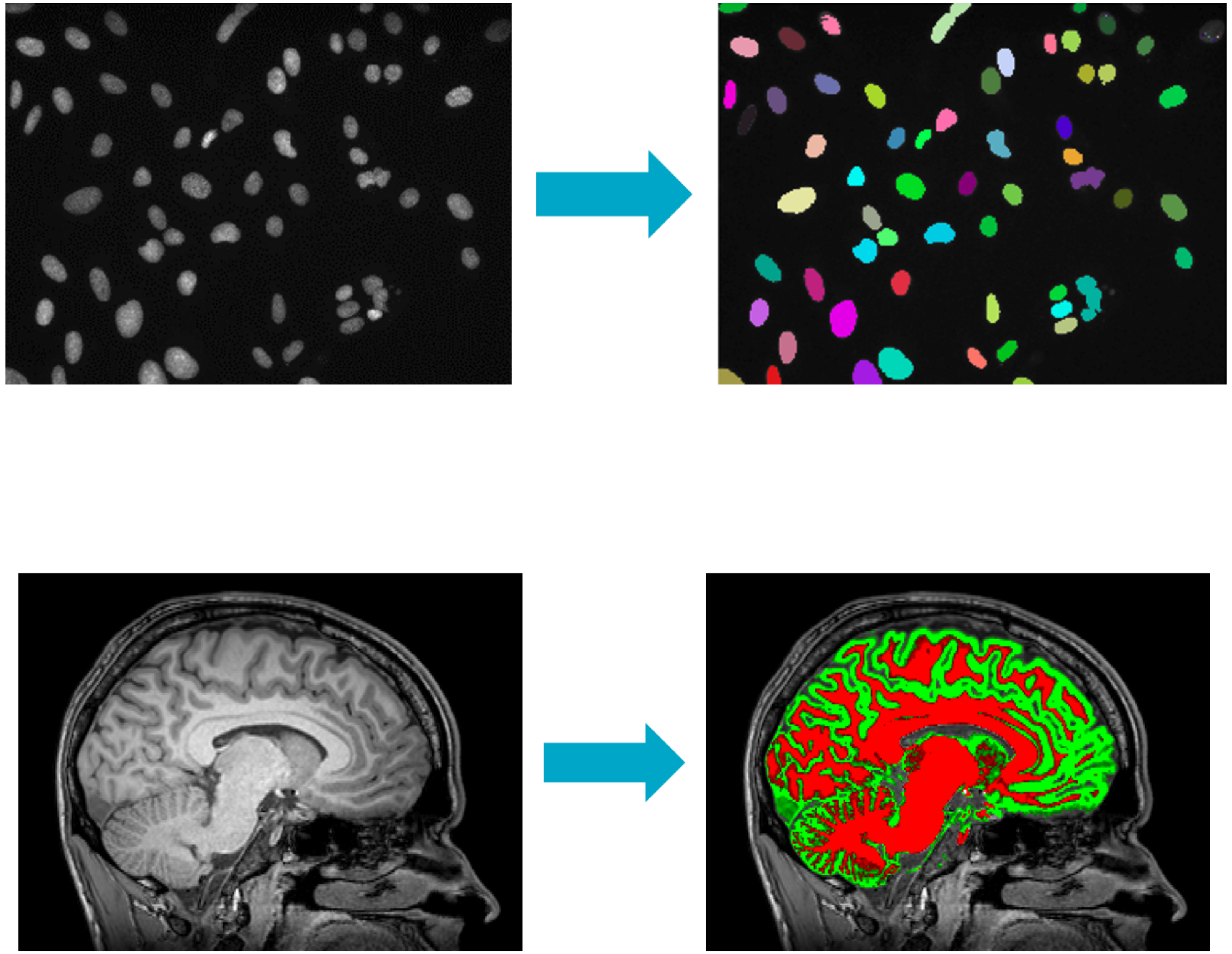
BIOMEDICAL IMAGE ANALYSIS IN PYTHON

Objects and Labels

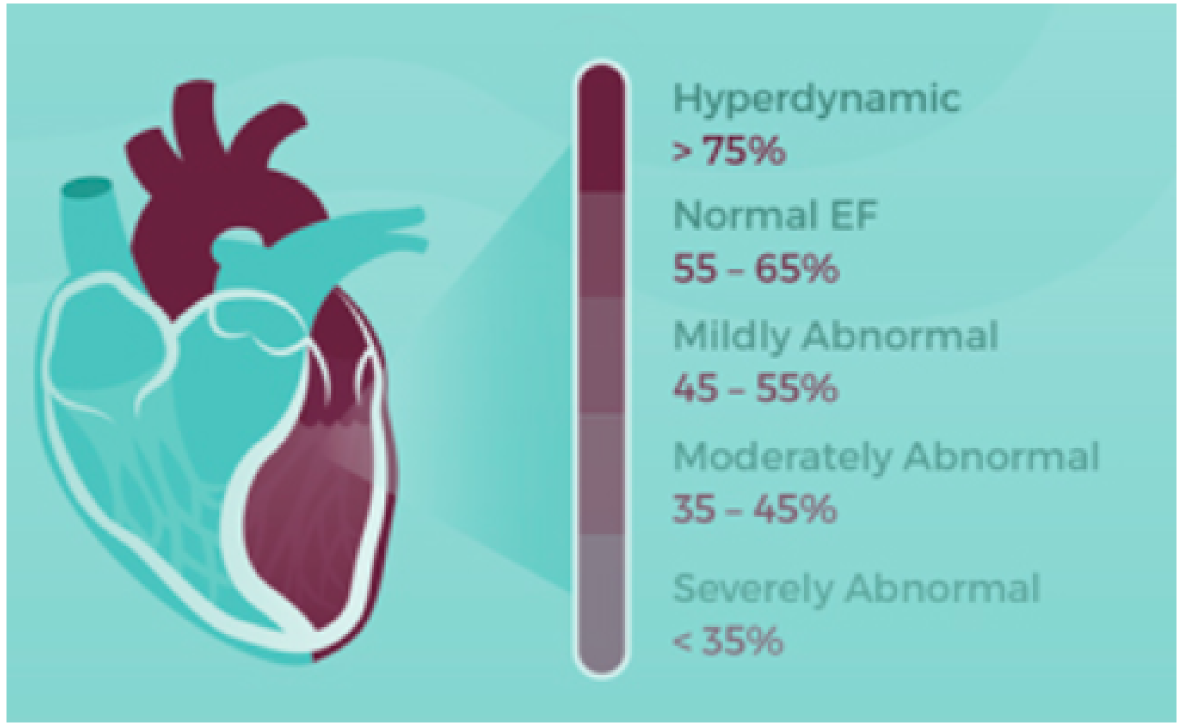
Stephen Bailey
Instructor

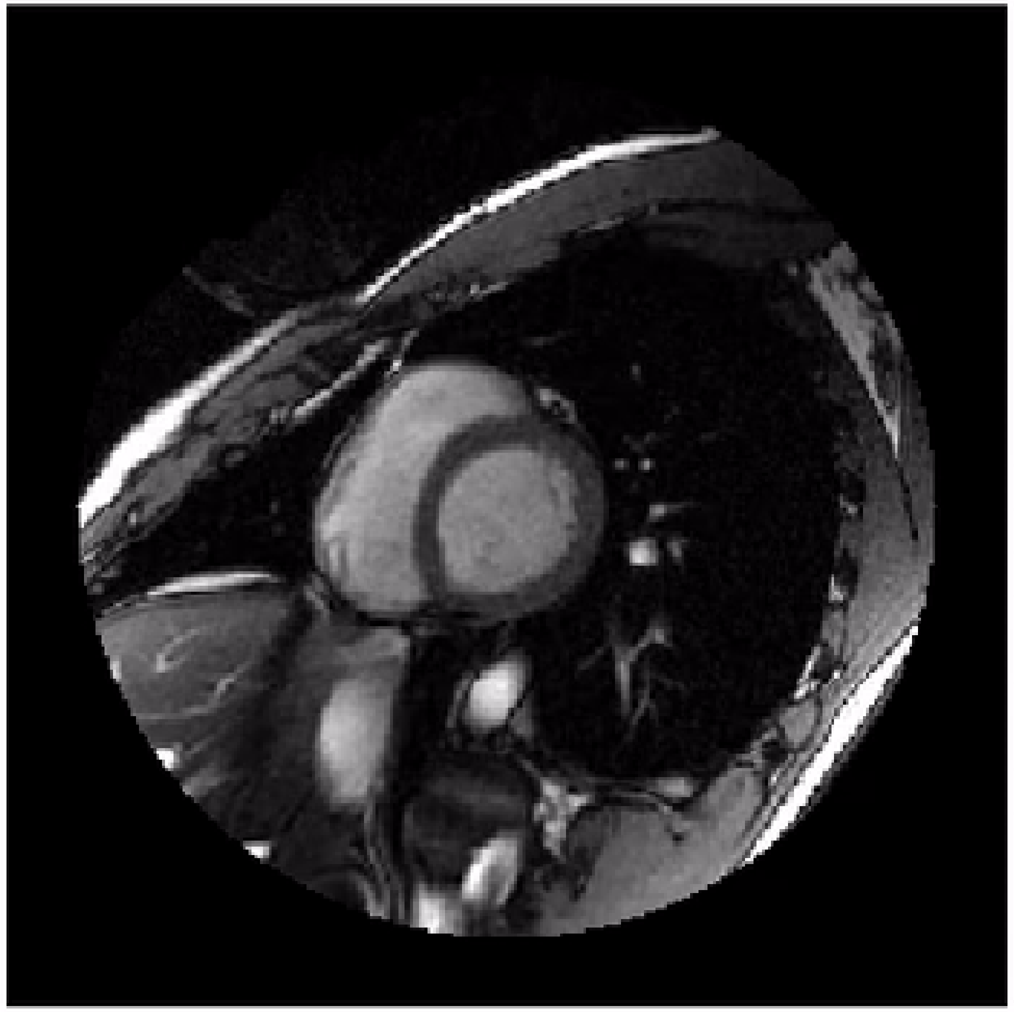
Segmentation splits an image into parts



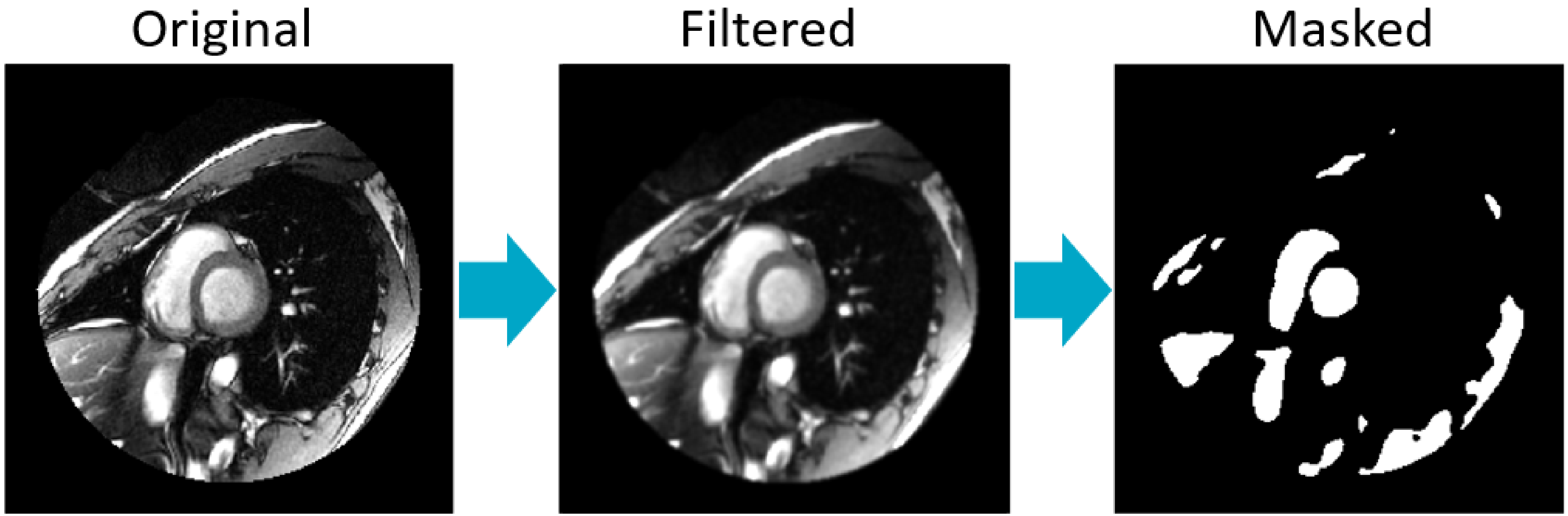
Sunnybrook Cardiac Database

