MandelbrotzuWale

June 16, 2020

1 Das Apfelmännchen

1.1 Definition

Wir zeichnen zuerst das Apfelmännchen, auch **Mandelbrotmenge** genannt. Wir definieren die Mandelbrotmenge salopp als

> Die Menge aller Punkte $c \in \mathbb{C}$, so dass die komplexe Abbildung
>
> $$f(z) = z^2 + c$$
>
> mit Startwert $z_0 = 0$ nicht divergiert.

1.1.1 Beispielorbits

Bevor wir uns überlegen, wie man diese Menge berechnet, wollen wir ihre Form anschauen. Wir berechnen ein paar Orbits für bestimmte Startwerte c.

[1]:
```
%matplotlib inline
#%matplotlib notebook
%config InlineBackend.figure_format = 'pdf'

from numba import jit # just in time compiler: @jit

@jit
def mf(z,c) :
    return z*z+c
```

[2]:
```
@jit
def make_mfOrbit(c, anzahl_iterationen) :
    cOrbit = [0]
    for k in range(anzahl_iterationen):
        cOrbit.append(mf(cOrbit[k],c))
    return cOrbit
```

[3]:
```
make_mfOrbit(1,8)
```

[3]:
```
[0, 1, 2, 5, 26, 677, 458330, 210066388901, 3275921592928522330]
```

[4]:
```
make_mfOrbit(0.1,5)
```

[4]:
```
[0.0, 0.1, 0.11000000000000001, 0.1121, 0.11256641, 0.1126711966602881]
```

[5]:
```
make_mfOrbit(complex(0,1),10)
```

[5]:
```
[0j, 1j, (-1+1j), -1j, (-1+1j), -1j, (-1+1j), -1j, (-1+1j), -1j, (-1+1j)]
```

[6]:
```
make_mfOrbit(complex(0,-1),10)
```

[6]:
```
[0j, -1j, (-1-1j), 1j, (-1-1j), 1j, (-1-1j), 1j, (-1-1j), 1j, (-1-1j)]
```

[7]:
```
make_mfOrbit(complex(1,1),6)
```

[7]:
```
[0j, (1+1j), (1+3j), (-7+7j), (1-97j), (-9407-193j), (88454401+3631103j)]
```

1.2 Mandelbrot Plots

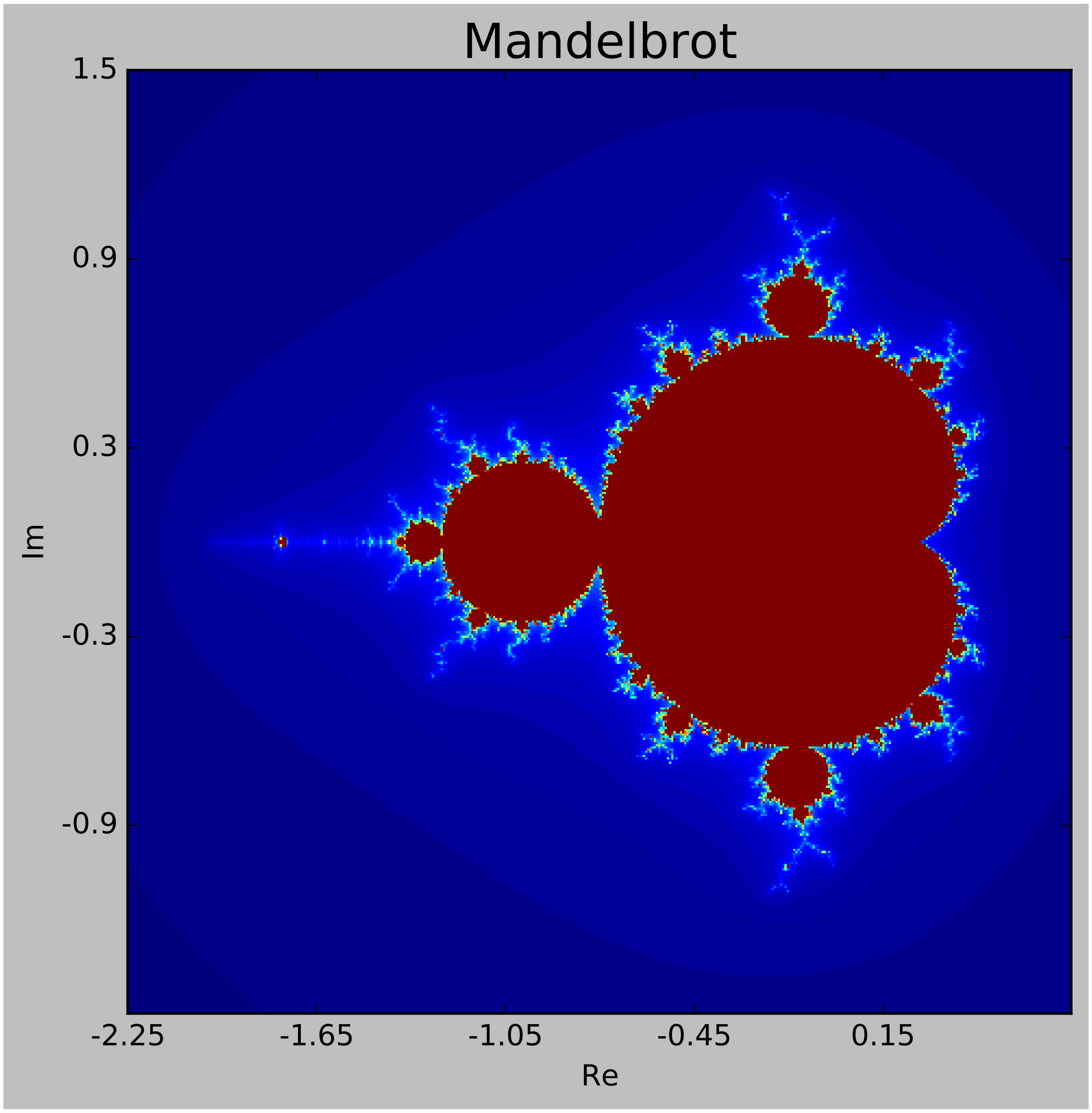
Jetzt wollen wir die Mandelbrotmenge plotten…

[8]:
```
# Hier zählen wir, ob bzw. nach wie vielen Iterationen klar ist, dass der Orbit␣
↪divergiert.
@jit
def iterationen_Bis_Divergent(c=complex(1,0), maxiter=100):
    z = complex(0,0)
    for iteration in range(maxiter):
        z = (z*z) + c

        if abs(z) > 4:
            break
            pass
        pass
    return iteration
```

