Programming Exercise 7 - K-means Clustering and Principal Component Analysis

March 15, 2017

0.1 Programming Exercise 7 - K-means Clustering and Principal Component Analysis

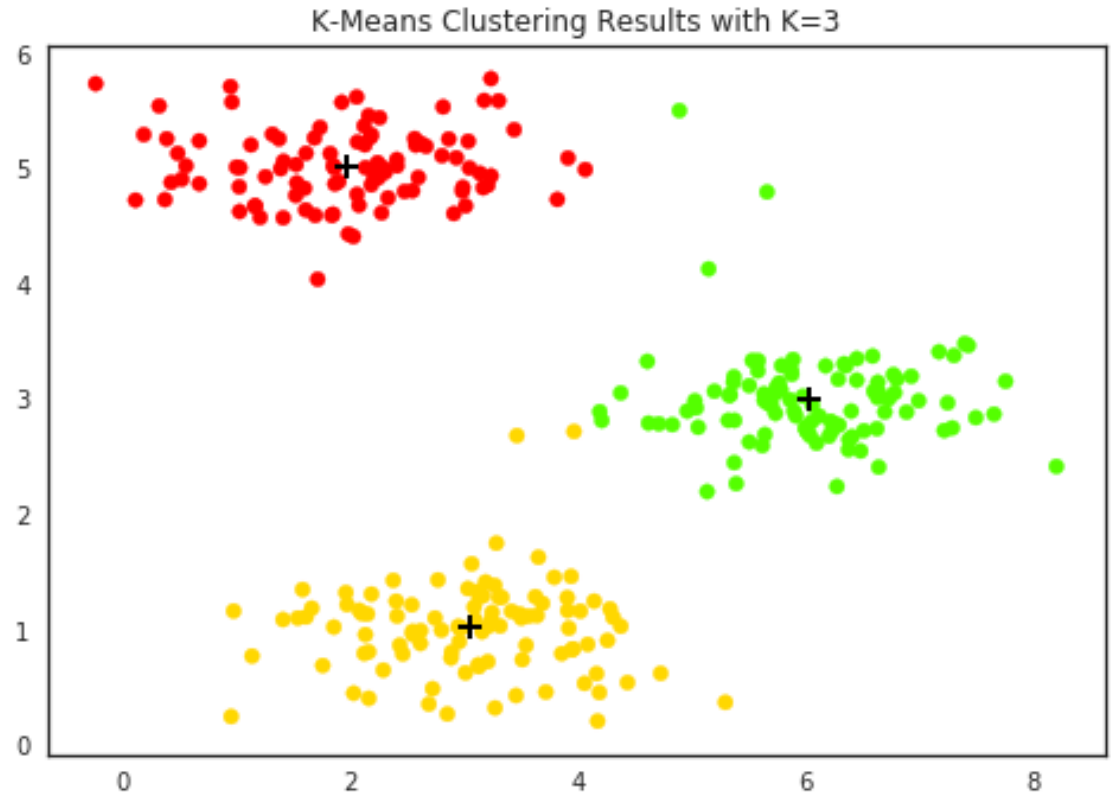
- K-means on example dataset
- Image compression with K-means>
- PCA on example data set

```
In [1]: # %load ../../../standard_import.txt
        import pandas as pd
        import numpy as np
        import matplotlib as mpl
        import matplotlib.pyplot as plt

        from scipy.io import loadmat
        from sklearn.cluster import KMeans
        from sklearn.preprocessing import StandardScaler
        from scipy import linalg

        pd.set_option('display.notebook_repr_html', False)
        pd.set_option('display.max_columns', None)
        pd.set_option('display.max_rows', 150)
        pd.set_option('display.max_seq_items', None)

        #%config InlineBackend.figure_formats = {'pdf',}
        %matplotlib inline

        import seaborn as sns
        sns.set_context('notebook')
        sns.set_style('white')
```

