

In [16]:

```
%matplotlib notebook
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from sklearn.model_selection import train_test_split
fruits = pd.read_table('fruit_data_with_colors.txt')
```

In [17]:

```
from matplotlib import cm

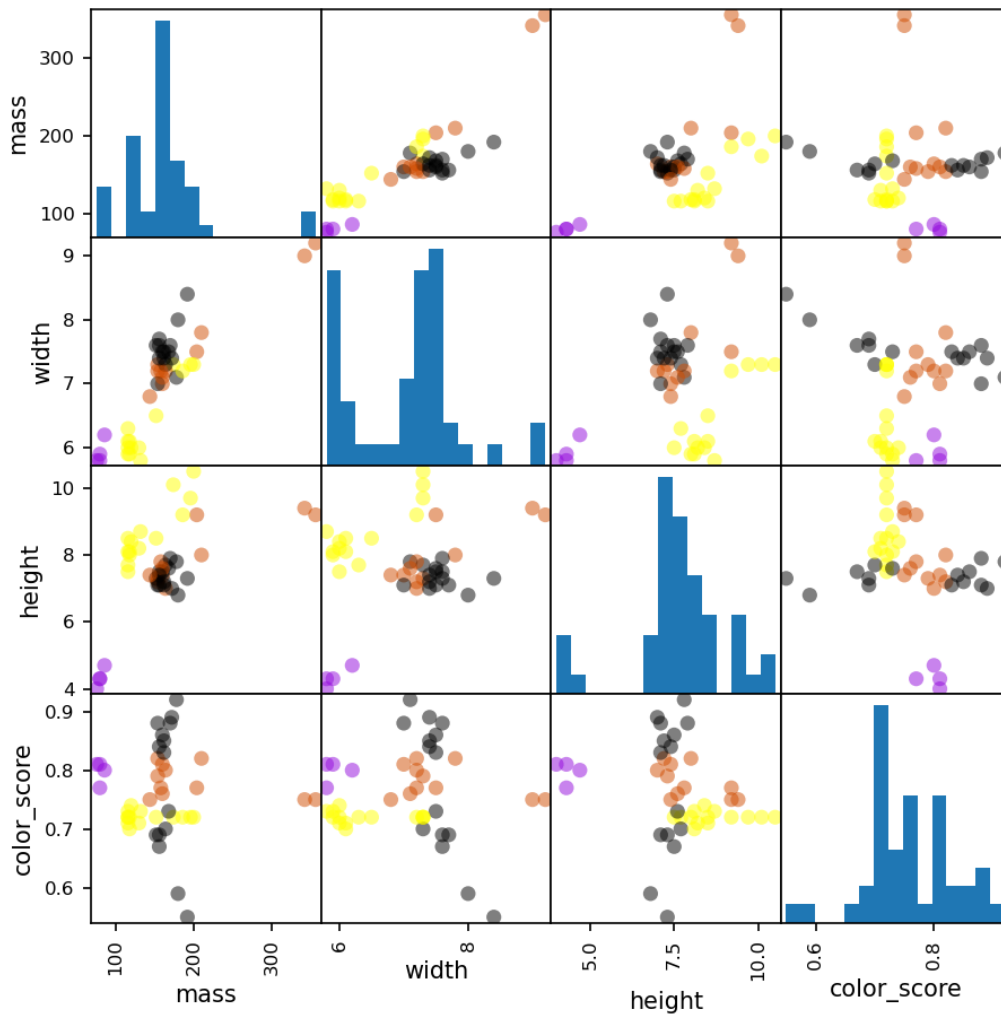
X = fruits[['mass', 'width', 'height', 'color_score']]
y = fruits['fruit_label']
X_train, X_test, y_train, y_test = train_test_split(X, y, random_state = 0)
```

In [18]:

```

cmap = cm.get_cmap('gnuplot')
scatter = pd.plotting.scatter_matrix(X_train,
                                    c = y_train, marker = 'o',
                                    s = 40,
                                    hist_kwds = {'bins' : 15},
                                    figsize = (7, 7),
                                    cmap = cmap)