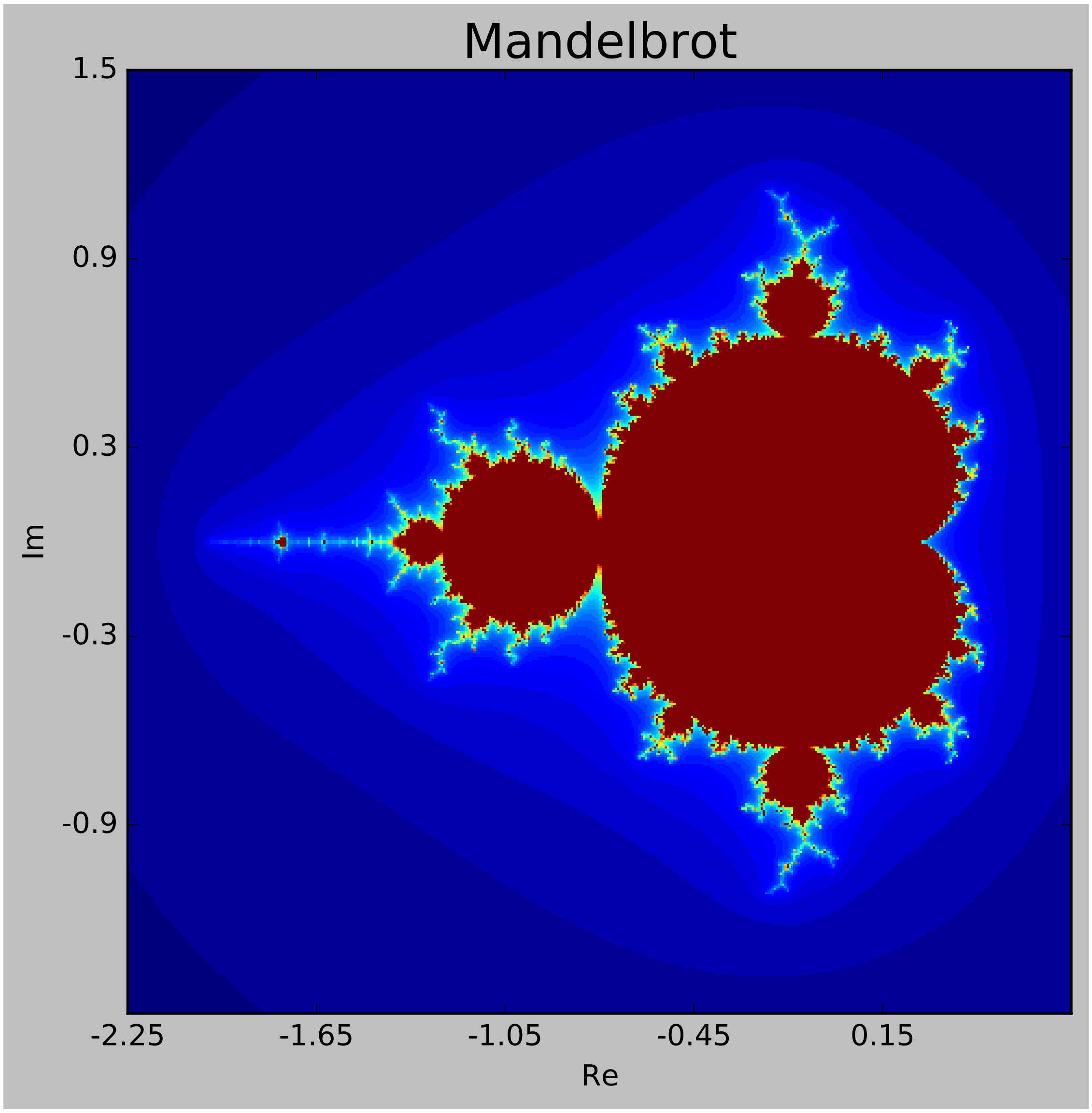
[9]:
```
import numpy as np
#import matplotlib
#matplotlib.rcParams['text.usetex'] = True # set LaTeX Fonts for Plots
import matplotlib.pyplot as plt

#plt.style.use('default')
#plt.style.use('seaborn')
plt.style.use('classic')
#plt.style.use('bmh')

# create atlas, plot mandelbrot set, display set
def mandelbrot(x_fromto=[-2.25, 0.75], y_fromto=[-1.5, -1.5+3],␣
↪maxiter_perpoint=100, resolution=1000):
```

```python
    # location and size of the atlas rectangle
    realAxis = np.linspace(x_fromto[0], x_fromto[1], resolution)
    imaginaryAxis = np.linspace(y_fromto[1], y_fromto[0], resolution)
    realAxisLen = len(realAxis)
    imaginaryAxisLen = len(imaginaryAxis)

    # 2-D array to represent mandelbrot atlas
    atlas = np.empty((realAxisLen, imaginaryAxisLen))

    # color each point in the atlas depending on the iteration count
    for ix in range(realAxisLen):
        for iy in range(imaginaryAxisLen):
            cx = realAxis[ix]
            cy = imaginaryAxis[iy]
            c = complex(cx, cy)

            atlas[ix, iy] = iterationen_Bis_Divergent(c, maxiter_perpoint)
            pass
        pass

    # plot and display mandelbrot set
    fig = plt.figure()
    fig.set_size_inches(7, 7, forward=True)
    ax = fig.add_subplot(111)
    #ax = fig.add_axes([0.1, 0.1, 0.8, 0.8]) # main axes
    ax.set_xlabel('Re')
    ax.set_ylabel('Im')
    ax.set_title('Mandelbrot', fontsize=20)
    ax.set_xticks([0,1*resolution/5,2*resolution/5,3*resolution/5,4*resolution/5,resolution])
    ax.set_xticklabels([x_fromto[0], round(x_fromto[0]+(x_fromto[1]-(x_fromto[0]))*1/5, 5), round(x_fromto[0]+(x_fromto[1]-(x_fromto[0]))*2/5, 5), round(x_fromto[0]+(x_fromto[1]-(x_fromto[0]))*3/5, 5), round(x_fromto[0]+(x_fromto[1]-(x_fromto[0]))*4/5, 5), x_fromto[1]])
    ax.set_yticks([0,1*resolution/5,2*resolution/5,3*resolution/5,4*resolution/5,resolution])
    ax.set_yticklabels([y_fromto[1], round(y_fromto[1]+(y_fromto[0]-(y_fromto[1]))*1/5, 5), round(y_fromto[1]+(y_fromto[0]-(y_fromto[1]))*2/5, 5), round(y_fromto[1]+(y_fromto[0]-(y_fromto[1]))*3/5, 5), round(y_fromto[1]+(y_fromto[0]-(y_fromto[1]))*4/5, 5), y_fromto[0]])
    plt.imshow(atlas.T, interpolation="nearest")
    plt.show()
```

