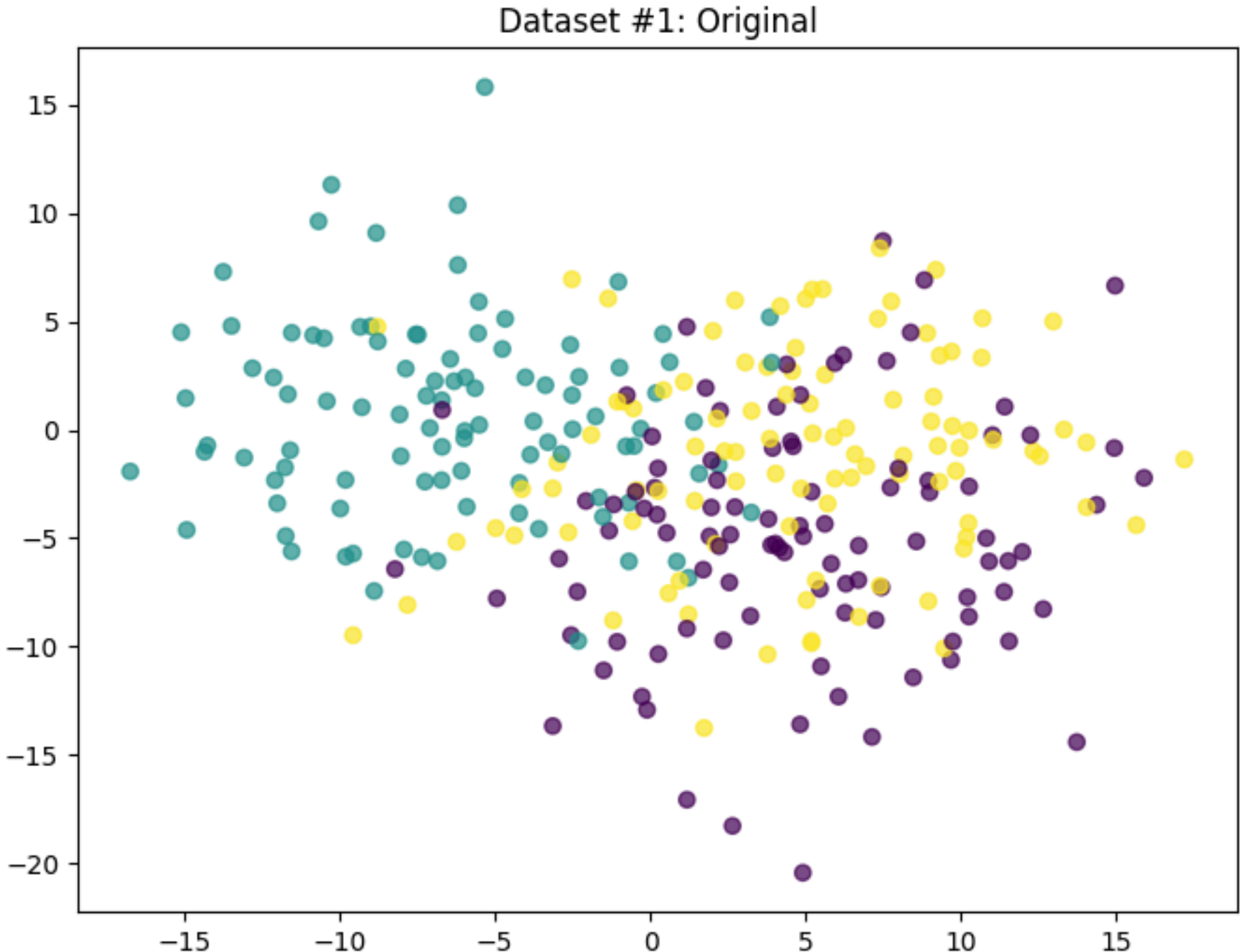


```
In [17]: from sklearn.datasets import make_blobs, make_moons
from sklearn.cluster import KMeans, AgglomerativeClustering, DBSCAN
import pandas as pd
import matplotlib.pyplot as plt
```

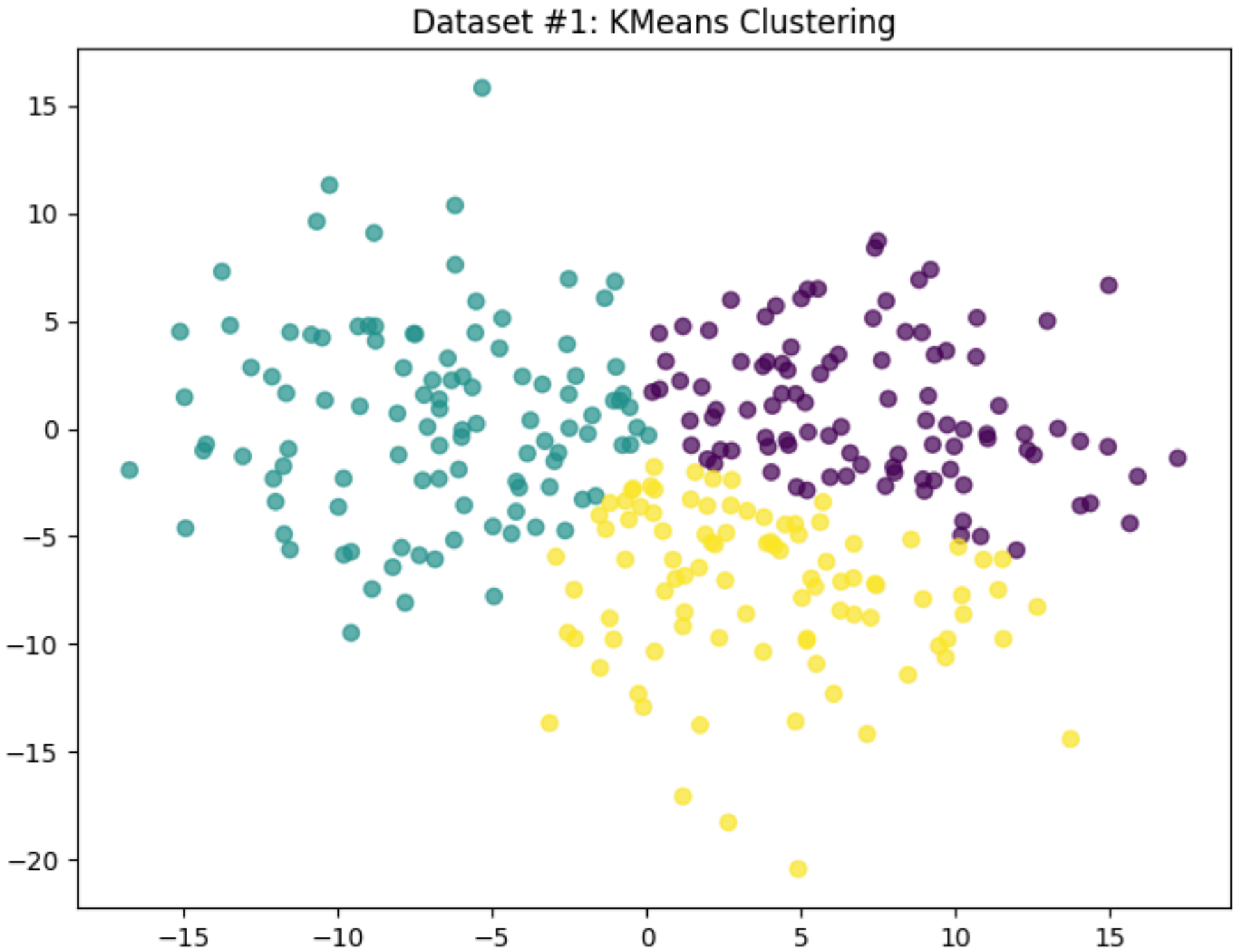
```
In [18]: # Generar datos de muestra (puedes usar make_moons si lo deseas)
X, labels = make_blobs(n_samples=300, n_features=2, centers=3, cluster_std=5, random_state=123)
```

```
In [19]: # Visualizar datos originales
