ME2 Computing

Graphics and
Vector analysis

Representing and Plotting numerical functions

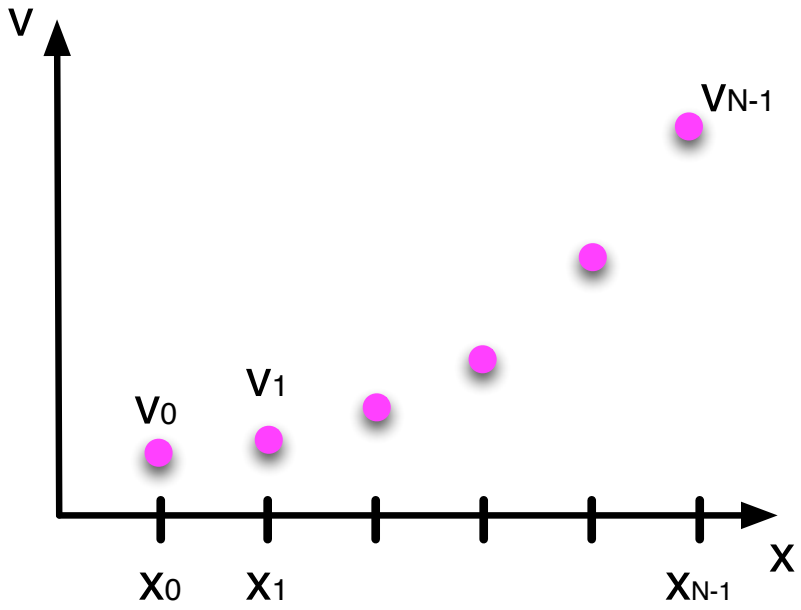
Scalar 1D

$v = f(x)$

x	1	2	3	4	5	6	7
v	4	2	7	1	-1	4	3

length N

1D arrays

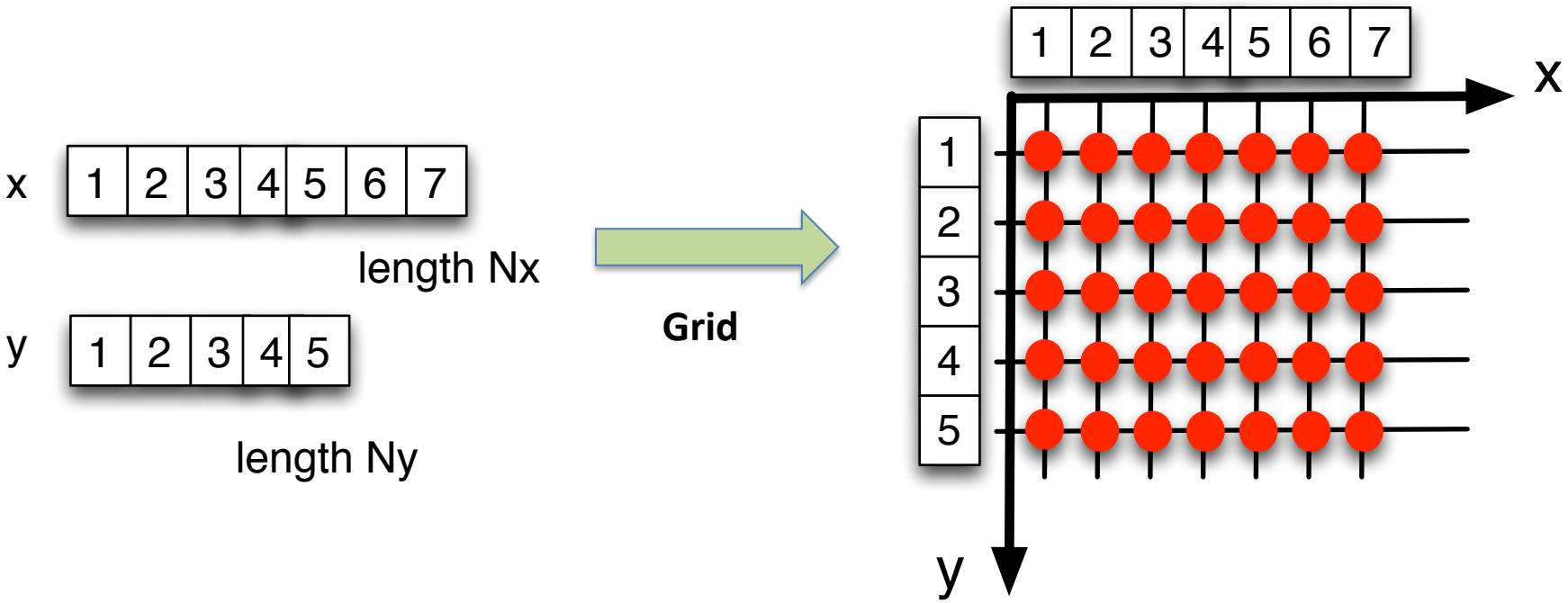


plot(x,y)
scatter(x,y)
bar(x,y)

Representing and Plotting numerical functions

Scalar 2D

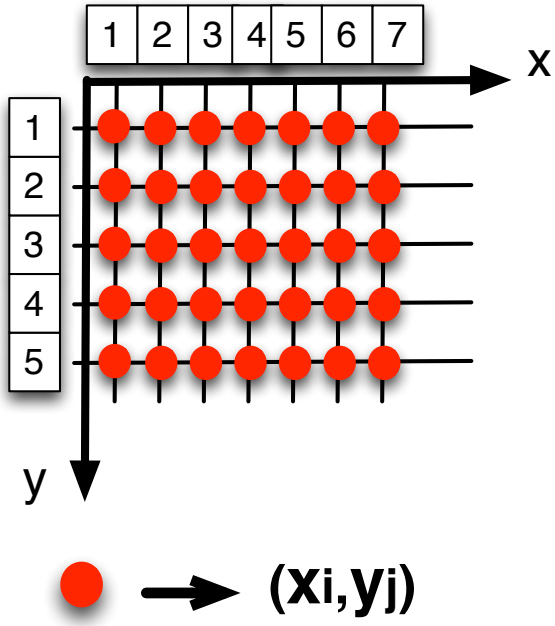
$v = f(x, y)$



Representing and Plotting numerical functions

Scalar 2D

Grid



Grid: two 2D matrices

Xg , Yg = np.meshgrid(x,y)

