

9-manifolds_spectral-clust

April 19, 2021

#

Ciencia de Datos

Víctor Muñiz Sánchez

Maestría en Cómputo Estadístico

Enero a junio 2021

1 Spectral clustering e imágenes.

1.1 Ejemplo 1

```
[39]: import numpy as np
import matplotlib.pyplot as plt
from sklearn import cluster
from skimage.io import imread
from skimage.color import rgb2gray
from scipy.misc import imresize (deprecated)
from PIL import Image
import matplotlib.pyplot as pylab

%matplotlib inline

import os
os.chdir('/home/victor/cursos/ciencia_de_datos_2020/')
```

```
[40]: im_orig = imread('data/baseball12.jpg')
pylab.figure(figsize=(8,8))
pylab.axis('off')
pylab.imshow(im_orig)
```

```
[40]: <matplotlib.image.AxesImage at 0x7fb1f80d2e80>
```



```
[41]: im_gray = rgb2gray(im_orig)
      pylab.figure(figsize=(8,8))
      plt.axis('off')
      pylab.imshow(im_gray, cmap=plt.cm.gray)
```

```
[41]: <matplotlib.image.AxesImage at 0x7fb1f0af8d60>
```



```
[42]: #im = imresize(imread('data/baseball2.jpg'), (100,100,3)) (deprecated)
      im = np.array(Image.fromarray(imread('data/baseball2.jpg')).resize((100,100)))
      X = np.reshape(im, (-1, im.shape[-1]))
```

```
[44]: X.shape
```

```
[44]: (10000, 3)
```

Trabajaremos con una imagen de 100×100 . ¿Cómo construyo X ?

```
[17]: #construye X tomando el valor de cada canal, en cada pixel
      print('imagen \n',im[:3,:4,:])
      print('Arreglo \n',X[:9,])
```

imagen

```

[[[159 127 100]
  [155 124 97]
  [157 126 98]
  [157 126 97]]

 [[156 124 99]
  [157 125 100]
  [157 126 100]
  [155 125 97]]

 [[156 125 96]
  [159 128 100]
  [156 125 97]
  [161 127 100]]]
Arreglo
[[159 127 100]
 [155 124 97]
 [157 126 98]
 [157 126 97]
 [157 127 99]
 [157 127 99]
 [155 124 95]
 [160 129 101]
 [159 128 102]]

```

```

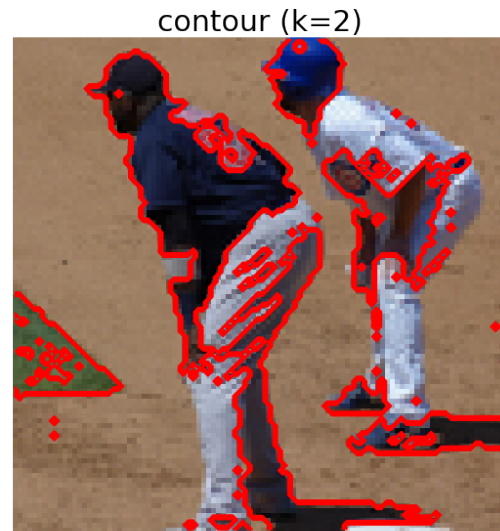
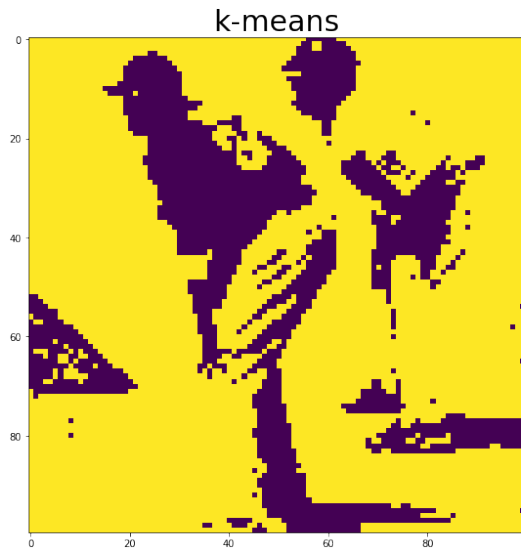
[45]: k = 2
k_means = cluster.MinibatchKMeans(n_clusters=k, random_state=10)
k_means.fit(X)
y_pred = k_means.predict(X)
labels = np.reshape(y_pred, im.shape[:2])
pylab.figure(figsize=(20,20))
pylab.subplot(121)
pylab.imshow(np.reshape(y_pred, im.shape[:2])), pylab.title('k-means', size=30)
pylab.subplot(122)
pylab.imshow(im), pylab.contour(labels == 0, colors='red'), pylab.axis('off')
pylab.title('contour (k=2)', size=30)