Ejection fraction: the proportion of blood pumped out of the heart's left ventricle (LV).





Labeling image components



Labeling image components

```
import scipy.ndimage as ndi

im=imageio.imread('SCD4201-2d.dcm')
filt=ndi.gaussian_filter(im,
                         sigma=2)
mask = filt > 150

labels, nlabels = ndi.label(mask)
```

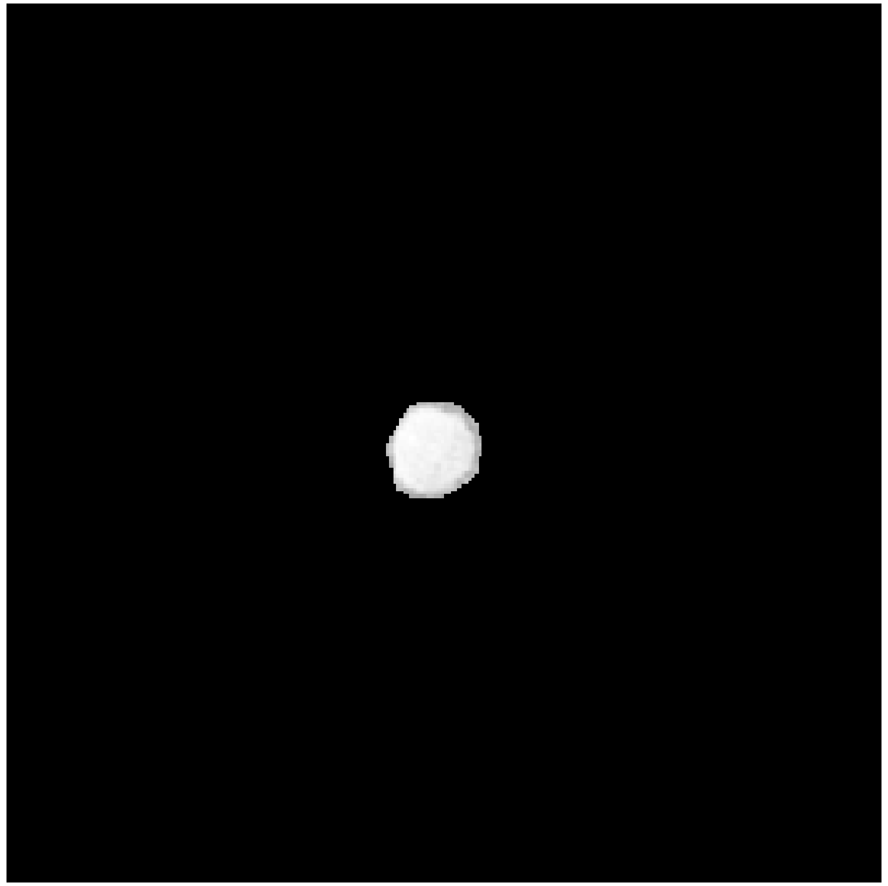
```
nlabels
    14
```

```
plt.imshow(labels, cmap='rainbow')
plt.axis('off')
plt.show()
```

Label selection

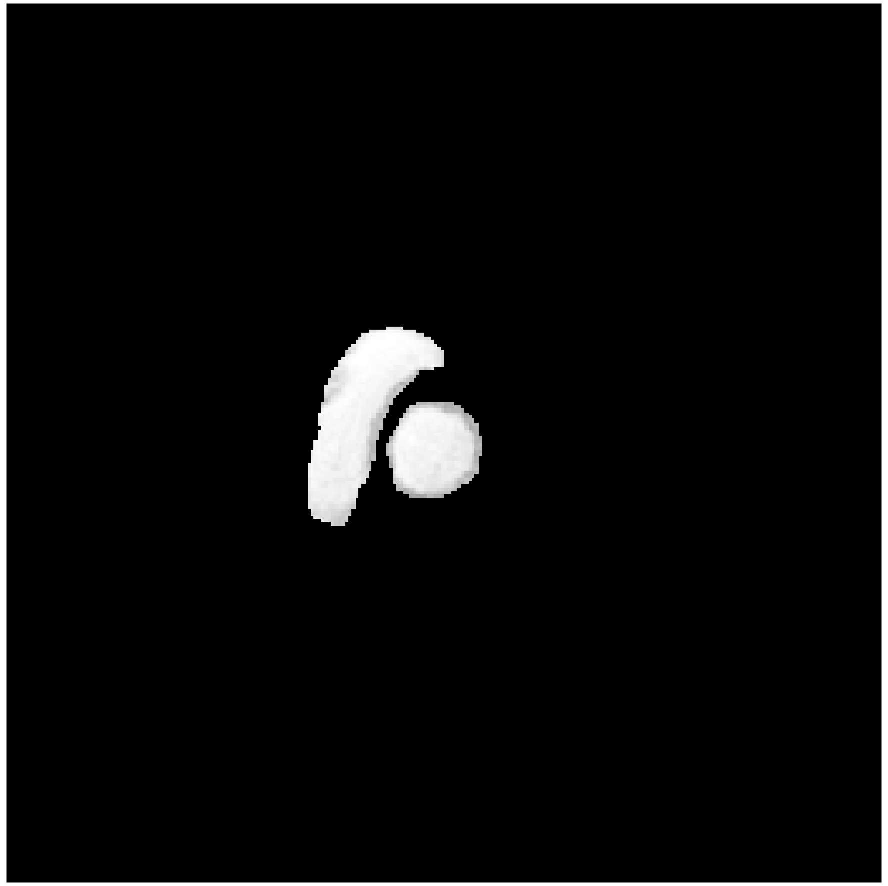
Select a single label within image:

```
np.where(labels == 1, im, 0)
```