All x values

Xg

1	2	3	4	5	6	7
1	2	3	4	5	6	7
1	2	3	4	5	6	7
1	2	3	4	5	6	7
1	2	3	4	5	6	7

All y values

Yg

1	1	1	1	1	1	1
2	2	2	2	2	2	2
3	3	3	3	3	3	3
4	4	4	4	4	4	4
5	5	5	5	5	5	5

Size Ny by Nx

Representing and Plotting numerical functions

Scalar 2D

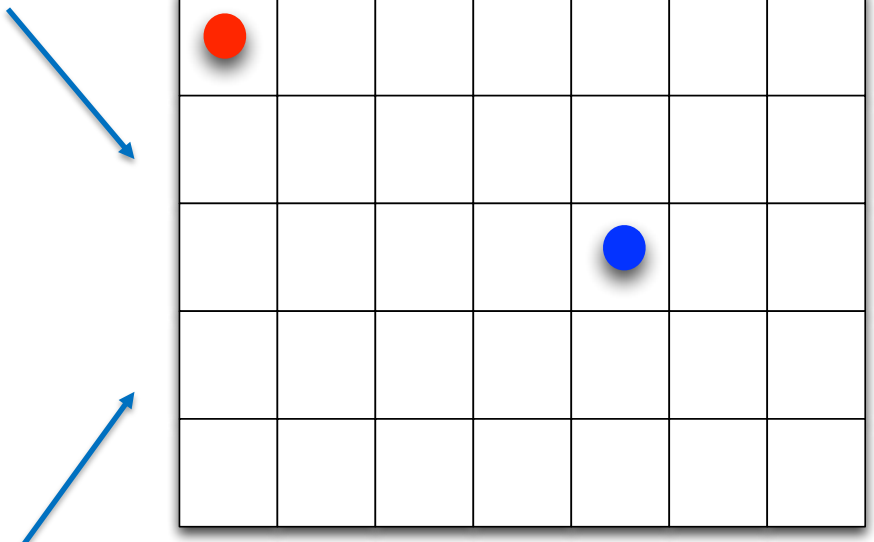
$v = f(x, y)$

Xg

1	2	3	4	5	6	7
1	2	3	4	5	6	7
1	2	3	4	5	6	7
1	2	3	4	5	6	7
1	2	3	4	5	6	7

Yg

1	1	1	1	1	1	1
2	2	2	2	2	2	2
3	3	3	3	3	3	3
4	4	4	4	4	4	4
5	5	5	5	5	5	5



All sizes Ny by Nx

Representing and Plotting numerical functions

Scalar 2D

$$v = f(x, y) = x^2 + y^2$$

x | 1 | 2 | 3 | 4 | 5 | 6 | 7 |

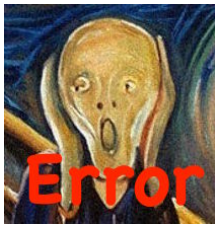
length Nx

y | 1 | 2 | 3 | 4 | 5 |

length Ny

Grid vs non-grid

```
V = np.ndarray(Ny,Nx)
V = x**2 + y**2
```



With Grid

```
V = np.ndarray(Ny,Nx)
V = Xg**2 + Yg**2
```

Without Grid

```
V = np.ndarray(Ny,Nx)
for i in range(0,Ny):
    for j in range (0,Nx):
        V[i,j] = x[j]**2 + y[i]**2
```

Representing and Plotting numerical functions

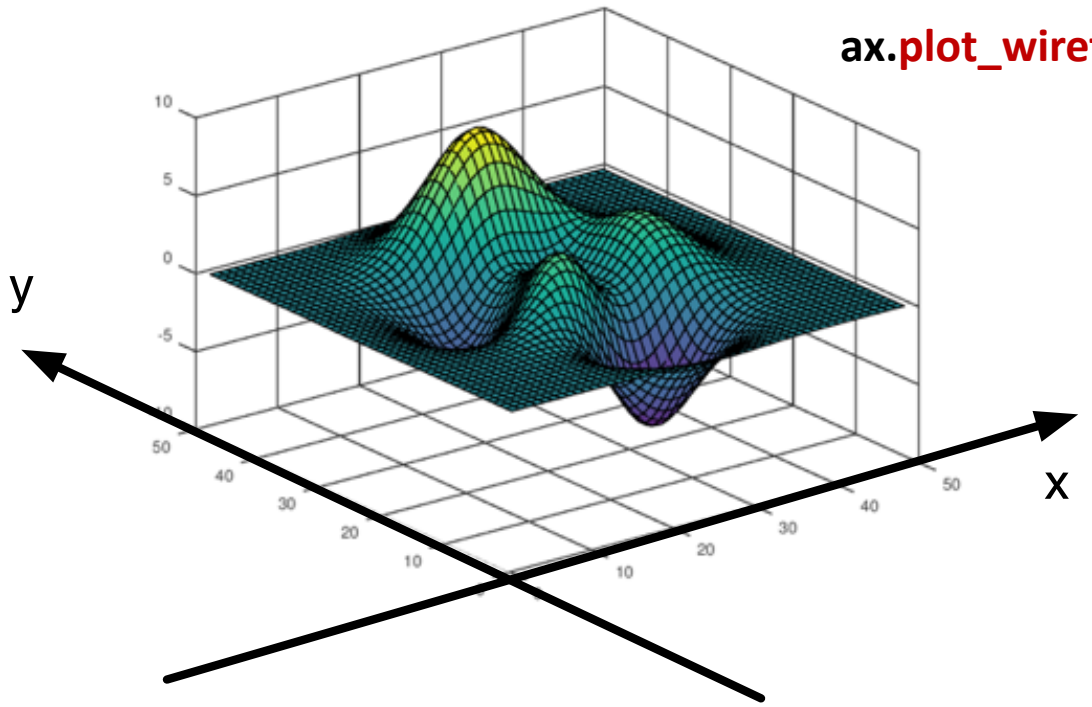
Scalar 2D

$$v = f(x, y)$$

Surface plot

```
from mpl_toolkits import mplot3d
ax = pl.axes(projection='3d')
ax.plot_surface(Xg,Yg,V)

ax.plot_wireframe(Xg,Yg,V)
```