```

```

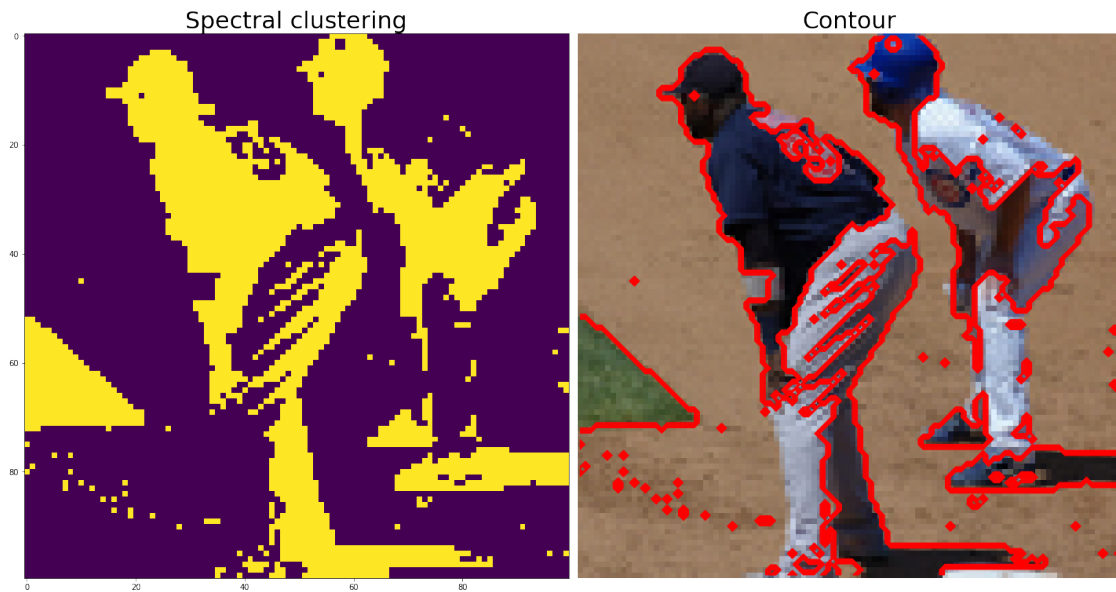
[45]: Text(0.5, 1.0, 'contour (k=2)')

```



```
[23]: spectral = cluster.SpectralClustering(n_clusters=k, eigen_solver='arpack',
    ↪affinity="nearest_neighbors",
    n_neighbors=800, random_state=10)

spectral.fit(X)
y_pred = spectral.labels_.astype(np.int)
labels = np.reshape(y_pred, im.shape[:2])
pylab.figure(figsize=(20,20))
pylab.subplot(121)
pylab.imshow(np.reshape(y_pred, im.shape[:2])), pylab.title('Spectral
    ↪clustering', size=30)
pylab.subplot(122)
pylab.imshow(im),
pylab.contour(labels == 0, colors='red'), pylab.axis('off'), pylab.
    ↪title('Contour', size=30), pylab.tight_layout()
pylab.show()
```



1.2 Spectral clustering e imágenes. Ejemplo 2

```
[46]: im_orig = imread('data/flower2.jpg')
      im_gray = rgb2gray(im_orig)
      pylab.figure(figsize=(15,8))
      pylab.subplot(121)
      pylab.axis('off')
      pylab.imshow(im_orig)
      pylab.subplot(122)
      plt.axis('off')
      pylab.imshow(im_gray, cmap=plt.cm.gray)
```

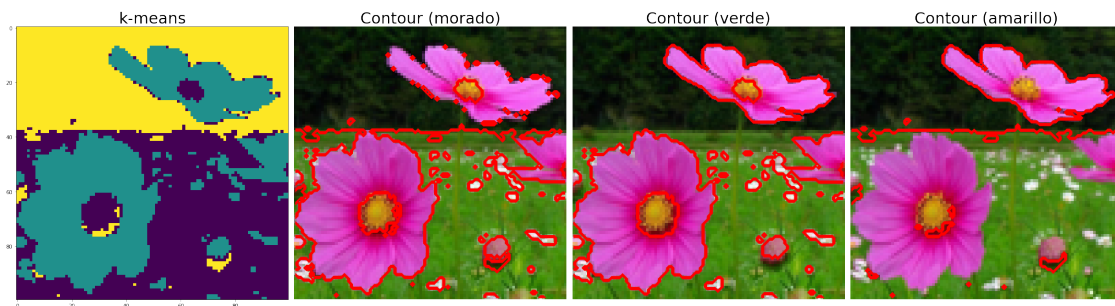
```
[46]: <matplotlib.image.AxesImage at 0x7fb1f0f2c7f0>
```




```
[47]: im = np.array(Image.fromarray(imread('data/flower2.jpg')).resize((100,100)))
      X = np.reshape(im, (-1, im.shape[-1]))
```

```
[48]: k = 3
      k_means = cluster.MinibatchKMeans(n_clusters=k, random_state=10)
      k_means.fit(X)
      y_pred = k_means.predict(X)
      labels = np.reshape(y_pred, im.shape[:2])

      pylab.figure(figsize=(30,15))
      pylab.subplot(141)
      pylab.imshow(np.reshape(y_pred, im.shape[:2])), pylab.title('k-means', size=30)
      pylab.subplot(142)
      pylab.imshow(im), pylab.contour(labels == 0, colors='red'), pylab.axis('off'),
      ↪pylab.title('Contour (morado)', size=30), pylab.tight_layout()
      pylab.subplot(143)
      pylab.imshow(im), pylab.contour(labels == 1, colors='red'), pylab.axis('off'),
      ↪pylab.title('Contour (verde)', size=30), pylab.tight_layout()
      pylab.subplot(144)
      pylab.imshow(im), pylab.contour(labels == 2, colors='red'), pylab.axis('off'),
      ↪pylab.title('Contour (amarillo)', size=30), pylab.tight_layout()
      pylab.show()
```



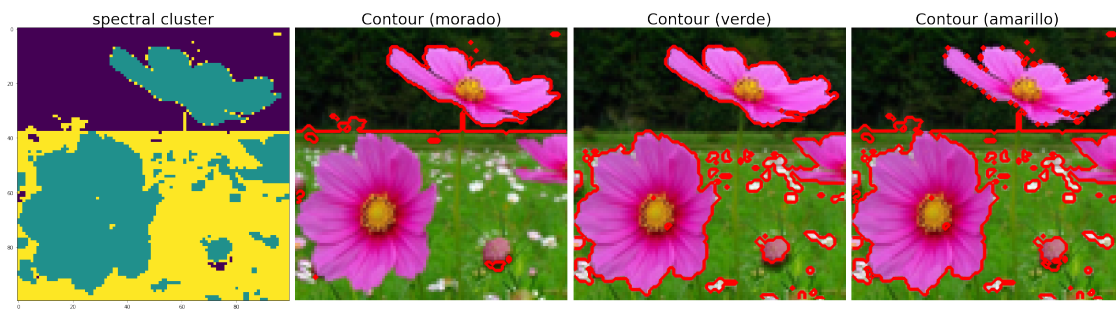
```
[30]: spectral = cluster.SpectralClustering(n_clusters=k, eigen_solver='arpack',
      ↪affinity="nearest_neighbors",
      n_neighbors=100, random_state=10)
      spectral.fit(X)
      y_pred = spectral.labels_.astype(np.int)
```

```
[31]: labels = np.reshape(y_pred, im.shape[:2])
      pylab.figure(figsize=(30,15))
      pylab.subplot(141)
```

```

pylab.imshow(np.reshape(y_pred, im.shape[:2])), pylab.title('spectral cluster',
↪size=30)
pylab.subplot(142)
pylab.imshow(im), pylab.contour(labels == 0, colors='red'), pylab.axis('off'),
↪pylab.title('Contour (morado)', size=30), pylab.tight_layout()
pylab.subplot(143)
pylab.imshow(im), pylab.contour(labels == 1, colors='red'), pylab.axis('off'),
↪pylab.title('Contour (verde)', size=30), pylab.tight_layout()
pylab.subplot(144)
pylab.imshow(im), pylab.contour(labels == 2, colors='red'), pylab.axis('off'),
↪pylab.title('Contour (amarillo)', size=30), pylab.tight_layout()
pylab.show()