Select many labels within image:

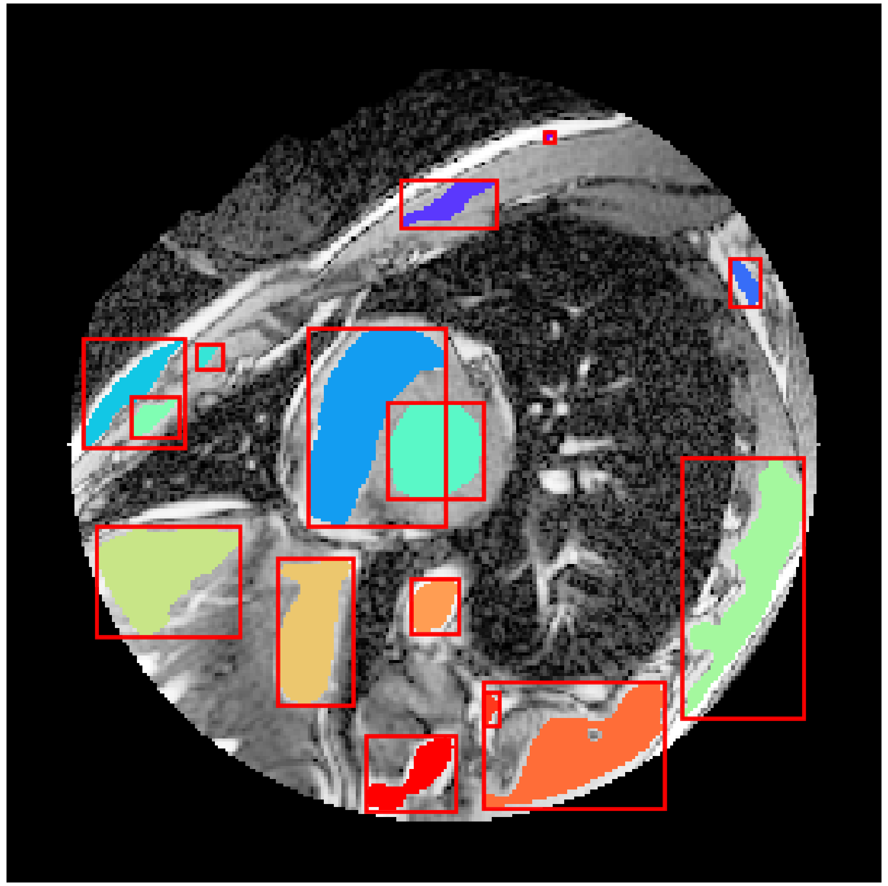
```
np.where(labels < 3, im, 0)
```



Object extraction

- **Bounding box**: range of pixels that completely encloses an object

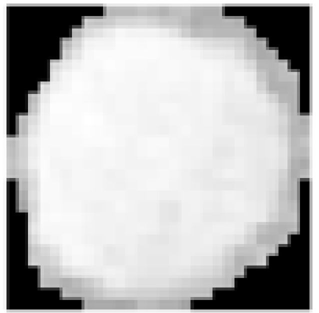
- `ndi.find_objects()` returns a list of bounding box coordinates



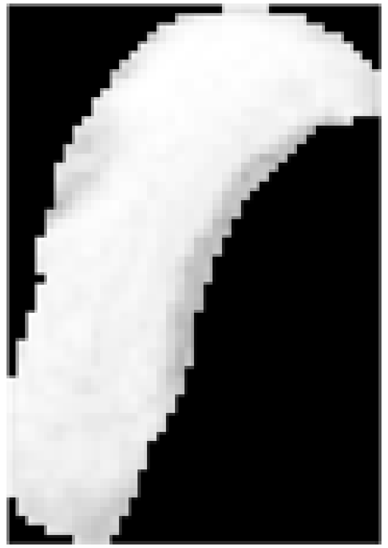
Object extraction

```
labels, nlabels = ndi.label(mask)
boxes = ndi.find_objects(labels)

boxes[0]
    (slice(116,139), slice(120, 141))
```

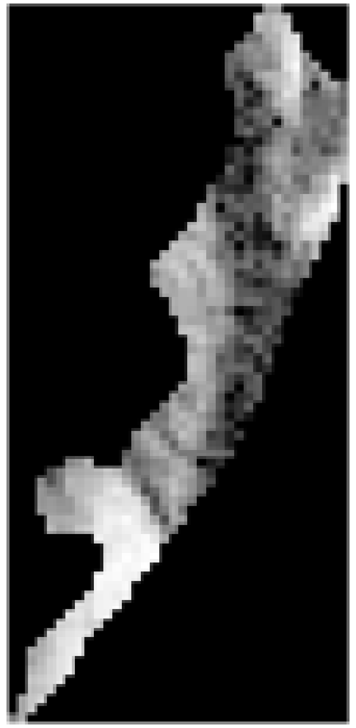
im[boxes[0]]



im[boxes[1]]



im[boxes[2]]



BIOMEDICAL IMAGE ANALYSIS IN PYTHON

Let's practice!