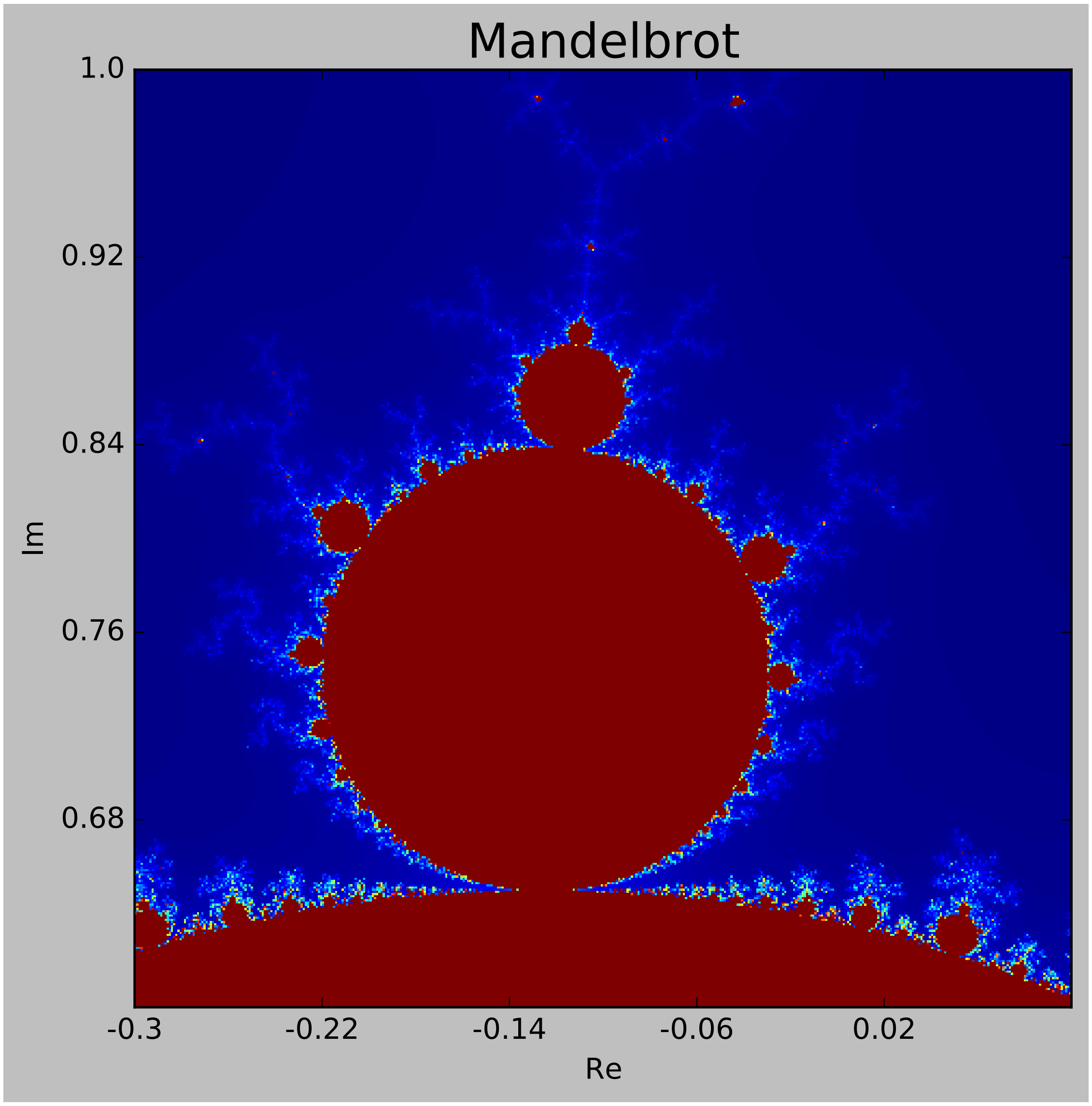
[10]:
```python
mandelbrot()
```



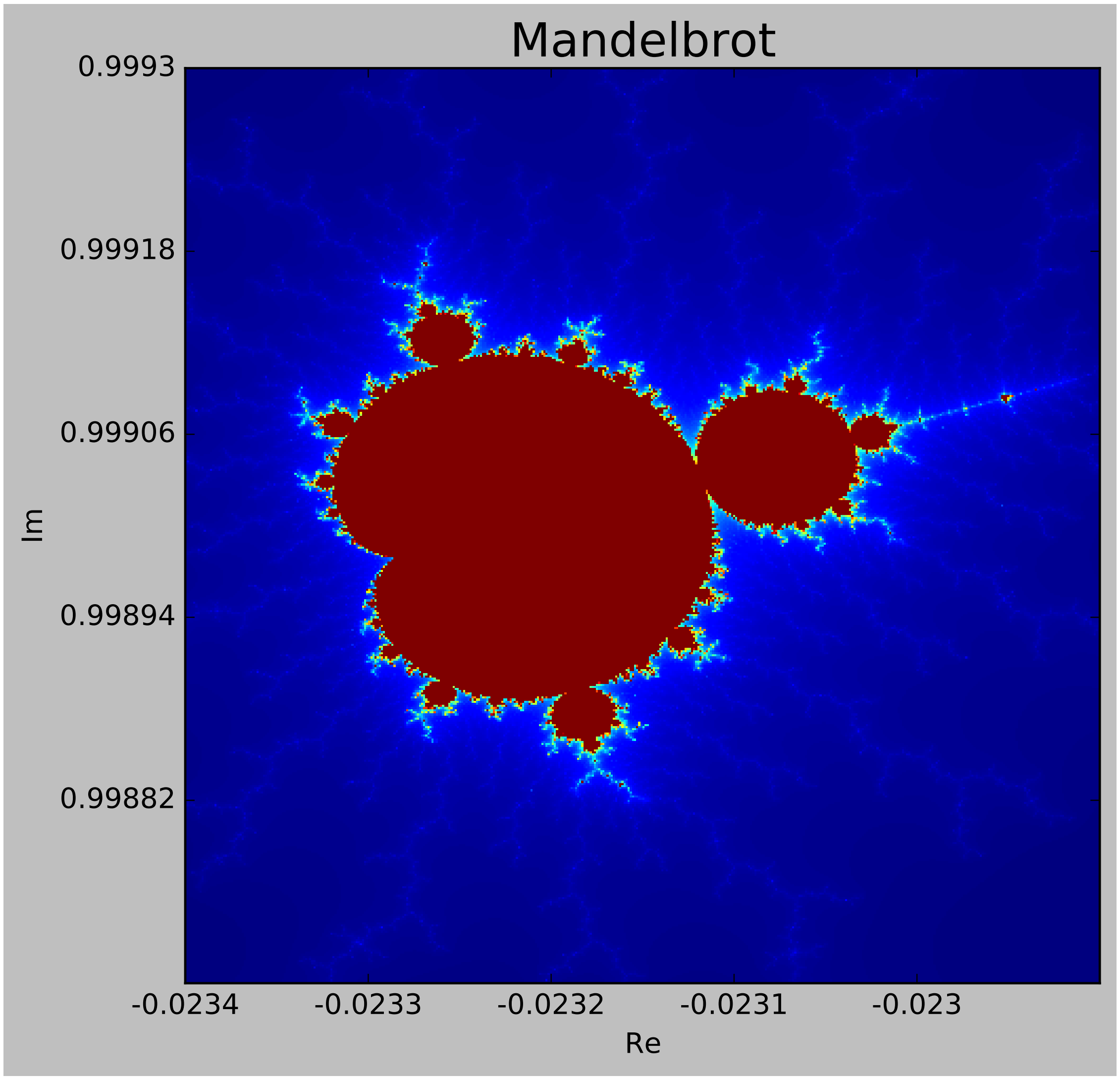
[11]:
```
mandelbrot([-2.25, 0.75], [-1.5, -1.5+3], 50, 5120)
```



