```python
In [4]: import matplotlib.pyplot as plt
        from sklearn import datasets
        from sklearn.cluster import KMeans
        import pandas as pd
        import numpy as np
        iris = datasets.load_iris()
        X = pd.DataFrame(iris.data)
        X.columns = ['Sepal_Length','Sepal_Width','Petal_Length','Petal_Width']
        y = pd.DataFrame(iris.target)
        y.columns = ['Targets']
        model = KMeans(n_clusters=3)
        model.fit(X)
        plt.figure(figsize=(14,14))
        colormap = np.array(['red', 'lime', 'black'])
        plt.subplot(2, 2, 1)
        plt.scatter(X.Petal_Length, X.Petal_Width, c=colormap[y.Targets], s=40)
        plt.title('Real Clusters')
        plt.xlabel('Petal Length')
        plt.ylabel('Petal Width')
        plt.subplot(2, 2, 2)
        plt.scatter(X.Petal_Length, X.Petal_Width, c=colormap[model.labels_], s
        =40)
        plt.title('K-Means Clustering')
        plt.xlabel('Petal Length')
        plt.ylabel('Petal Width')
        from sklearn import preprocessing
        scaler = preprocessing.StandardScaler()
        scaler.fit(X)
        xsa = scaler.transform(X)
        xs = pd.DataFrame(xsa, columns = X.columns)
        from sklearn.mixture import GaussianMixture
        gmm = GaussianMixture(n_components=3)
        gmm.fit(xs)
        gmm_y = gmm.predict(xs)
        plt.subplot(2, 2, 3)
        plt.scatter(X.Petal_Length, X.Petal_Width, c=colormap[gmm_y], s=40)
```
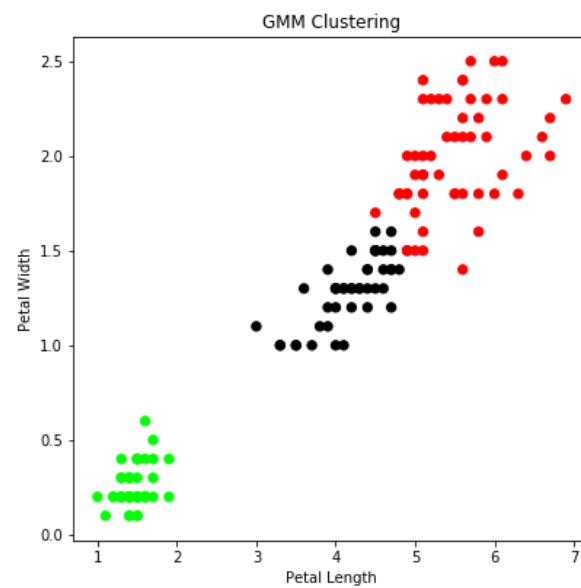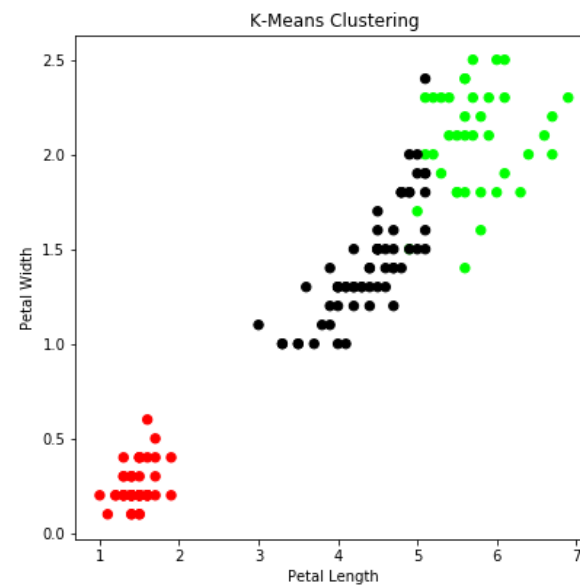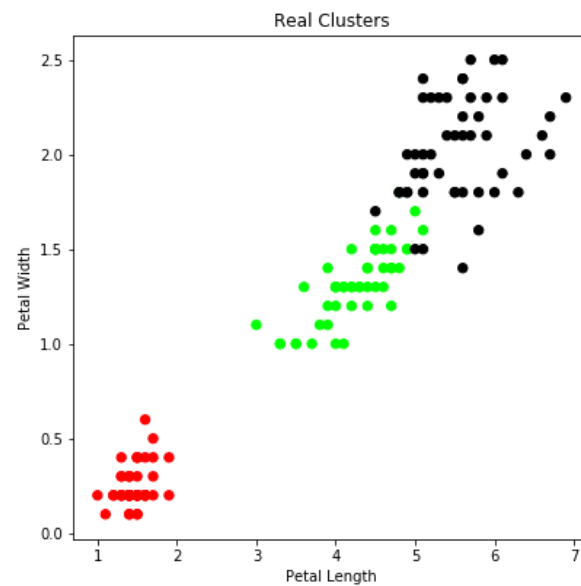
```python
plt.title('GMM Clustering')
plt.xlabel('Petal Length')
plt.ylabel('Petal Width')
print('Observation: The GMM using EM algorithm based clustering matched
 the true labels more closely than the Kmeans.')
```

Observation: The GMM using EM algorithm based clustering matched the true labels more closely than the Kmeans.

```
In [ ]:
```

```
In [ ]:
```