Representing and Plotting numerical functions

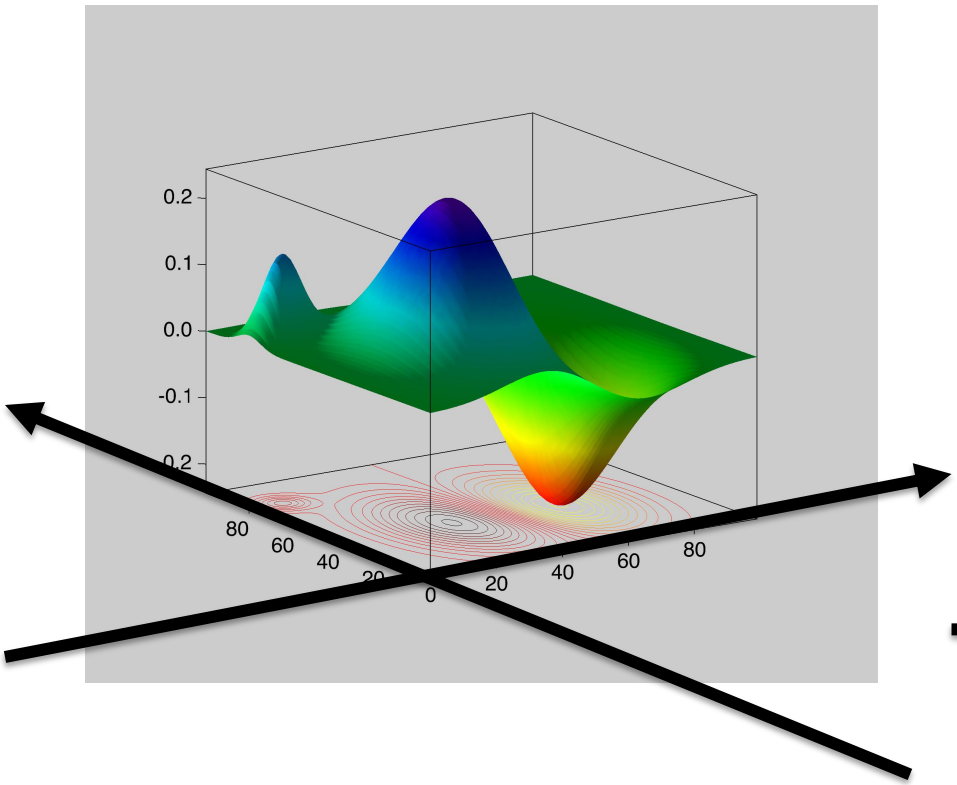
Scalar 2D

$$v = f(x, y)$$

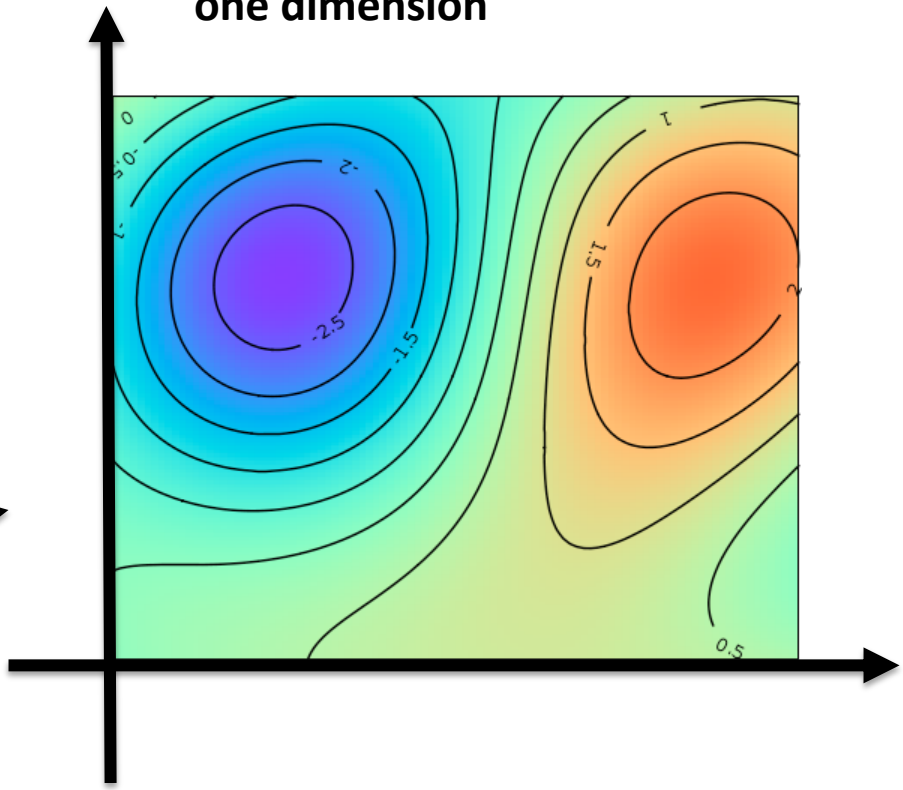
Contour plot

pl.contour(Xg,Yg,V)

pl.contour(x,v,V)



Scales down the plot by one dimension



Representing and Plotting numerical functions

Scalar multidimensional

$v = f(x, y, z)$ $v = f(x, y, t)$

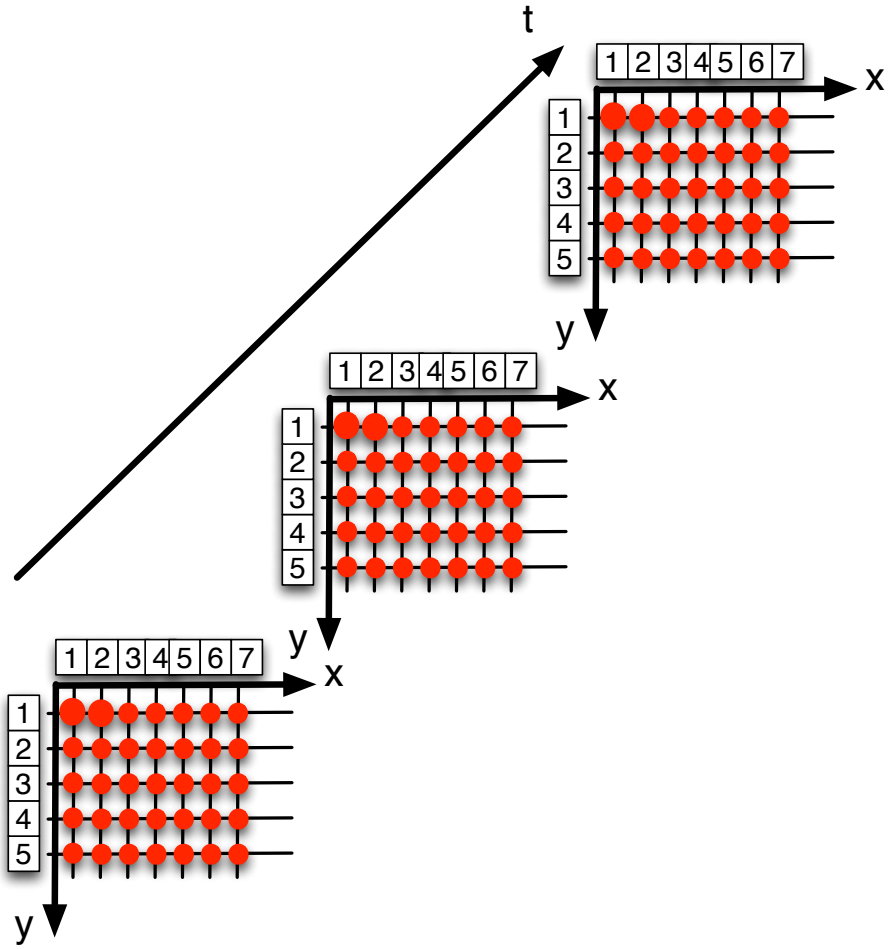
$v = f(x, y, z, t)$

x | 1 | 2 | 3 | 4 | 5 | 6 | 7 | length Nx

y | 1 | 2 | 3 | 4 | 5 | length Ny

t | 1 | 2 | 3 | length Nt

$v = f(x, y, t)$



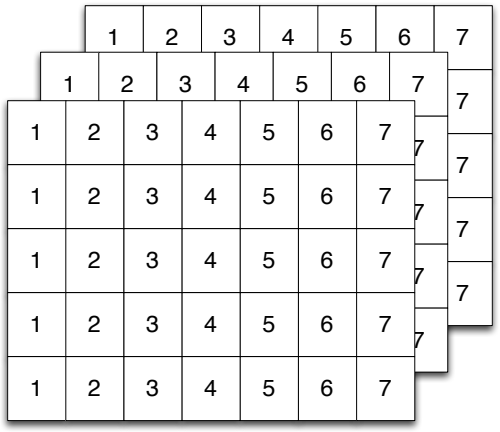
Representing and Plotting numerical functions

