```
from google.colab import files
import pandas as pd, numpy as np, matplotlib.pyplot as plt
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.decomposition import PCA
from sklearn.cluster import AgglomerativeClustering, KMeans
from scipy.cluster.hierarchy import linkage, dendrogram
```

```
uploaded = files.upload()
df = pd.read_csv(next(iter(uploaded)))
```

Choose Files vehicle_dataset.csv

vehicle_dataset.csv(text/csv) - 2092 bytes, last modified: 11/8/2025 - 100% done

```
Saving vehicle dataset.csv to vehicle dataset.csv
```

```
for c in df.select_dtypes('object'):
    df[c] = LabelEncoder().fit_transform(df[c])
X = StandardScaler().fit_transform(df.select_dtypes(np.number))
X2 = PCA(2).fit_transform(X)
```

```
for m in ["single", "complete", "average"]:
    Z = linkage(X, m)
    plt.figure(figsize=(8,4))
    dendrogram(Z)
    plt.title(f"{m.title()} Linkage Dendrogram")
    plt.show()
    lab = AgglomerativeClustering(4, linkage=m).fit_predict(X)
    plt.scatter(X2[:,0], X2[:,1], c=lab)
    plt.title(f"Agglomerative - {m}")
    plt.show()
```

```
def div(X,k=4):
    c={0:np.arange(len(X))}; nid=1
```

```
    while len(c)<k:
        s={i:np.sum((X[v]-X[v].mean(0))**2) for i,v in c.items()}
        t=max(s,key=s.get); v=c.pop(t)
        if len(v)<=1: c[t]=v; break
        km=KMeans(2).fit(X[v])
        c[nid]=v[km.labels_==0]; c[nid+1]=v[km.labels_==1]; nid+=2
    lab=np.full(len(X),-1,int)
    for i,v in c.items(): lab[v]=i
    return np.unique(lab,return_inverse=True)[1]
```

```
lab = div(X,4)
plt.scatter(X2[:,0], X2[:,1], c=lab)
plt.title("Divisive Clustering")
plt.show()
```