2D Animations

```
# %matplotlib inline
import numpy as np # numpy (arrays of numbers)
import matplotlib
import matplotlib.pyplot as plt # plotting with matplotlib

from matplotlib import animation, rc  # animations with matplotlib
from IPython.display import HTML # HTML and javascript support
```

```
fig, ax = plt.subplots() # assigning the figure and axes to 2 variables

ax.set_xlim(-8, 8) # limits on the x-axis
ax.set_ylim(-11, 11) # limits on the y-axis

# initializing the variable line with an empty plot:
# it will be updated in the following lines of code
print("ax.plot() returns a reference to the list of 2D lines")
line_0 = ax.plot([], [], linewidth=2)[0]
print("We have initialized our line object, saying that it's a 2D line")
```

```
print("Here we define the initialization function:")
def init():
    line_0.set_data([], []) # initialize 'line' so that it doesn't contain any points
    return [line_0] # animations need an iterable list of 2D lines. Here we use only one --> list of
```

```
print("Here we define the animation function, which depend on the evolution parameter 'i'")
def animate(i):
    x0 = -8
    m = i/100
    x = np.linspace(x0, x0 + i/10, 100)# 100 points between x0 and x0+i/100
    y = x + 5*np.sin(x - np.pi/2)
    line_0.set_data(x, y)
    return [line_0]
```

```
Here we define the animation function, which depend on the evolution parameter 'i'
```

```
frn = 180 # number of frames i=0,...,179
itv = 20 # number of milliseconds between one frame and the other
anim = animation.FuncAnimation(fig, animate, init_func=init, frames=frn, interval=20, blit=True)
```

```
jsanim = HTML(anim.to_jshtml())
display(jsanim)
```