Scalar multidimensional

$$v = f(x, y, t)$$

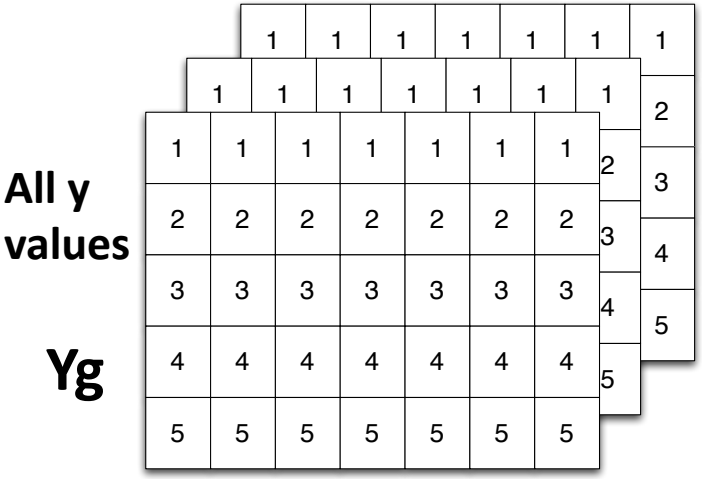
All x values

Xg



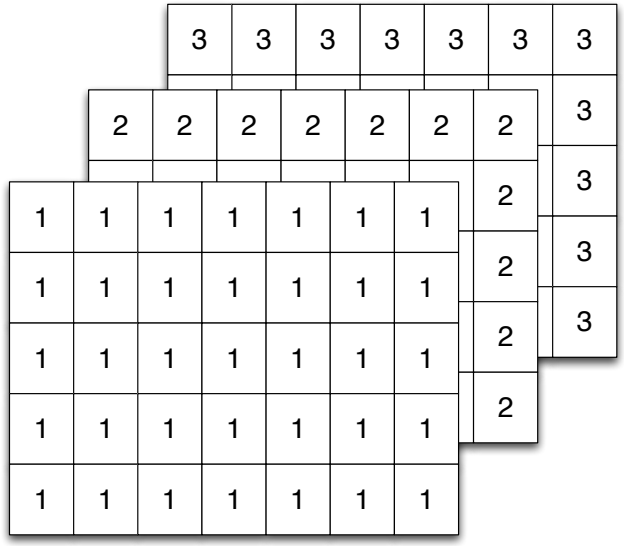
All y values

Yg



All t values

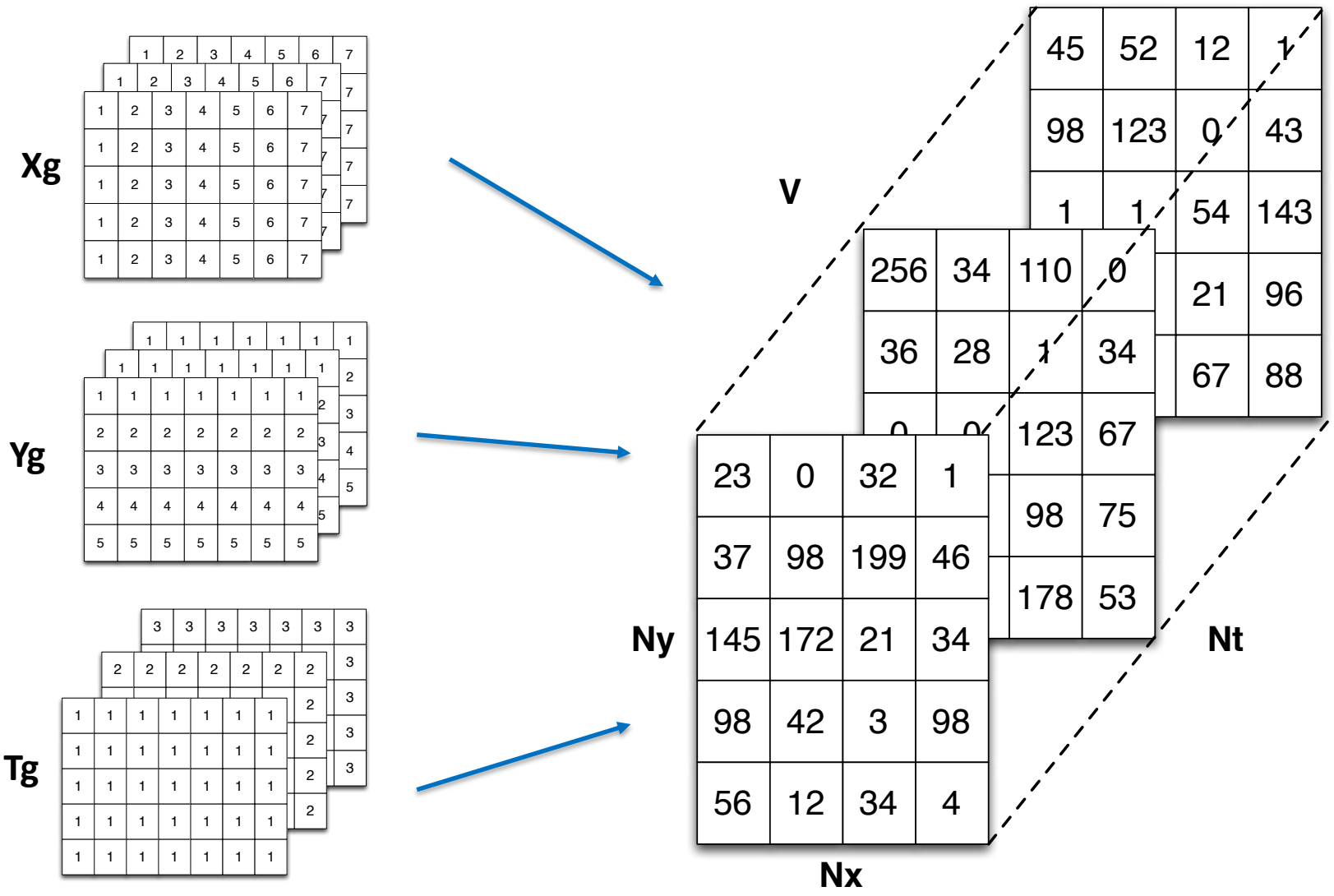
Tg



Representing and Plotting numerical functions

Scalar multidimensional

$$v = f(x, y, t)$$



Representing and Plotting numerical functions

Scalar multidimensional

$$v = f(x, y, t)$$

Frames

23	0	32	1
37	98	199	46
145	172	21	34
98	42	3	98
56	12	34	4

t = 1

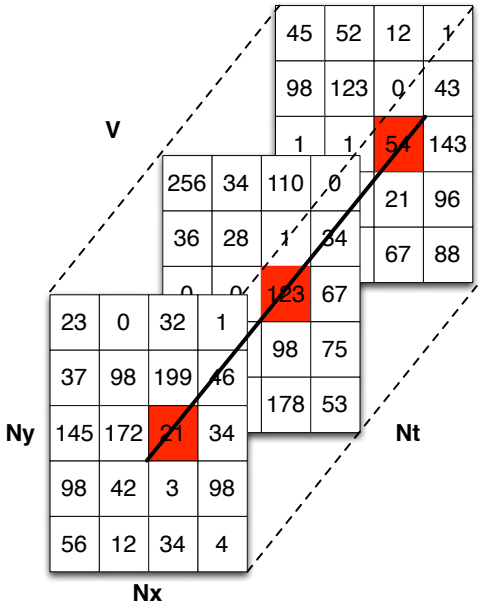
256	34	110	0
36	28	1	34
0	0	123	67
1	1	98	75
0	0	178	53