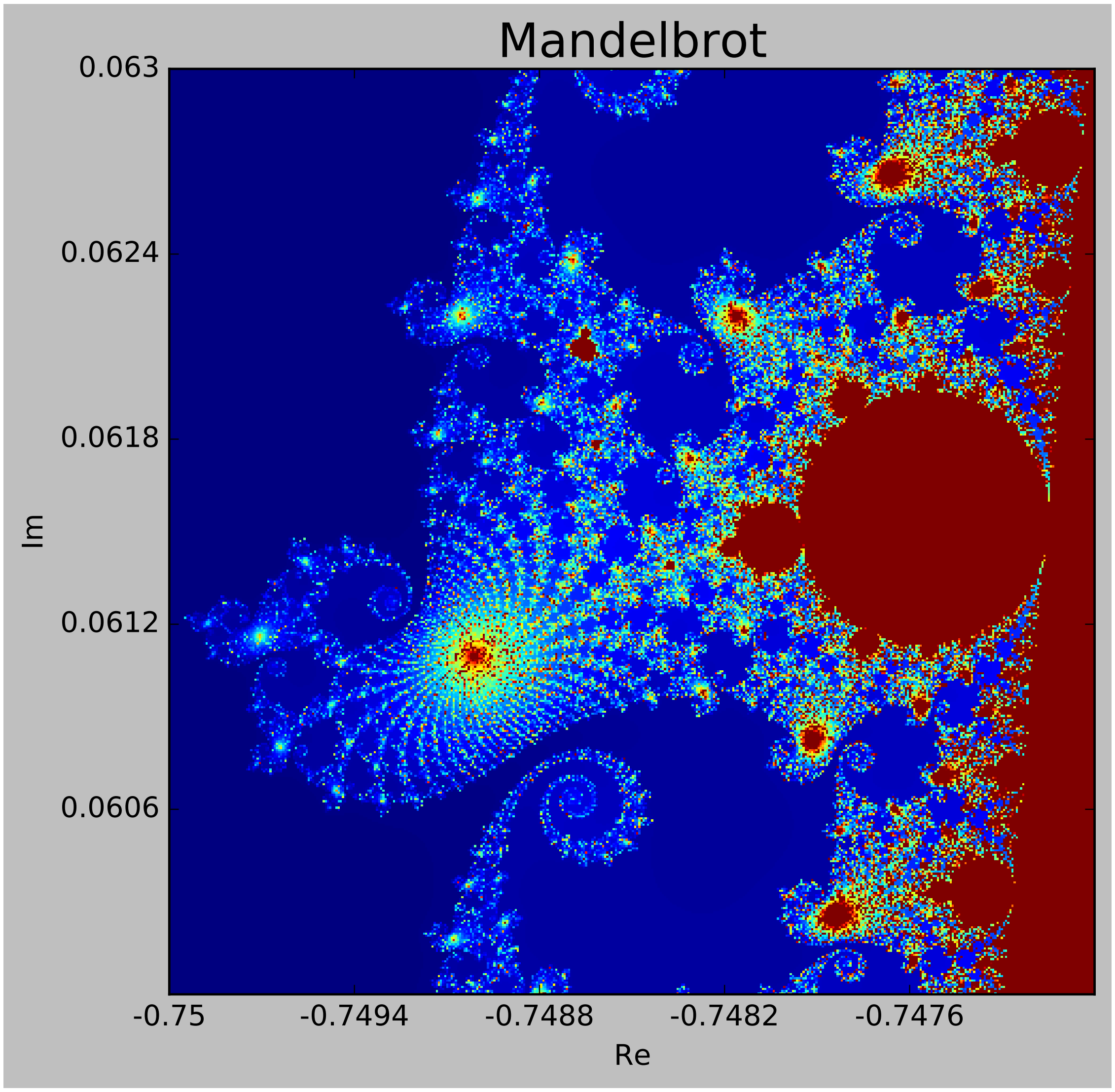
[12]:
```
mandelbrot([-0.3, 0.1], [0.6, 1.0], 600, 5120)
```