BIOMEDICAL IMAGE ANALYSIS IN PYTHON

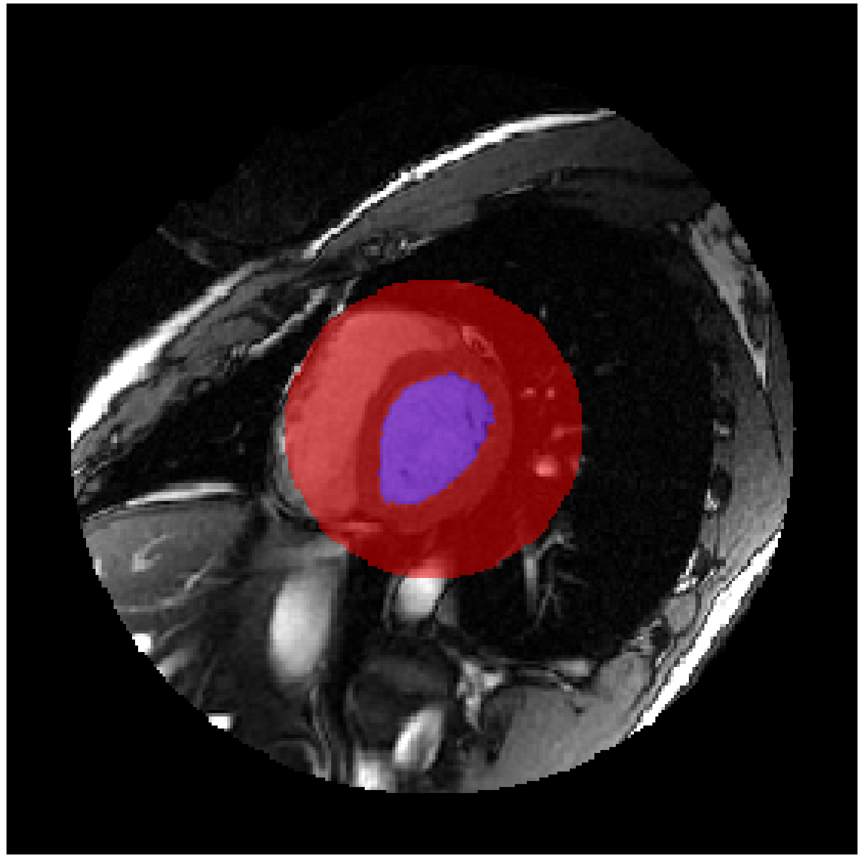
Measuring Intensity

Stephen Bailey
Instructor

Measuring intensity

We have the following labels for a single volume of the cardiac time series:

1. Left ventricle
2. Central portion



Functions

`scipy.ndimage.measurements`:

- `ndi.mean()`
- `ndi.median()`
- `ndi.sum()`
- `ndi.maximum()`
- `ndi.standard_deviation()`
- `ndi.variance()`

Functions applied over all dimensions, optionally at specific labels.

Custom functions:

- `ndi.labeled_comprehension()`

Calling measurement functions

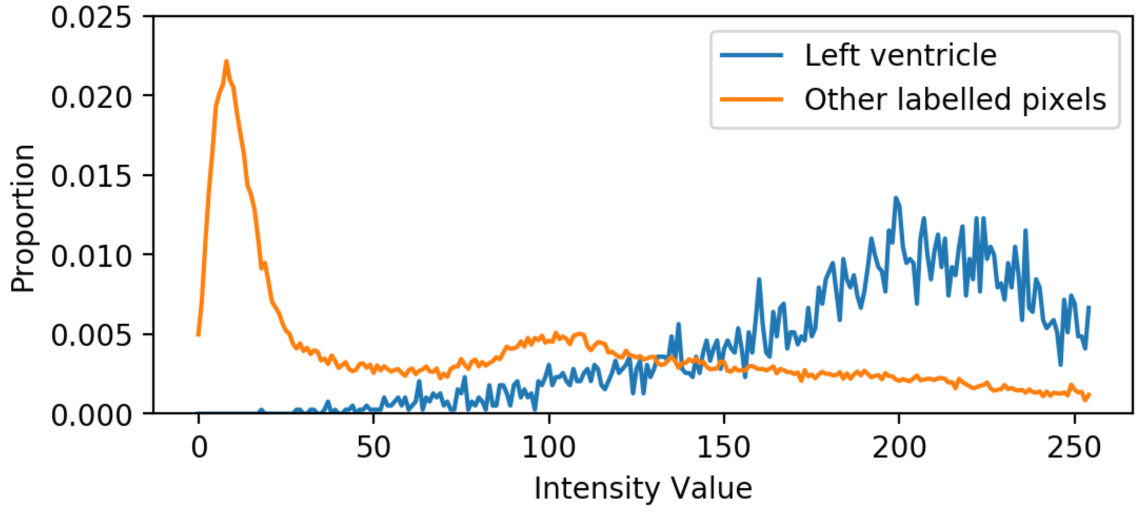
```
import imageio
import scipy.ndimage as ndi

vol=imageio.volread('SCD-3d.npz')
label=imageio.volread('labels.npz')

# All pixels
ndi.mean(vol)
    3.7892

# Labeled pixels
ndi.mean(vol, label)
    89.2342

# Label 1
ndi.mean(vol, label, index=1)
    163.2930

# Labels 1 and 2
ndi.mean(vol, label, index=[1,2])
    [163.2930, 60.2847]
```