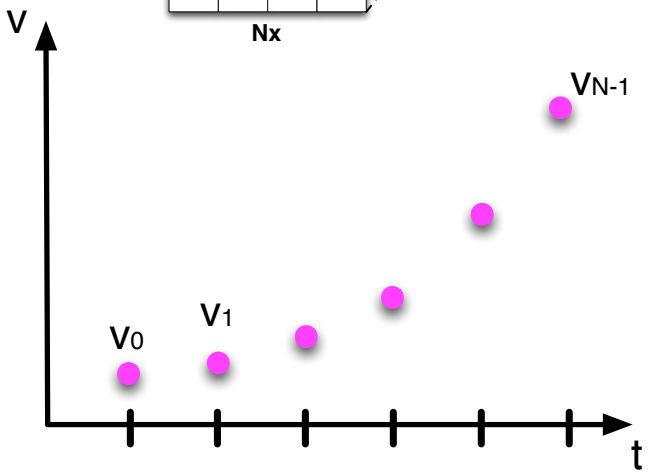
t = 2

45	52	12	1
98	123	0	43
1	1	54	143
0	0	21	96
1	1	67	88

t = 3

Time evolution





Representing and Plotting numerical functions

Vectors (of one variable)

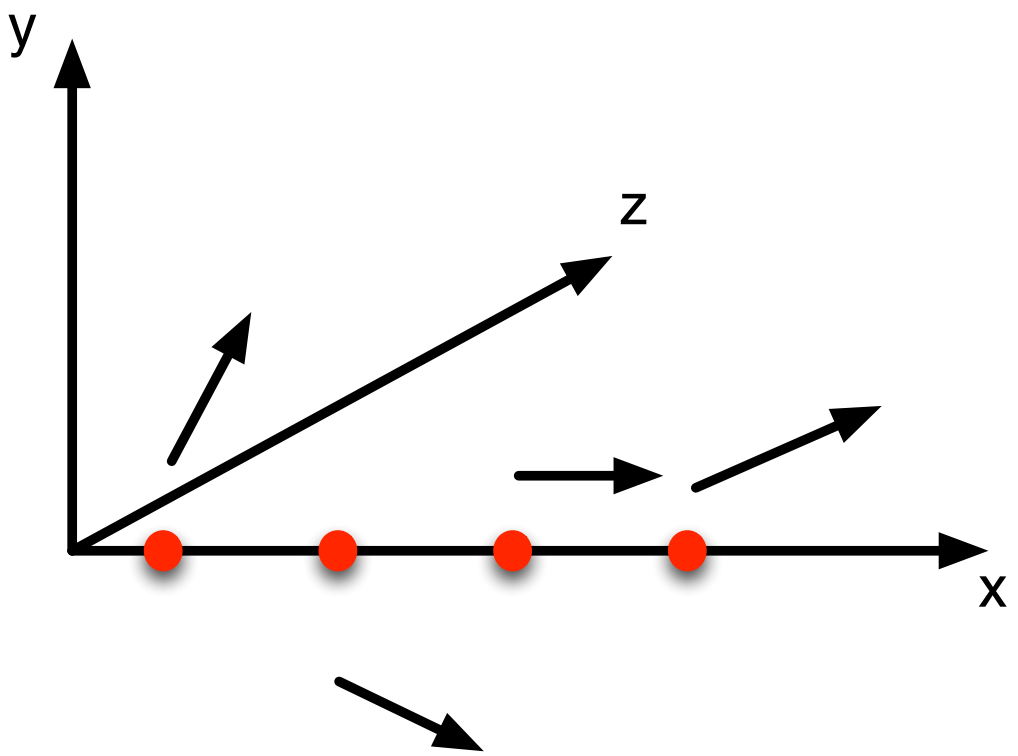
$$\boldsymbol{v} = \boldsymbol{f}(x) = f_1(x)\boldsymbol{i} + f_2(x)\boldsymbol{j} + f_3(x)\boldsymbol{k}$$

