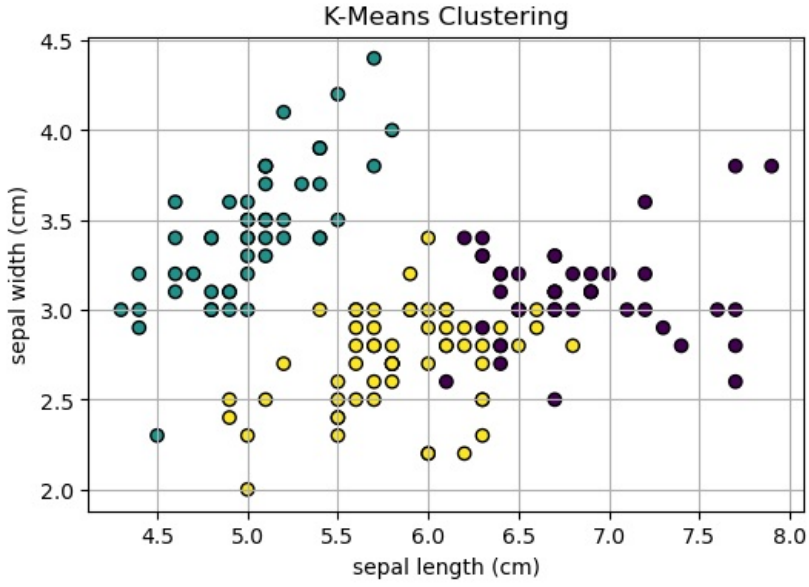
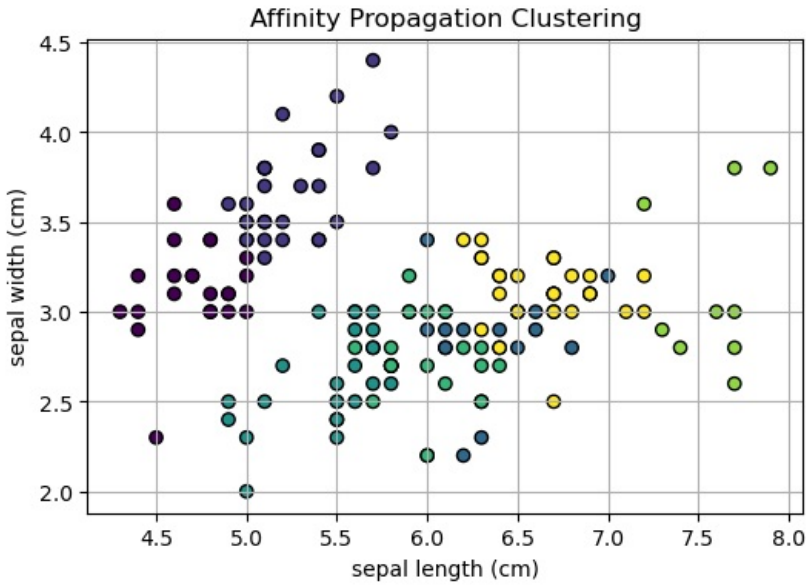
In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans, AffinityPropagation, Birch
from sklearn.metrics import silhouette_score, davies_bouldin_score, calinski_harabasz_score
from sklearn.datasets import load_iris
iris = load_iris()
data = pd.DataFrame(data=np.c_[iris['data'], iris['target']],
                    columns=iris['feature_names'] + ['target'])
selected_features = ['sepal length (cm)', 'sepal width (cm)', 'petal length (cm)', 'petal width (cm)']
X = data[selected_features]
def kmeans_clustering(X, n_clusters=3):
    model = KMeans(n_clusters=n_clusters, random_state=42)
    labels = model.fit_predict(X)
    return labels
def affinity_propagation_clustering(X):
    model = AffinityPropagation()
    labels = model.fit_predict(X)
    return labels
def birch_clustering(X, n_clusters=3):
    model = Birch(n_clusters=n_clusters)
    labels = model.fit_predict(X)
    return labels
def evaluate_clustering(X, labels, algorithm):
    silhouette = silhouette_score(X, labels)
    db_index = davies_bouldin_score(X, labels)
    ch_index = calinski_harabasz_score(X, labels)