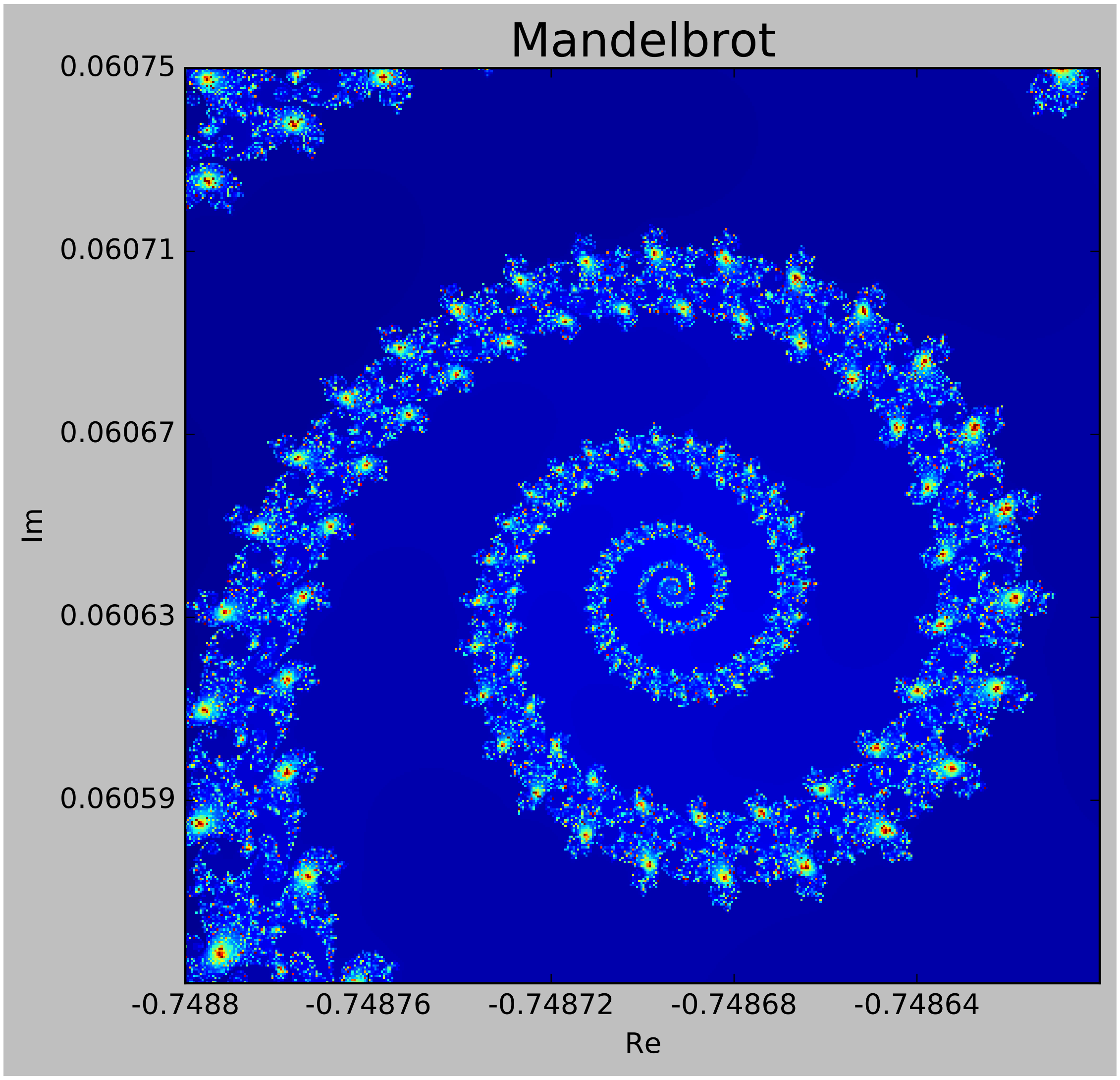
[13]:
```
mandelbrot([-0.0234, -0.0229], [0.9987, 0.9993], 600, 5120)
```



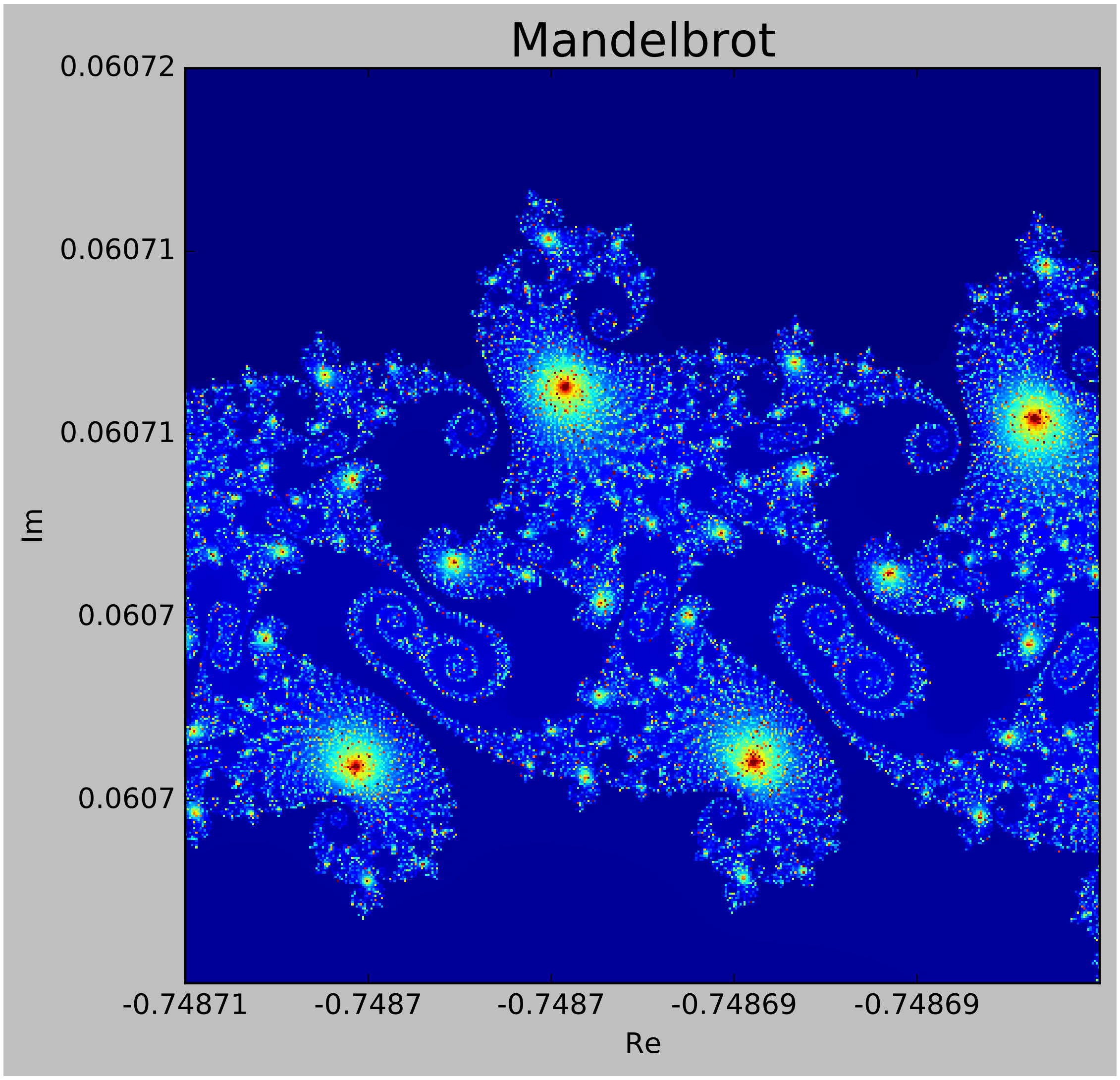
[14]:
```
mandelbrot([-0.75, -0.747], [0.06, 0.063], 2000, 5120)
```