```



1.3 Spectral clustering e imágenes. Ejemplo 3

```

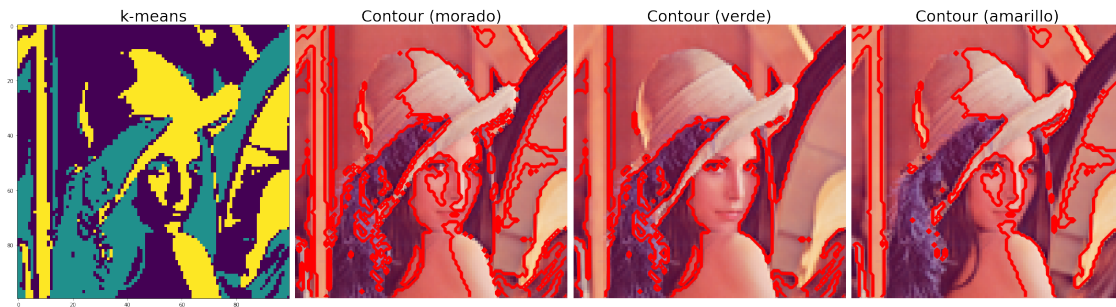
[33]: im_orig = imread('data/lena.jpg')
im_gray = rgb2gray(im_orig)
pylab.figure(figsize=(15,8))
pylab.subplot(121)
pylab.axis('off')
pylab.imshow(im_orig)
pylab.subplot(122)
plt.axis('off')
pylab.imshow(im_gray, cmap=plt.cm.gray)
plt.show()

```




```
[49]: im = np.array(Image.fromarray(imread('data/lena.jpg')).resize((100,100)))
      X = np.reshape(im, (-1, im.shape[-1]))
```

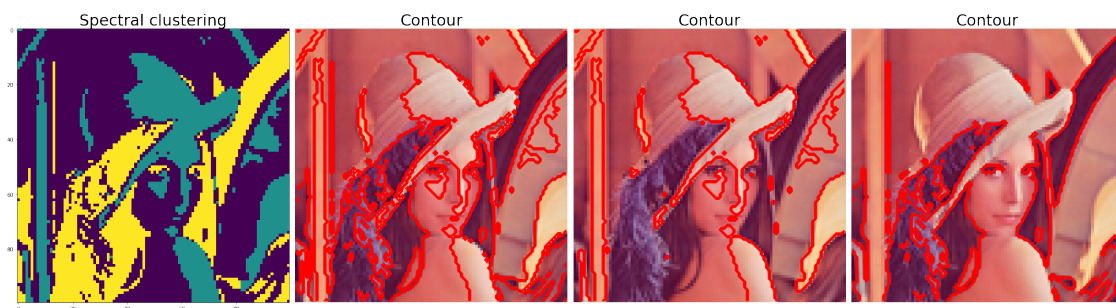
```
[50]: k = 3
      k_means = cluster.MinibatchKMeans(n_clusters=k, random_state=0)
      k_means.fit(X)
      y_pred = k_means.predict(X)
      labels = np.reshape(y_pred, im.shape[:2])
      pylab.figure(figsize=(30,15))
      pylab.subplot(141)
      pylab.imshow(np.reshape(y_pred, im.shape[:2])), pylab.title('k-means', size=30)
      pylab.subplot(142)
      pylab.imshow(im), pylab.contour(labels == 0, colors='red'), pylab.axis('off'),
      ↪pylab.title('Contour (morado)', size=30), pylab.tight_layout()
      pylab.subplot(143)
      pylab.imshow(im), pylab.contour(labels == 1, colors='red'), pylab.axis('off'),
      ↪pylab.title('Contour (verde)', size=30), pylab.tight_layout()
      pylab.subplot(144)
      pylab.imshow(im), pylab.contour(labels == 2, colors='red'), pylab.axis('off'),
      ↪pylab.title('Contour (amarillo)', size=30), pylab.tight_layout()
      pylab.show()
```



```
[36]: spectral = cluster.SpectralClustering(n_clusters=k, eigen_solver='arpack',
    ↪affinity="nearest_neighbors",
    n_neighbors=800, random_state=0)

spectral.fit(X)
y_pred = spectral.labels_.astype(np.int)

[37]: labels = np.reshape(y_pred, im.shape[:2])
pylab.figure(figsize=(30,15))
pylab.subplot(141)
pylab.imshow(np.reshape(y_pred, im.shape[:2])), pylab.title('Spectral
    ↪clustering', size=30)
pylab.subplot(142)
pylab.imshow(im), pylab.contour(labels == 0, colors='red'), pylab.axis('off'),
    ↪pylab.title('Contour', size=30), pylab.tight_layout()
pylab.subplot(143)
pylab.imshow(im), pylab.contour(labels == 1, colors='red'), pylab.axis('off'),
    ↪pylab.title('Contour', size=30), pylab.tight_layout()
pylab.subplot(144)
pylab.imshow(im), pylab.contour(labels == 2, colors='red'), pylab.axis('off'),
    ↪pylab.title('Contour', size=30), pylab.tight_layout()
pylab.show()
```



```
[38]: im_orig = imread('figs/losing_lena.jpg')
      pylab.figure(figsize=(15,10))
      pylab.axis('off')
      pylab.imshow(im_orig)
```

[38]: <matplotlib.image.AxesImage at 0x7fb1f86b3a30>