0.1.1 K-means on example dataset

```
In [2]: data1 = loadmat('data/ex7data2.mat')
        data1.keys()

Out[2]: dict_keys(['__header__', '__version__', '__globals__', 'X'])
```

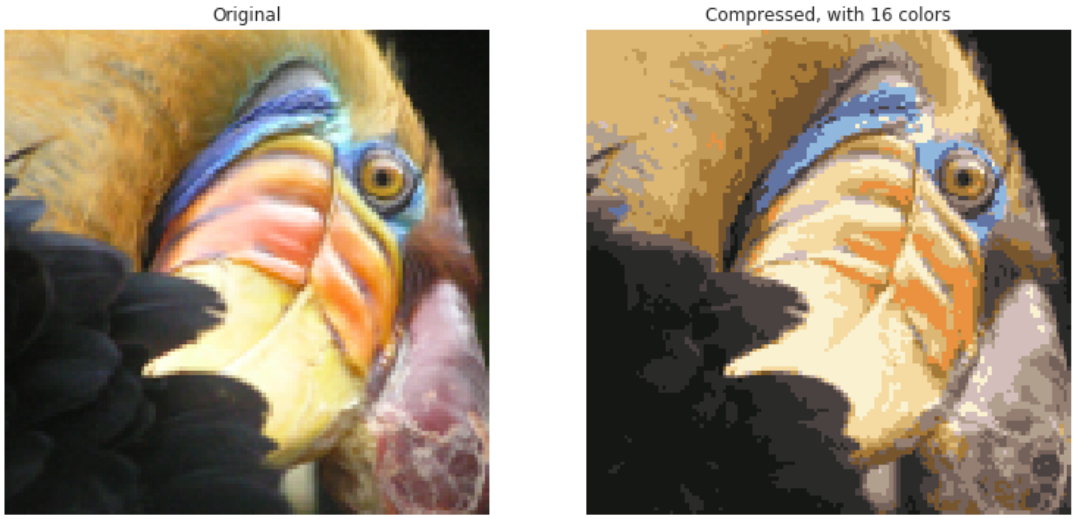
```
In [3]: X1 = data1['X']
        print('X1:', X1.shape)

X1: (300, 2)


In [4]: km1 = KMeans(3)
        km1.fit(X1)

Out[4]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
            n_clusters=3, n_init=10, n_jobs=1, precompute_distances='auto',
            random_state=None, tol=0.0001, verbose=0)

In [5]: plt.scatter(X1[:,0], X1[:,1], s=40, c=km1.labels_, cmap=plt.cm.prism)
        plt.title('K-Means Clustering Results with K=3')
        plt.scatter(km1.cluster_centers_[:,0], km1.cluster_centers_[:,1], marker='+

/home/ubuntu/anaconda3/lib/python3.6/site-packages/matplotlib/font_manager.py:1297:
  (prop.get_family(), self.defaultFamily[fontext]))
```



0.1.2 Image compression with K-means

```
In [6]: img = plt.imread('data/bird_small.png')
        img_shape = img.shape
        img_shape

Out[6]: (128, 128, 3)

In [7]: A = img/255

In [8]: AA = A.reshape(128*128,3)
        AA.shape

Out[8]: (16384, 3)

In [9]: km2 = KMeans(16)
        km2.fit(AA)

Out[9]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
            n_clusters=16, n_init=10, n_jobs=1, precompute_distances='auto',
            random_state=None, tol=0.0001, verbose=0)

In [10]: B = km2.cluster_centers_[km2.labels_].reshape(img_shape[0], img_shape[1],

In [11]: fig, (ax1, ax2) = plt.subplots(1,2, figsize=(13,9))
         ax1.imshow(img)
         ax1.set_title('Original')
         ax2.imshow(B*255)
         ax2.set_title('Compressed, with 16 colors')

         for ax in fig.axes:
             ax.axis('off')
```