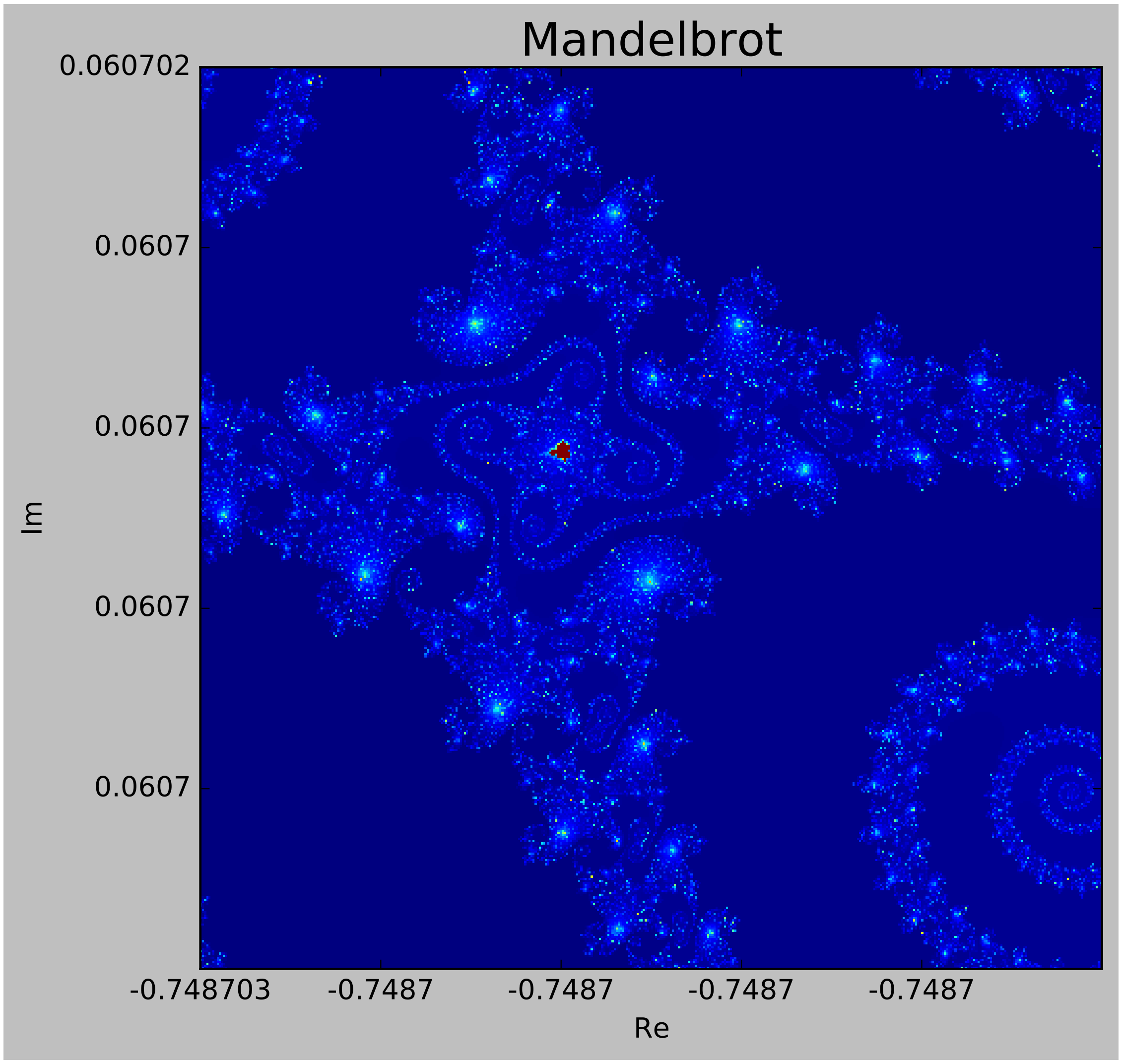
[15]:
```
mandelbrot([-0.7488, -0.7486], [0.06055, 0.06075], 2000, 5120)
```



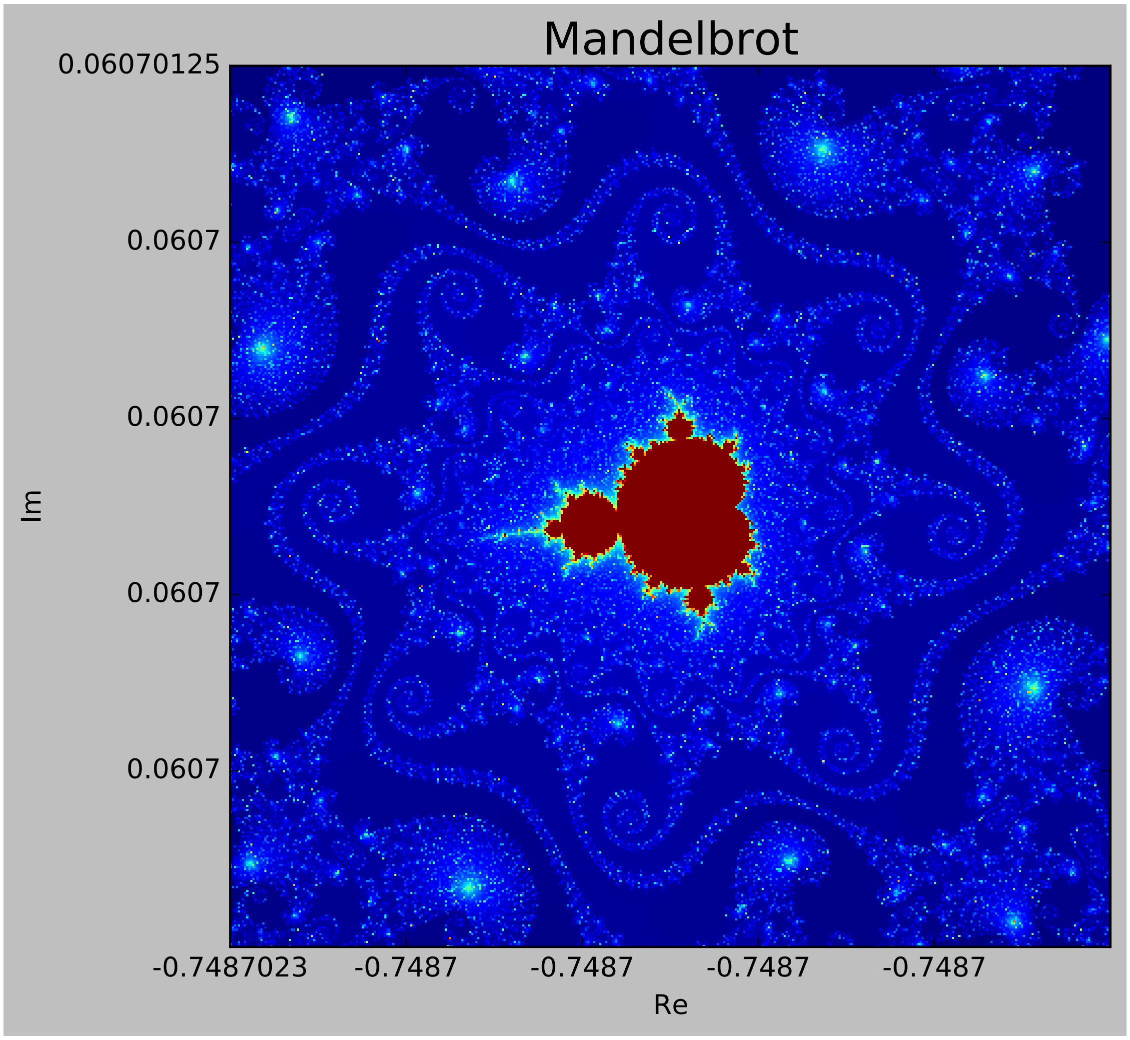
[16]:
```
mandelbrot([-0.74871, -0.74868], [0.06069, 0.06072], 2500, 5120)
```