plt.figure(figsize=(8, 6))
plt.scatter(X[:, 0], X[:, 1], c=labels, alpha=0.7)
plt.title('Original')
plt.show()
```



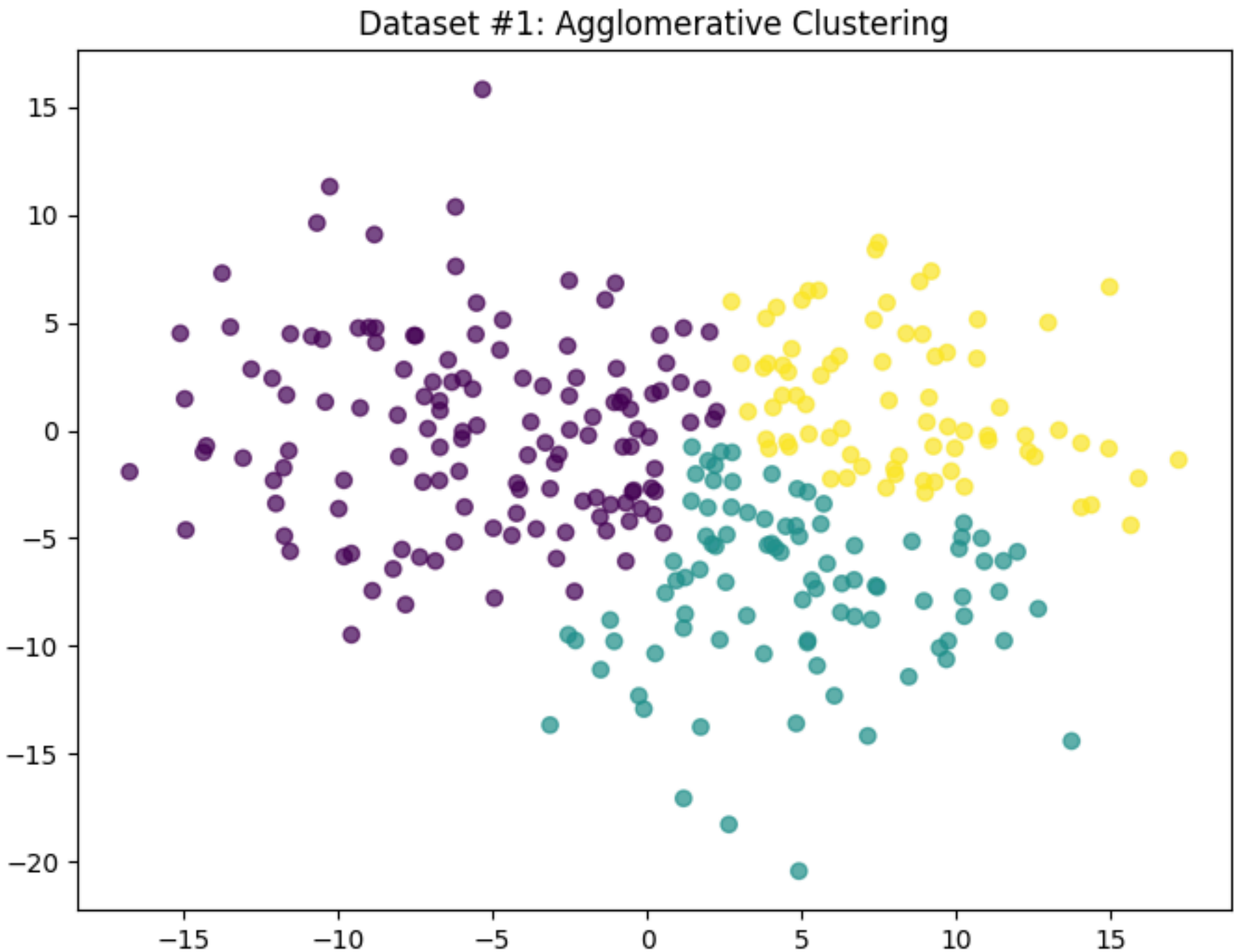
```
In [20]: # 1. KMeans
kmeans = KMeans(n_clusters=3, random_state=42)
kmeans_labels = kmeans.fit_predict(X)
```

```
In [21]: plt.figure(figsize=(8, 6))
plt.scatter(X[:, 0], X[:, 1], c=kmeans_labels, alpha=0.7)
plt.title('KMeans Clustering')
plt.show()
```



```
In [22]: # 2. Agrupamiento Aglomerativo
agglomerative = AgglomerativeClustering(n_clusters=3)
agglomerative_labels = agglomerative.fit_predict(X)
```

```
In [ ]: plt.figure(figsize=(8, 6))
plt.scatter(X[:, 0], X[:, 1], c=agglomerative_labels, alpha=0.7)
plt.title('Clustering Aglomerativo')
plt.show()
```



```
In [ ]: # 3. DBSCAN
dbscan = DBSCAN(eps=3, min_samples=5)
dbscan_labels = dbscan.fit_predict(X)
```

```
In [25]: plt.figure(figsize=(8, 6))
plt.scatter(X[:, 0], X[:, 1], c=dbscan_labels, alpha=0.7)
plt.title('DBSCAN Clustering')
plt.show()
```