Object histograms

```
hist=ndi.histogram(vol, min=0, max=255, bins=256)

obj_hists=ndi.histogram(vol, 0, 255, 256,
                        labels, index=[1, 2])

len(obj_hists)
    2
```

Object histograms

```
plt.plot(obj_hists[0],
    label='Left ventricle')
plt.plot(obj_hists[1],
    label='Other labelled pixels')
plt.legend()
plt.show()
```



- Histograms containing multiple tissue types will have several peaks
- Histograms for well-segmented tissue often resemble a normal distribution

BIOMEDICAL IMAGE ANALYSIS IN PYTHON

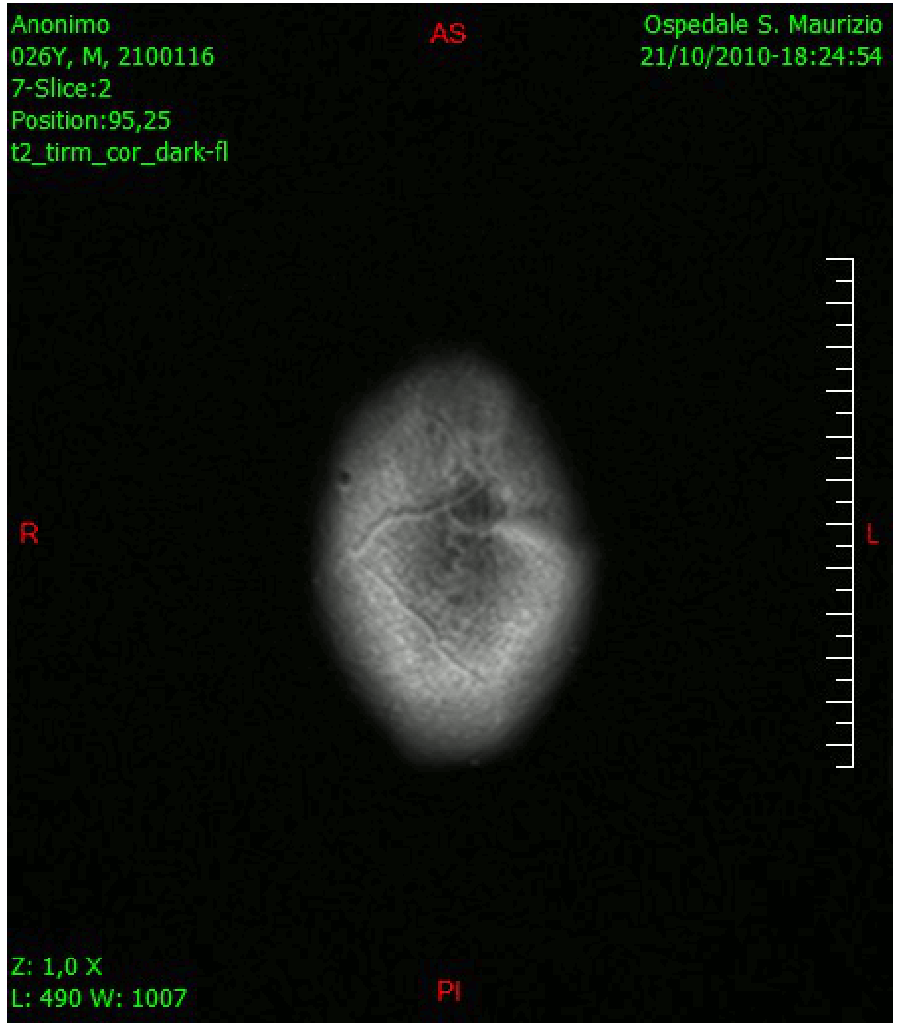
Let's practice!

BIOMEDICAL IMAGE ANALYSIS IN PYTHON

Measuring Morphology

Stephen Bailey
Instructor

Morphology


Anonimo
026Y, M, 2100116
7-Slice:2
Position:95,25
t2_tirm_cor_dark-fl
AS
Ospedale S. Maurizio
21/10/2010-18:24:54
R
L
Z: 1,0 X
L: 490 W: 1007
PI