[17]:
```
mandelbrot([-0.748703, -0.748701], [0.0607, 0.060702], 6000, 5120)
```



[18]:
```
mandelbrot([-0.7487023, -0.7487021], [0.06070105, 0.06070125], 6000, 5120)
```



Übung : Man guckt nach, an welcher Stelle hier jeweils “reingezoomt” wird.

1.3 Wie lässt man solche Bilder berechnen?

Grundsätzlich guckt man, ob für einen Wert $c \in \mathbb{C}$ der Orbit von $f(z) = z^2 + c$ mit Startwert $z = 0$ divergiert. Falls nein, dann wird er Punkt rot eingefärbt. Falls aber ja, dann wird der Punkt mit einer Farbe eingefärbt, die vom Index des Iterationsschrittes abhängt, an dem die Folge als divergent erklärt wird. Stellt sich also die Frage, wann dass eine Folge als divergent eingestuft werden darf?

Wer oben das `mandelbrot`Programm überflogen hat, dem ist aufgefallen, dass als Abbruchbedingung für die Iteration eines Startwerts die Bedingung $|z_k| > 2$ verwendet wurde. Man kann nämlich zeigen, dass, falls für einen Wert z_k in der Itartionsfolge $|z_k| > 2$ gilt, der Orbit sicher divergiert. Diesen Fact macht man sich in der Programmierung zu Nutze.

> Ist für ein $k \in \mathbb{N}$ der Betrag $|z_k| > 2$, so divergiert die Folge $< z_k >$ definiert durch $z_{k+1} = z_k^2 + c$ mit fixem $c \in \mathbb{C}$ und Startwert $z_0 = 0$.

Beweis : Für $f(z) = z^2 + c$ mit $z_0 = 0$ haben wir

$$z_0 = 0, z_1 = c, z_2 = c^2 + c, \dots$$

Ferner ist

$$\frac{|f(z)|}{|z|} = \frac{|z^2 + c|}{|z|} \geq \frac{|z|^2 - |c|}{|z|} = |z| - \frac{|c|}{|z|} > |z| - 1 > 1$$

wobei die Dreiecksungleichung verwendet wurde. Das heisst wir haben ein Wachstum pro Zeitschritt mit Faktor grösser 1.

Ist nun $|c| < 2$ und für ein k $|z_k| > 2$, dann erfüllt $z = z_k$ die Voraussetzung und die Folge wird divergieren. Ist $|c| > 2$, dann ist $|c^2 + c| \geq |c|^2 - |c| = |c| \cdot |c - 1| > |c| > 2$. Also erfüllt $z = z_2$ die Voraussetzung.

Der kommentierte Beweis zur Divergenz im Apfelmännchen von gym math gibts unter eben dem Link.

1.4 Was hat das Apfelmännchen mit den Walen zu tun?

Betrachte die Iteration $z_{k+1} = z_k^2 + c$ und setze eine affine Transformation $z_k = ax_k + b$ mit $a, b, x_k \in \mathbb{R}$. Es ist