x | 1 | 2 | 3 | 4 | 5 | 6 | 7 | length Nx

f

f1							
f2							
f3							

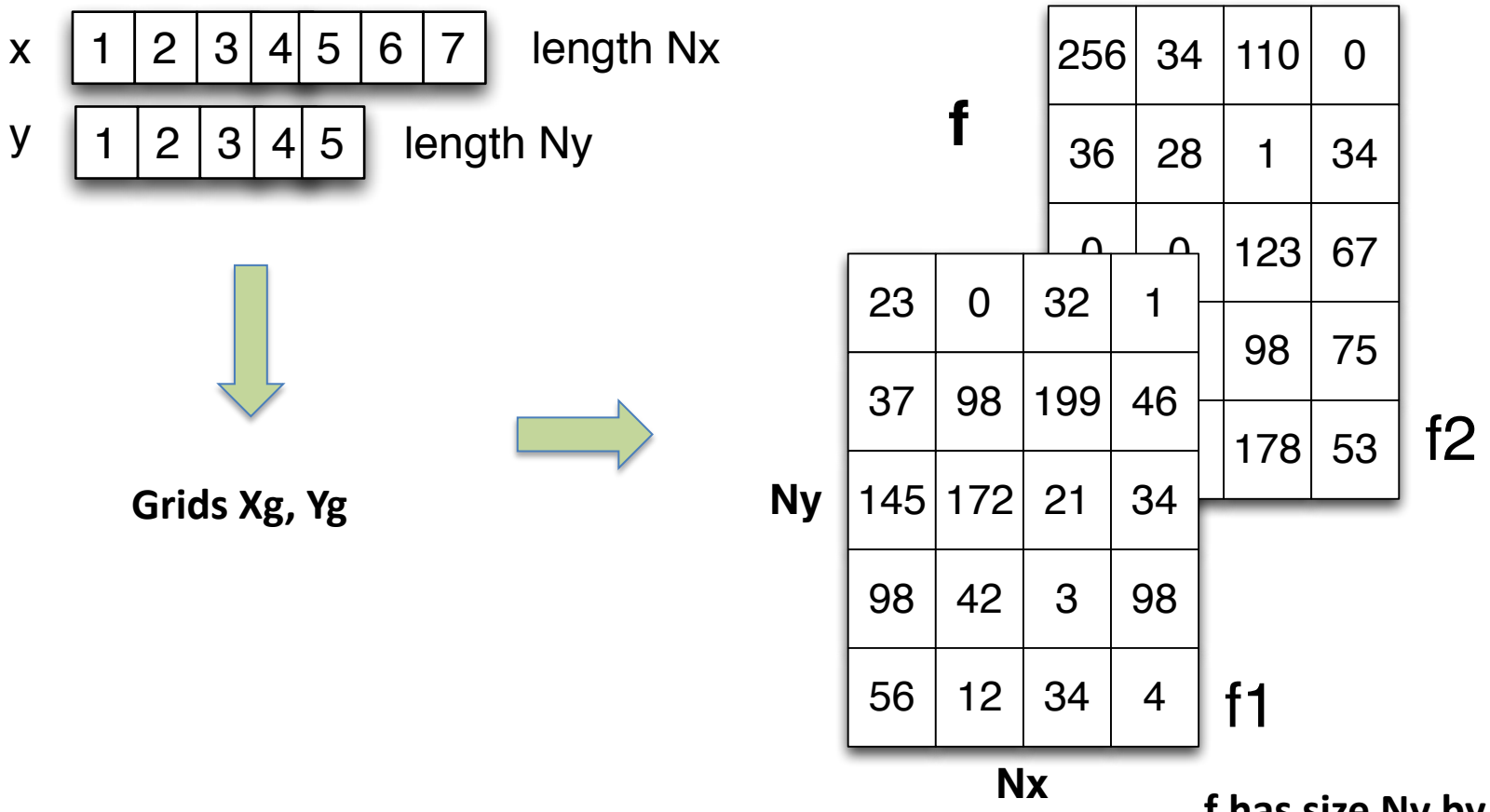
Nx by 3



Representing and Plotting numerical functions

Vectors (of two variables)

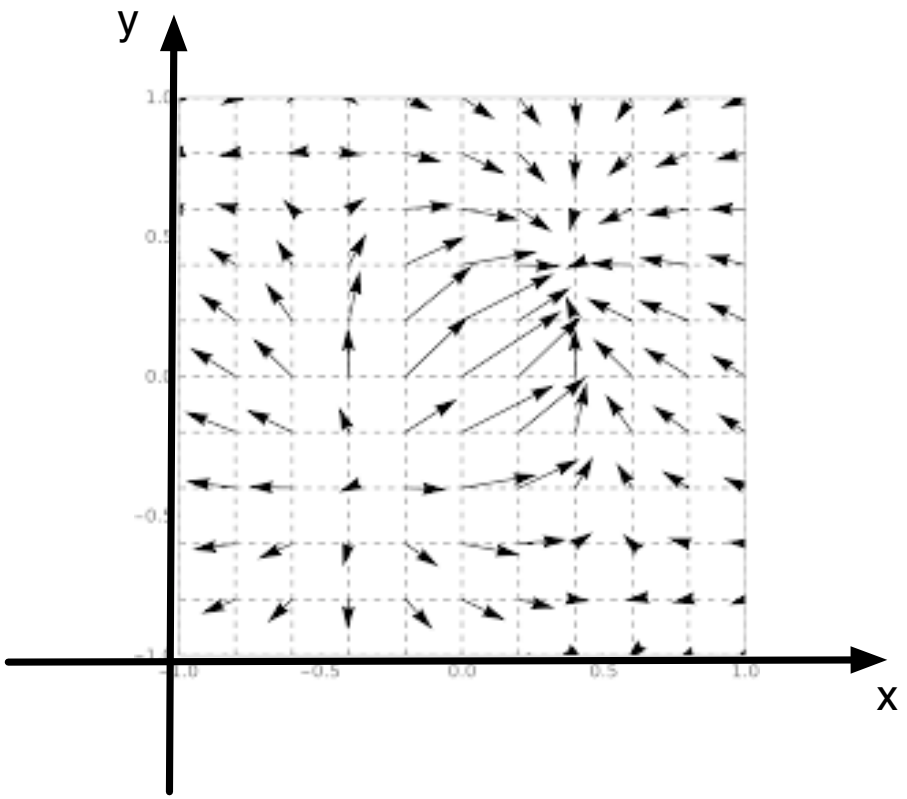
$$\boldsymbol{v} = \boldsymbol{f}(x, y) = f_1(x, y)\boldsymbol{i} + f_2(x, y)\boldsymbol{j}$$



Representing and Plotting numerical functions

Vectors (of two variables)

$$\boldsymbol{v} = \boldsymbol{f}(x, y) = f_1(x, y)\boldsymbol{i} + f_2(x, y)\boldsymbol{j}$$

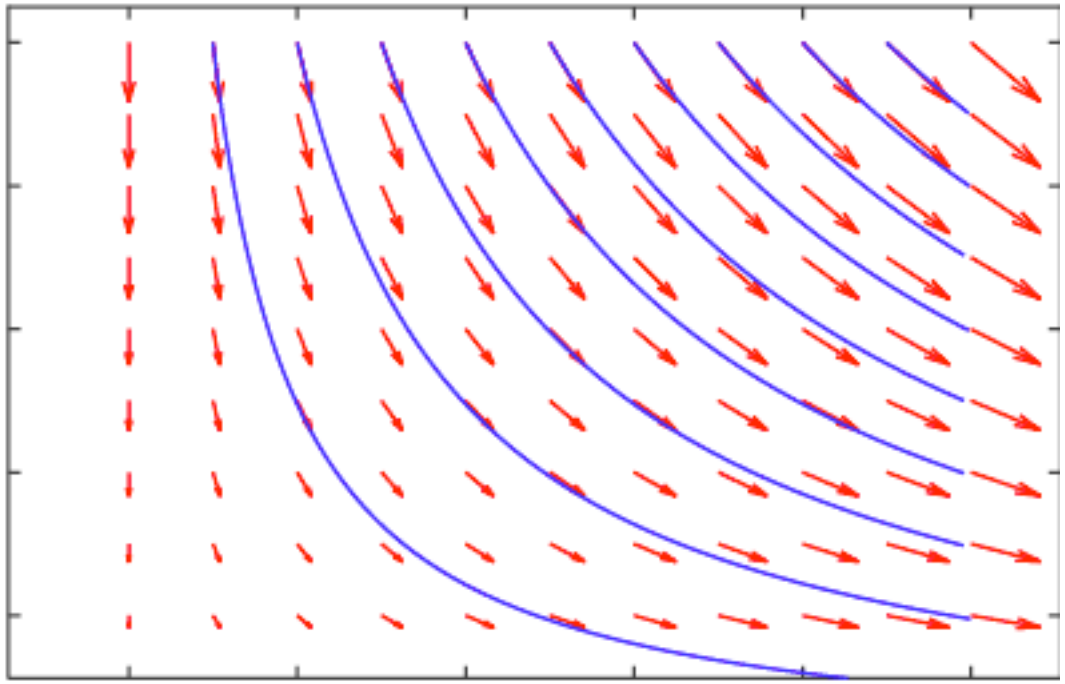


pl.quiver(Xg,Yg,f1,f2)

Representing and Plotting numerical functions

Vectors (of two variables)

$$\boldsymbol{v} = \boldsymbol{f}(x, y) = f_1(x, y)\boldsymbol{i} + f_2(x, y)\boldsymbol{j}$$