    print(f'Evaluation Metrics for {algorithm}:')
    print(f'  Silhouette Score: {silhouette:.4f}')
    print(f'  Davies-Bouldin Index: {db_index:.4f}')
    print(f'  Calinski-Harabasz Index: {ch_index:.4f}\n')
def plot_clusters(X, labels, algorithm):
    plt.figure(figsize=(6, 4))
    plt.scatter(X.iloc[:, 0], X.iloc[:, 1], c=labels, cmap='viridis', marker='o', edgecolors='k')
    plt.title(f'{algorithm} Clustering')
    plt.xlabel(X.columns[0])
    plt.ylabel(X.columns[1])
    plt.grid(True)
    plt.show()
kmeans_labels = kmeans_clustering(X)
evaluate_clustering(X, kmeans_labels, 'K-Means')
plot_clusters(X, kmeans_labels, 'K-Means')
affinity_labels = affinity_propagation_clustering(X)
evaluate_clustering(X, affinity_labels, 'Affinity Propagation')
plot_clusters(X, affinity_labels, 'Affinity Propagation')
birch_labels = birch_clustering(X)
evaluate_clustering(X, birch_labels, 'Birch')
plot_clusters(X, birch_labels, 'Birch')
```

```
Evaluation Metrics for K-Means:
  Silhouette Score: 0.5512
  Davies-Bouldin Index: 0.6660
  Calinski-Harabasz Index: 561.5937
```

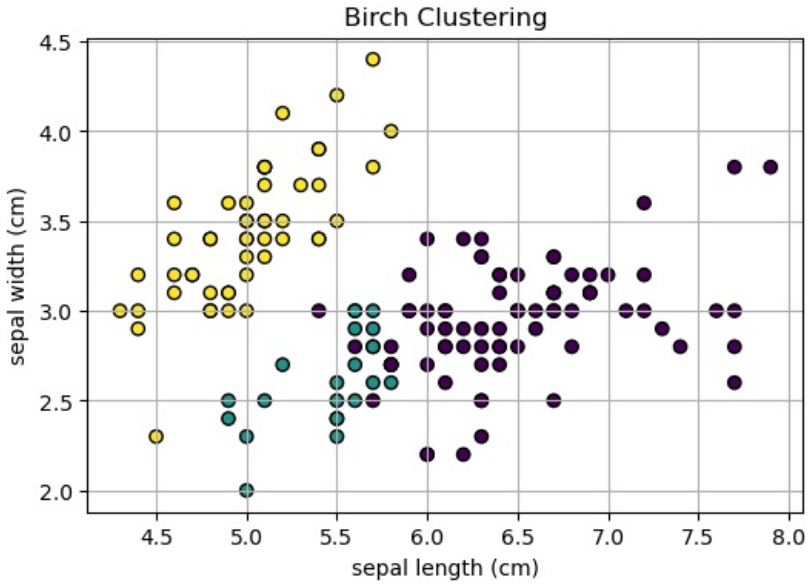
```
C:\Users\User\anaconda3\Lib\site-packages\sklearn\cluster\_kmeans.py:1419: UserWarning: KMeans is known to have a memory leak on Windows with MKL, when there are less chunks than available threads. You can avoid it by setting the environment variable OMP_NUM_THREADS=1.
  warnings.warn(
```



```
Evaluation Metrics for Affinity Propagation:
  Silhouette Score: 0.3474
  Davies-Bouldin Index: 0.9854
  Calinski-Harabasz Index: 443.7971
```




```
Evaluation Metrics for Birch:
  Silhouette Score: 0.5020
  Davies-Bouldin Index: 0.6258
  Calinski-Harabasz Index: 458.4725
```



In [ ]: