nbviewer.jupyter.org/urls/www.numfys.net/media/notebooks/intermediate_plotting_final.ipynb)
  - <http://nbviewer.jupyter.org/urls/www.numfys.net/media/notebooks/NumpyIntermediate.ipynb>
- <http://numpy.org/devdocs> (like greit å søke etter den funksjonen du lurer på, NumPy docs kommer ofte høyt i resultatlista)