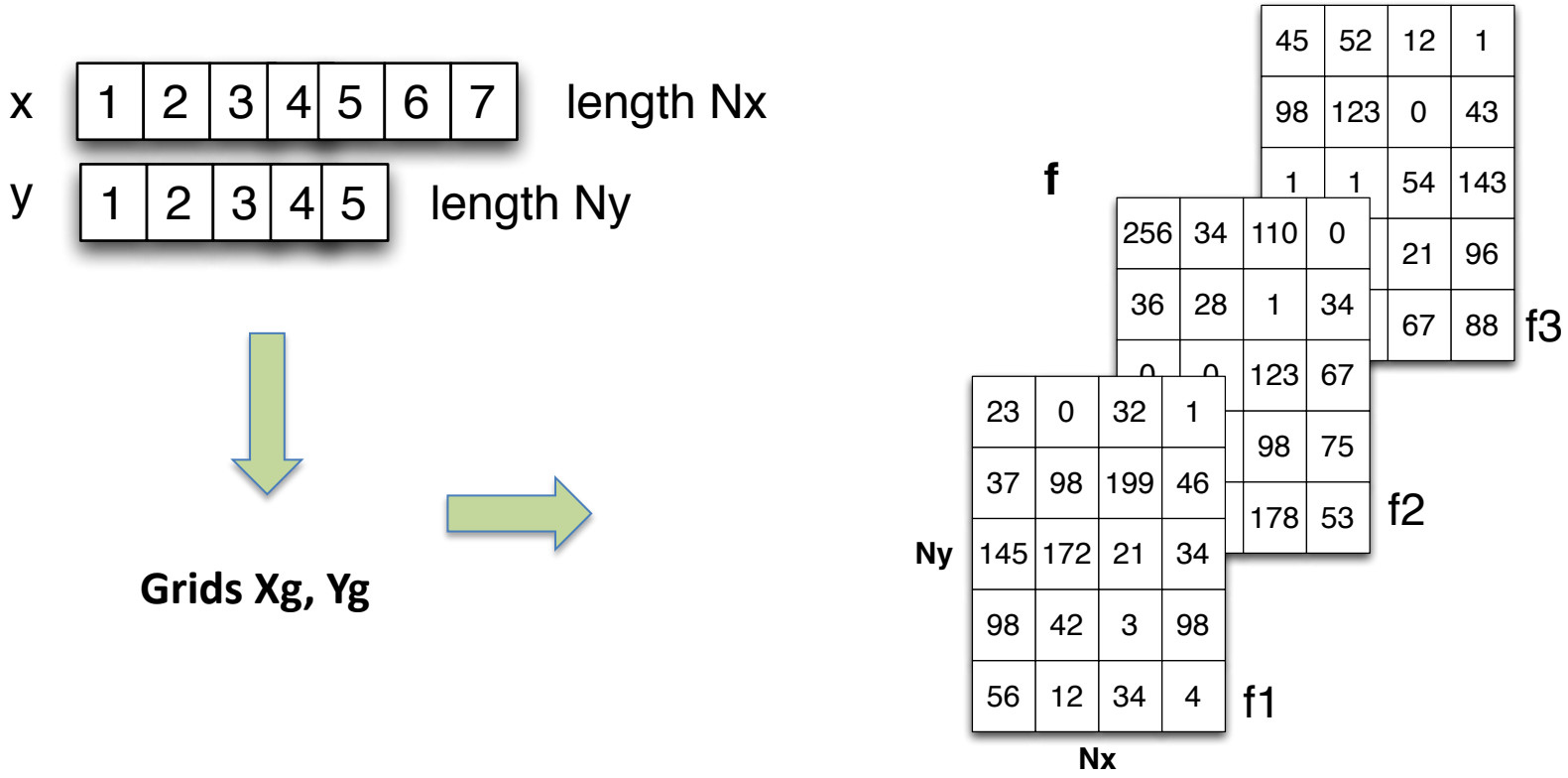


pl.streamplot(Xg,Yg,f1,f2)

Representing and Plotting numerical functions

Vectors (of two variables)

$$\boldsymbol{v} = \boldsymbol{f}(x, y) = f_1(x, y)\boldsymbol{i} + f_2(x, y)\boldsymbol{j} + f_3(x, y)\boldsymbol{k}$$

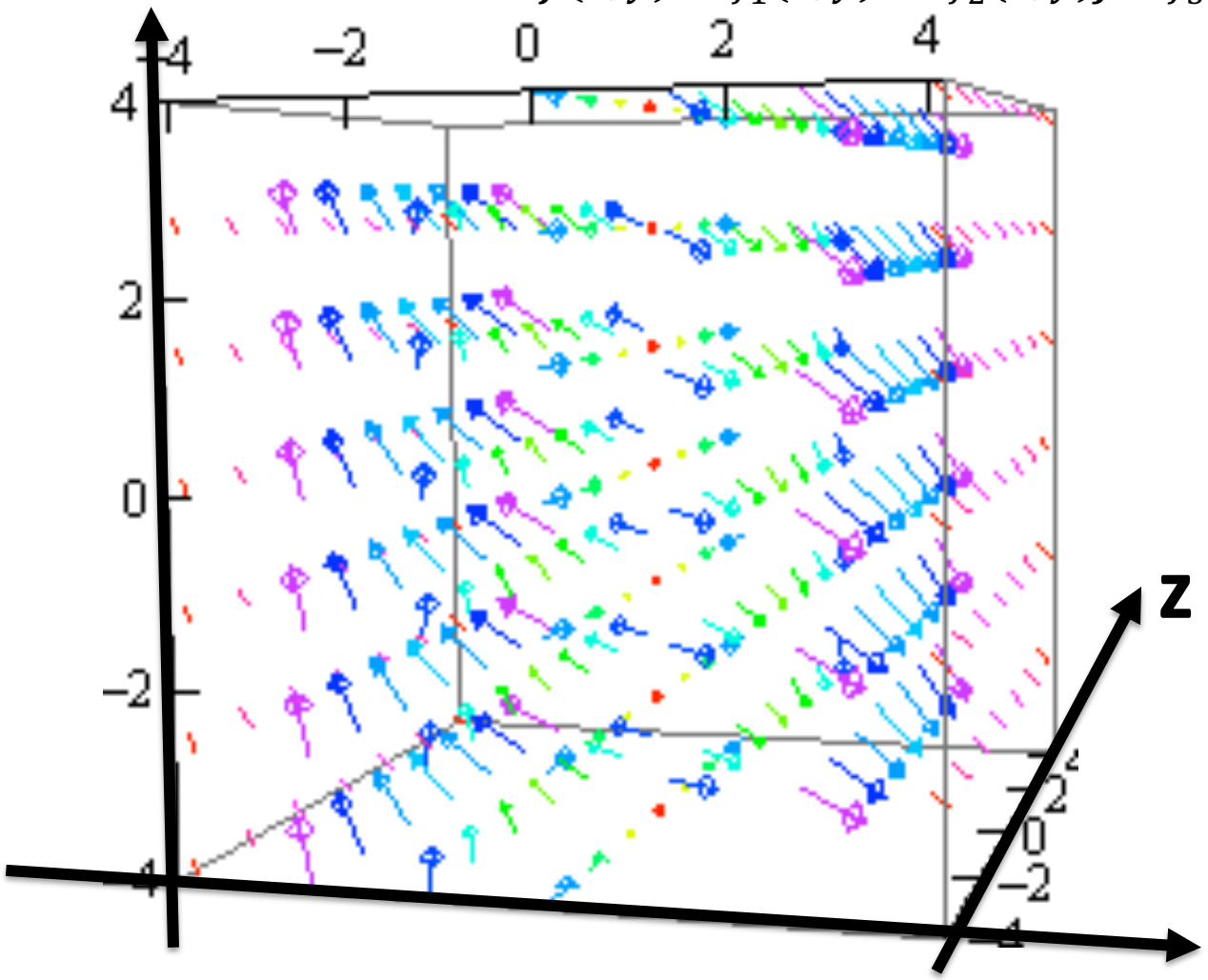


f has size Ny by Nx by 3

Representing and Plotting numerical functions

Vectors (of two variables)

$$\boldsymbol{v} = \boldsymbol{f}(x, y) = f_1(x, y)\boldsymbol{i} + f_2(x, y)\boldsymbol{j} + f_3(x, y)\boldsymbol{k}$$