$$ax_{k+1} + b = (ax_k + b)^2 + c = a^2 x_k^2 + 2abx_k + b^2 + c$$

Es folgt

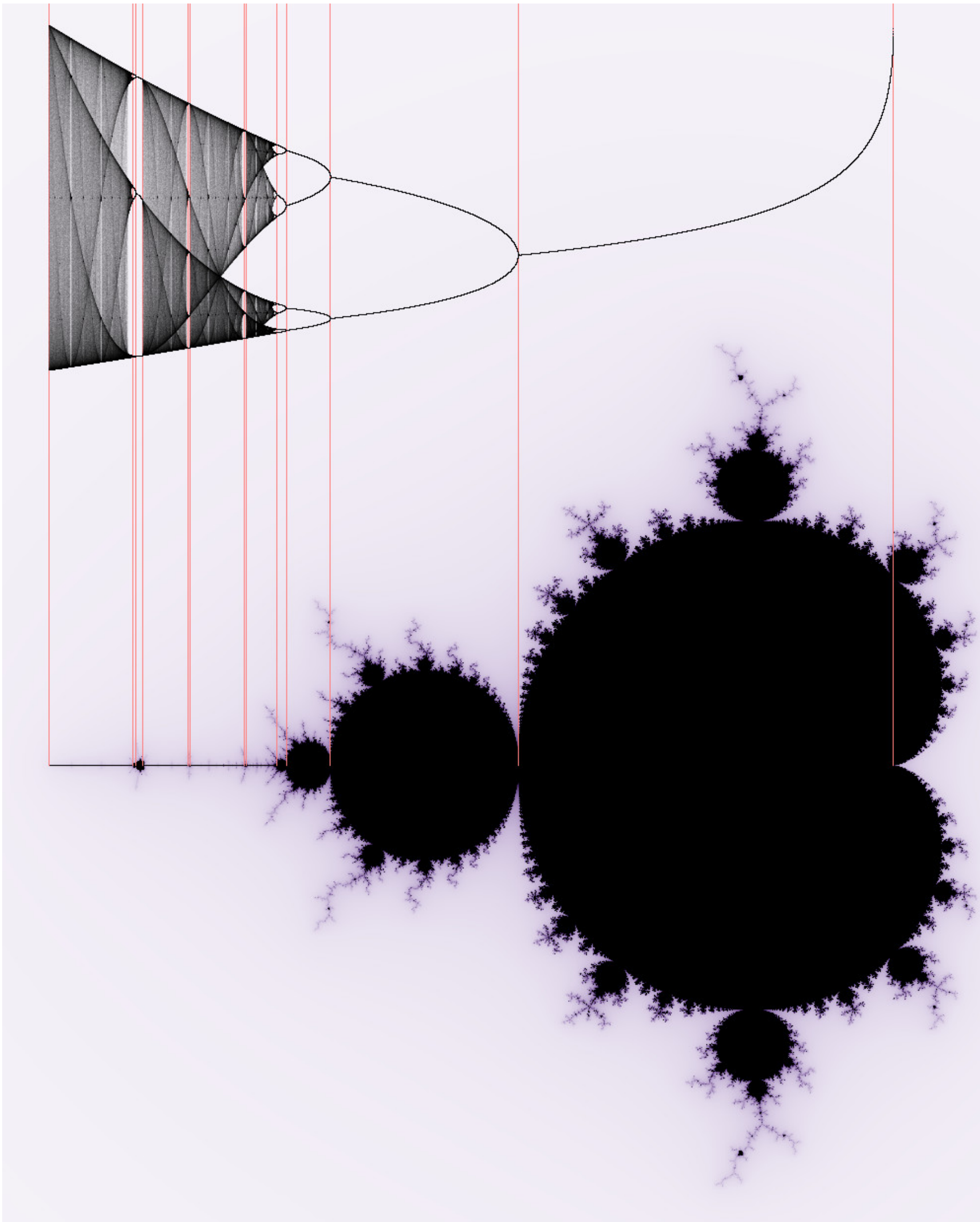
$$x_{k+1} = 2bx_k(1 + \frac{a}{2b}x_k) + \frac{b^2 - b + c}{a}$$

Übung : Zeige obige Folgerung. Falls es nicht klappt, dann kannst du dir den Zusammenhang zwischen Mandelbrot und Wale anschauen.

Übung : Begründe, dass ein Vergleich mit der logistischen Funktion $f_r(x)$ heisst: $b = \frac{r}{2}$ und $a = -r$. Ferner zeige man, dass der interessante Bereich $1 < r < 4$ den Werten $-2 < c < \frac{1}{4}$ entspricht.

Übung : Bestimme die c-Bereiche der Fixpunktregionen $1 < r < 3$ und $3 < r < 1 + \sqrt{6}$.

Abschliessend begreife man folgendes Bild !



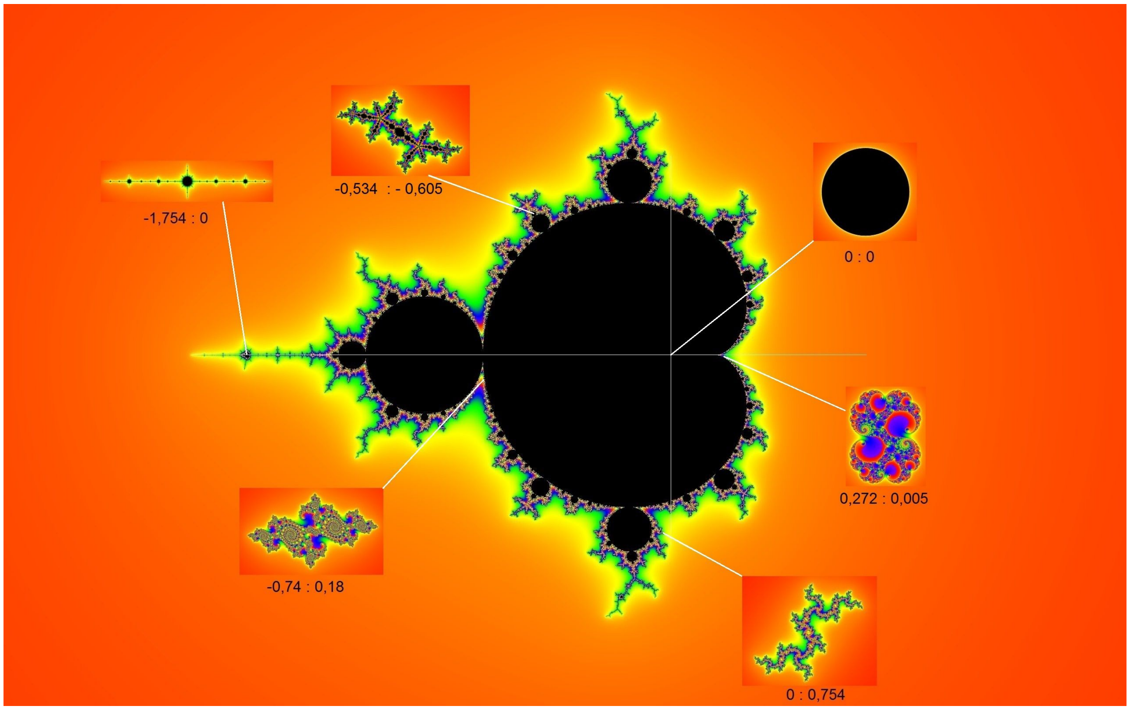
1.5 Enjoy! Don't Freak out!

Übung : Jetzt kommt eine gemütliche Pflichtaufgabe. Höre zum Video Mandelbrot Zoom, in dem eine Stelle in der komplexen Ebene vergrössert wird, den sinnigen Titel "Functionality" dazu. Mindestens bis Minute 7.5 den Zoom angucken. Dem Video voraus schicken möchte ich, dass die Dimension des Unversiums im Verhältnis zu einem Atomkern in etwa $10^{40} \div 1$ ist. In Längen gedacht heisst dies also für Zahlen, welche sich erst an der 40-sten Stelle nach dem Komma unterscheiden, dass das Universum um einen Atomdurchmesser verschoben wurde. Erinnere dich an diesen Fact, wenn jeweils die Grössenordnung im Video eingeblendet wird.

So, los gehts! Zuerst den Sound einrichten…

1) Functionality 2 von DJ Dimsa

2) Mandelbrot Zoom


-1,754 : 0
-0,534 : - 0,605
0 : 0
0,272 : 0,005
-0,74 : 0,18
0 : 0,754