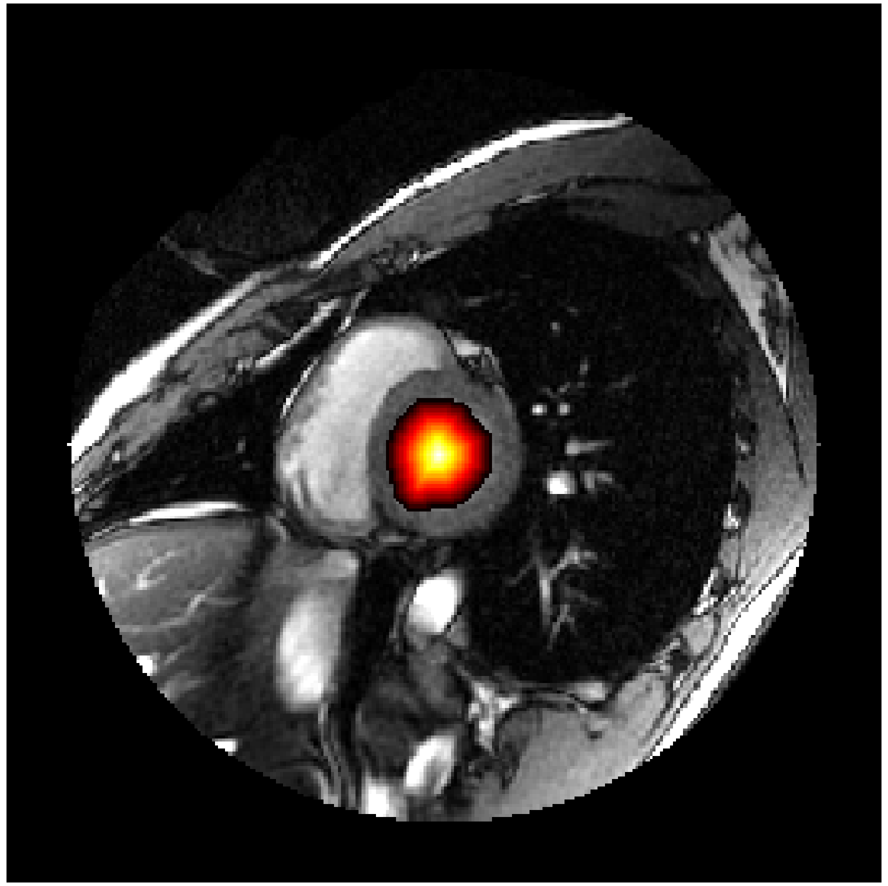
Spatial extent

Spatial extent is the product of:

1. Space occupied by each element
2. Number of array elements

```
# Calculate volume per voxel
d0, d1, d2 = vol.meta['sampling']
dvoxel = d0 * d1 * d2

# Count label voxels
nvoxels=ndi.sum(1, label, index=1)

# Calculate volume of label
volume = nvoxels * dvoxel
volume
    1249023
```

Distance transformation



Euclidean Distance

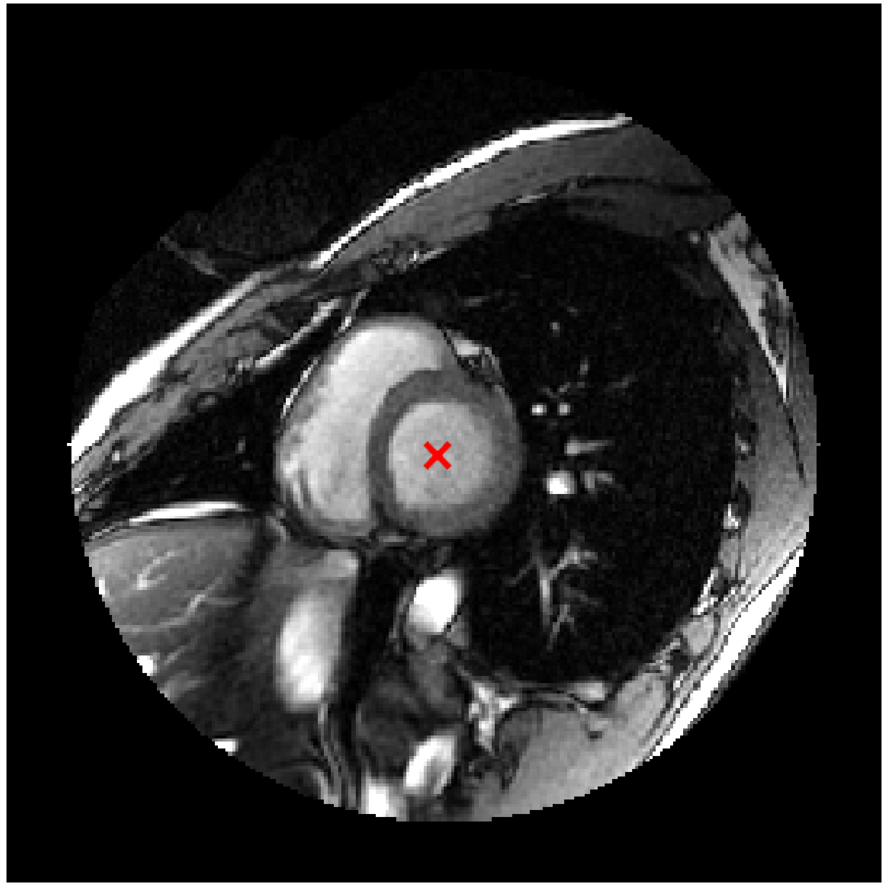
```
# Create a left ventricle mask
mask=np.where(labels == 1, 1, 0)

# In terms of voxels
d=ndi.distance_transform_edt(mask)

d.max()
    12.3847
```

```
# In terms of space
d=ndi.distance_transform_edt(mask,
    sampling=vol.meta['sampling'])

d.max()
    5.8038
```

Center of mass

```
com=ndi.center_of_mass(vol,
                       labels,
                       index=1)

com
    (5.5235, 128.0590, 128.0993)
```

```
plt.imshow(vol[5], cmap='gray')
plt.scatter(com[2], com[1])
plt.show()
```



BIOMEDICAL IMAGE ANALYSIS IN PYTHON

Let's practice!