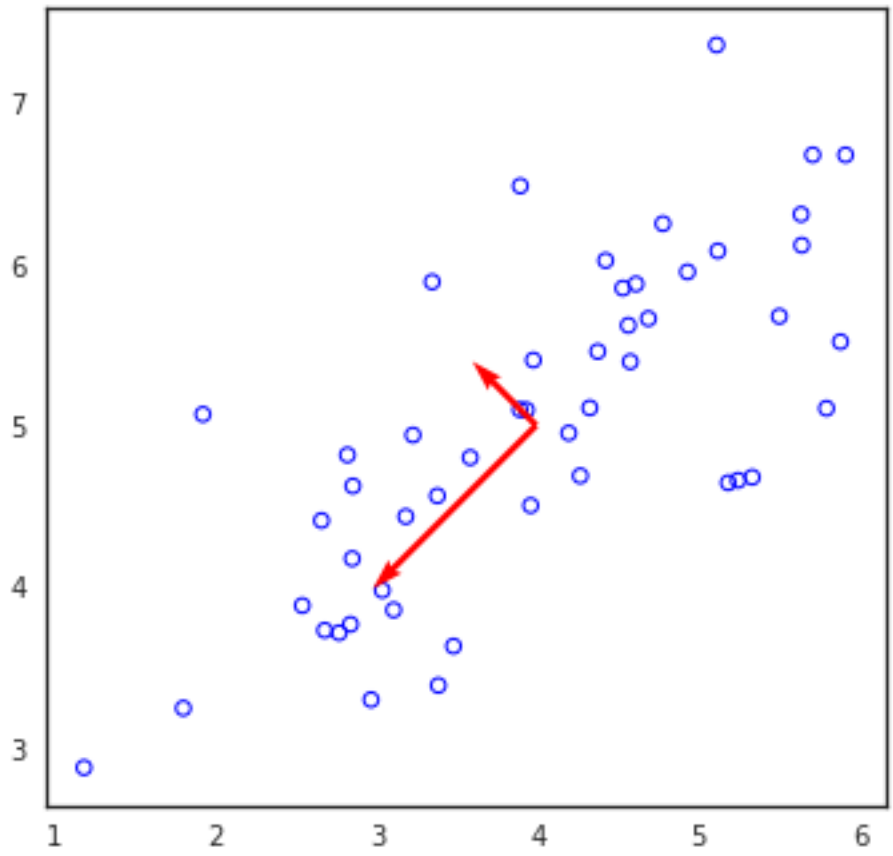
```
/home/ubuntu/anaconda3/lib/python3.6/site-packages/matplotlib/font_manager.py:1297:
  (prop.get_family(), self.defaultFamily[fontext]))
```



0.1.3 PCA on example data set

Using scipy instead of scikit-learn

```
In [12]: data2 = loadmat('data/ex7data1.mat')
         data2.keys()

Out[12]: dict_keys(['__header__', '__version__', '__globals__', 'X'])

In [13]: X2 = data2['X']
         print('X2:', X2.shape)

X2: (50, 2)


In [14]: # Standardizing the data.
         scaler = StandardScaler()
         scaler.fit(X2)

Out[14]: StandardScaler(copy=True, with_mean=True, with_std=True)

In [15]: U, S, V = linalg.svd(scaler.transform(X2).T)
         print(U)
         print(S)

[[-0.70710678 -0.70710678]
 [-0.70710678  0.70710678]]
[ 9.3153915   3.63641048]


In [16]: plt.scatter(X2[:,0], X2[:,1], s=30, edgecolors='b',facecolors='None', line
         # setting aspect ratio to 'equal' in order to show orthogonality of princi
         plt.gca().set_aspect('equal')
         plt.quiver(scaler.mean_[0], scaler.mean_[1], U[0,0], U[0,1], scale=S[1], c
         plt.quiver(scaler.mean_[0], scaler.mean_[1], U[1,0], U[1,1], scale=S[0], c

/home/ubuntu/anaconda3/lib/python3.6/site-packages/matplotlib/font_manager.py:1297:
  (prop.get_family(), self.defaultFamily[fontext]))
```


7
6
5
4
3
1
2
3
4
5
6