```



In [19]:

```

from sklearn.neighbors import KNeighborsClassifier
knn = KNeighborsClassifier(n_neighbors = 5)
knn.fit(X_train, y_train)

```

Out[19]:

KNeighborsClassifier()

In [20]:

```
#parameter change
k_range = range(1, 20)
scores = []
for k in k_range:
    knn = KNeighborsClassifier(n_neighbors = k)
    knn.fit(X_train, y_train)
    scores.append(knn.score(X_test, y_test))
```

In [21]:

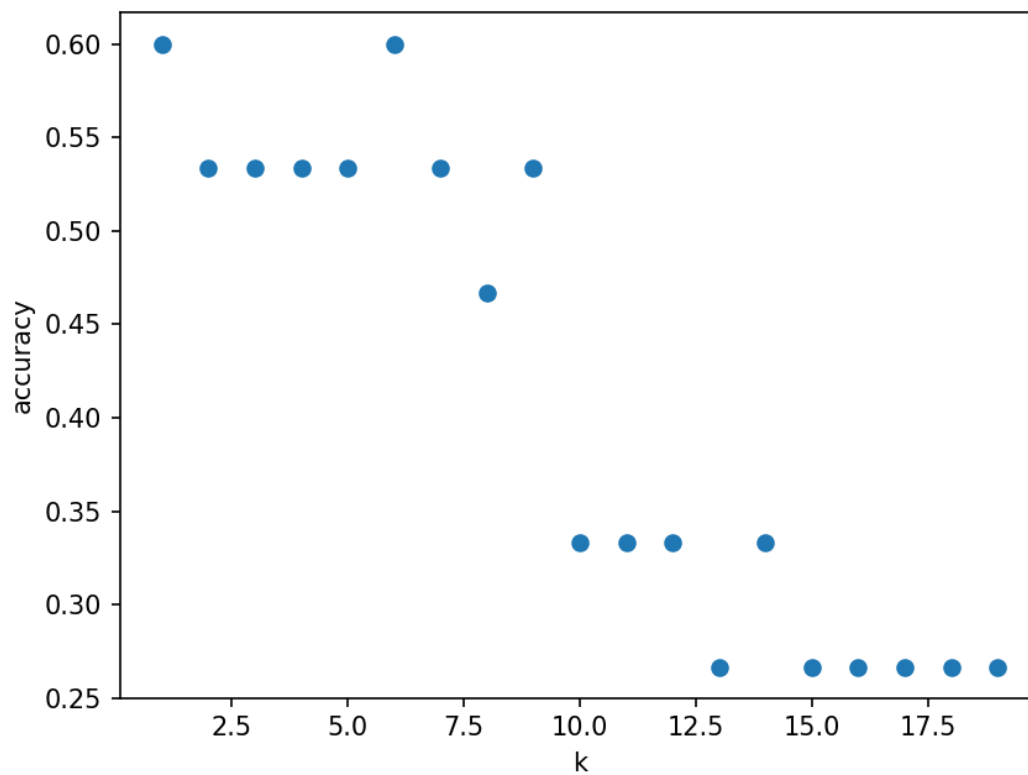
scores

Out[21]:

```
[0.6,
 0.5333333333333333,
 0.5333333333333333,
 0.5333333333333333,
 0.5333333333333333,
 0.6,
 0.5333333333333333,
 0.4666666666666667,
 0.5333333333333333,
 0.3333333333333333,
 0.3333333333333333,
 0.3333333333333333,
 0.2666666666666666,
 0.3333333333333333,
 0.2666666666666666,
 0.2666666666666666,
 0.2666666666666666,
 0.2666666666666666,
 0.2666666666666666]
```

In [22]:

```
plt.figure()  
plt.xlabel('k')  
plt.ylabel('accuracy')  
plt.scatter(k_range, scores)
```



Out[22]:

&lt;matplotlib.collections.PathCollection at 0x20d837d1370&gt;

1. 정확도는  $k=1$ ,  $k=6$ 에서 가장 좋은 성능을 보인다.
2. [무게가 120, 너비가 6, 높이가 8, color\_score가 0.7]인 과일은 레몬으로 추정한다.