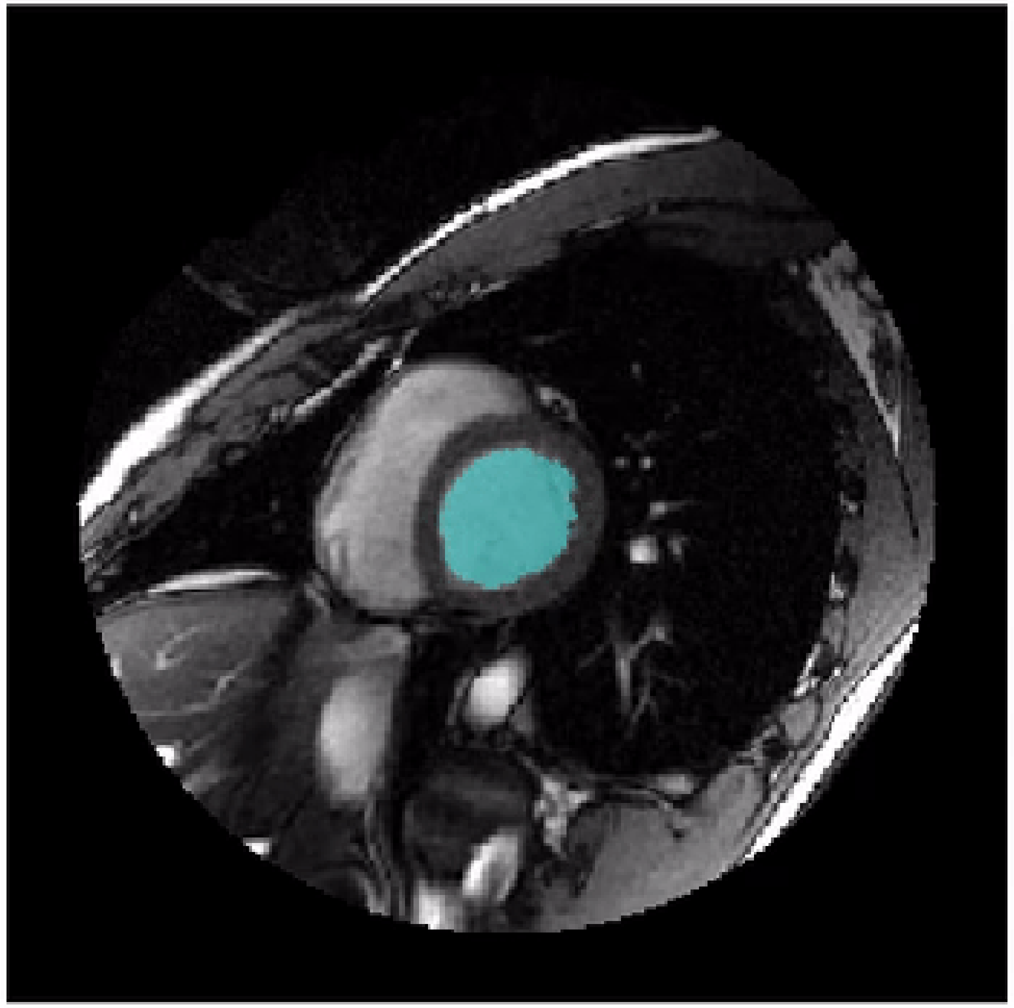
BIOMEDICAL IMAGE ANALYSIS IN PYTHON

Measuring in Time

Stephen Bailey
Instructor

Ejection fraction

$$Ejection\ Fraction = \frac{LV_{max} - LV_{min}}{LV_{max}}$$



Ejection fraction

Procedure

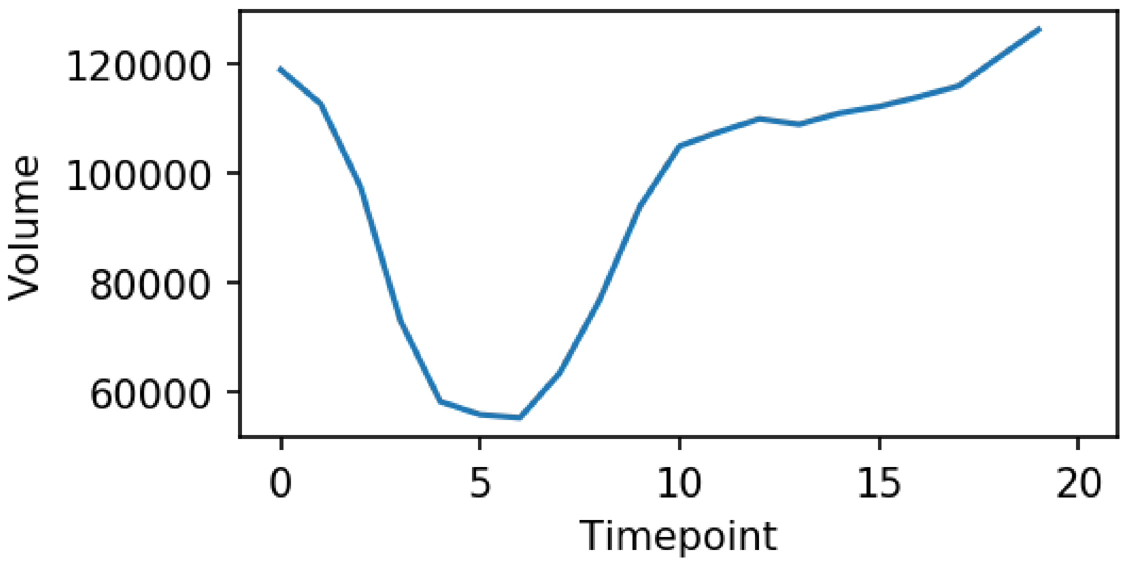
1. Segment left ventricle
2. For each 3D volume in the time series, calculate volume
3. Select minimum and maximum
4. Calculate ejection fraction

Calculate volume for each time point

```
# Stored in (t,z,x,y) format
vol_ts.shape
    (20, 12, 256, 256)
labels.shape
    (20, 12, 256, 256)
```

```
# Calculate voxel volume in mm^3
d0,d1,d2,d3=vol_ts.meta['sampling']
dvoxel = d1 * d2 * d3

# Instantiate empty list
ts = np.zeros(20)

# Loop through volume time series
for t in range(20):

    nvoxels=ndi.sum(1,
                    labels[t],
                    index=1)

    ts[t] = nvoxels * dvoxel
```

```
plt.plot(ts)
plt.show()
```



Calculate ejection fraction

```
min_vol = ts.min()
max_vol = ts.max()

ejec_frac = (max_vol - min_vol) / max_vol

ejec_frac
    0.58672
```

BIOMEDICAL IMAGE ANALYSIS IN PYTHON

Let's practice!