Representing and Plotting numerical functions

Vectors (of three variables)

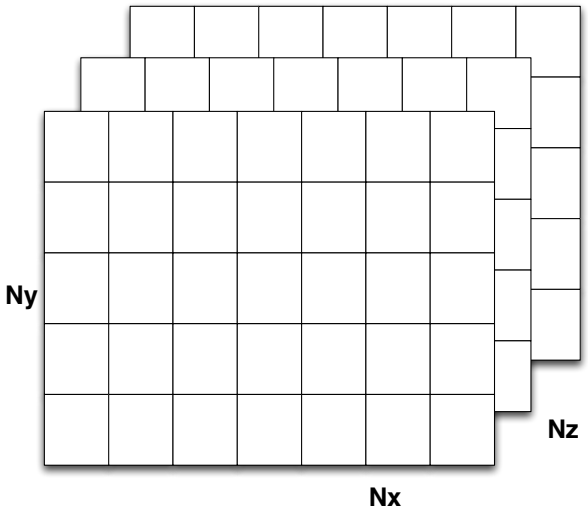
$$\boldsymbol{v} = \boldsymbol{f}(x, y, z) = f_1(x, y, z)\boldsymbol{i} + f_2(x, y, z)\boldsymbol{j} + f_3(x, y, z)\boldsymbol{k}$$

x | 1 | 2 | 3 | 4 | 5 | 6 | 7 | length Nx

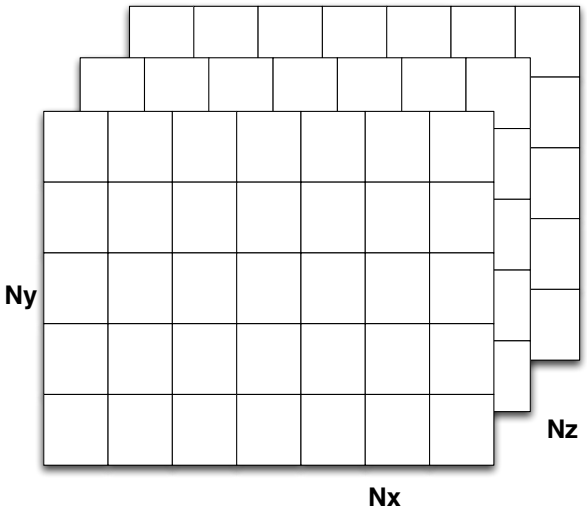
y | 1 | 2 | 3 | 4 | 5 | length Ny

z | 1 | 2 | 3 | length Nz

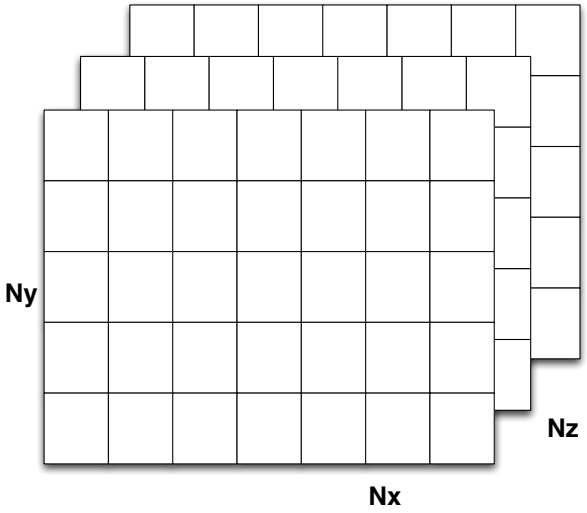
f1



f2



f3

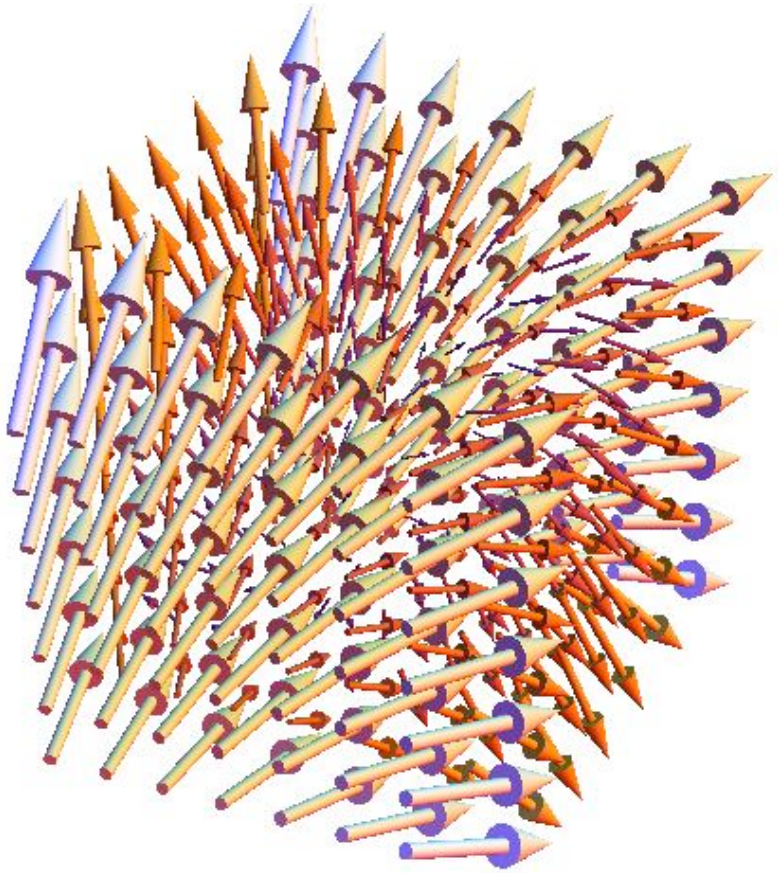


f has size Ny by Nx by Nz by 3

Representing and Plotting numerical functions

Vectors (of three variables)

$$\boldsymbol{v} = \boldsymbol{f}(x, y, z) = f_1(x, y, z)\boldsymbol{i} + f_2(x, y, z)\boldsymbol{j} + f_3(x, y, z)\boldsymbol{k}$$



Representing and Plotting numerical functions

Vectors (of three variables)

$$\boldsymbol{v} = \boldsymbol{f}(x, y, z) = f_1(x, y, z)\boldsymbol{i} + f_2(x, y, z)\boldsymbol{j} + f_3(x, y, z)\boldsymbol{k}$$