In [3]:

```python
import os
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
from IPython.display import display
from sklearn import datasets

# import some data to play with
iris = datasets.load_iris()
feat = iris.feature_names
S= iris.data[:, :2]   # we take only the first two features. We could
                       # avoid this ugly slicing by using a two-dim dataset
P = iris.target
P_name = ['Setosa', 'Versicolour', 'Virginica']
from sklearn.cluster import AgglomerativeClustering
clustering = AgglomerativeClustering(linkage="complete", n_clusters=3)
clustering.fit(S);
# MinMax scale the data so that it fits nicely onto the 0.0->1.0 axes of the plot.
from sklearn import preprocessing
S_plot = preprocessing.MinMaxScaler().fit_transform(S)

colours = 'rbg'
for i in range(X.shape[0]):
    plt.text(X_plot[i, 0], X_plot[i, 1], str(clustering.labels_[i]),
            color=colours[y[i]],
            fontdict={'weight': 'bold', 'size': 9}
        )

plt.xticks([])
plt.yticks([])
plt.axis('off')
plt.show()
```
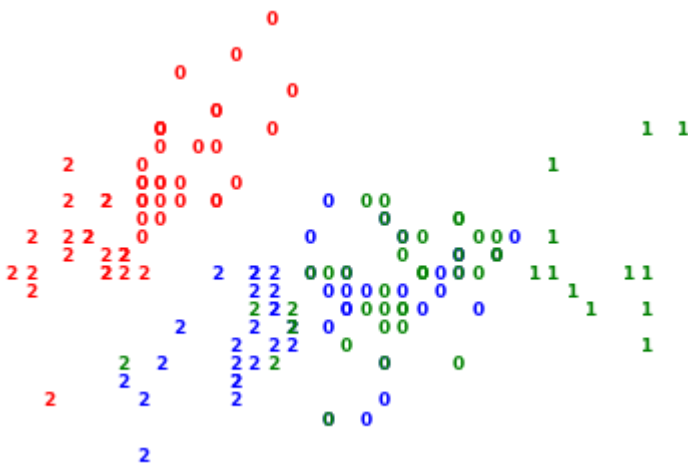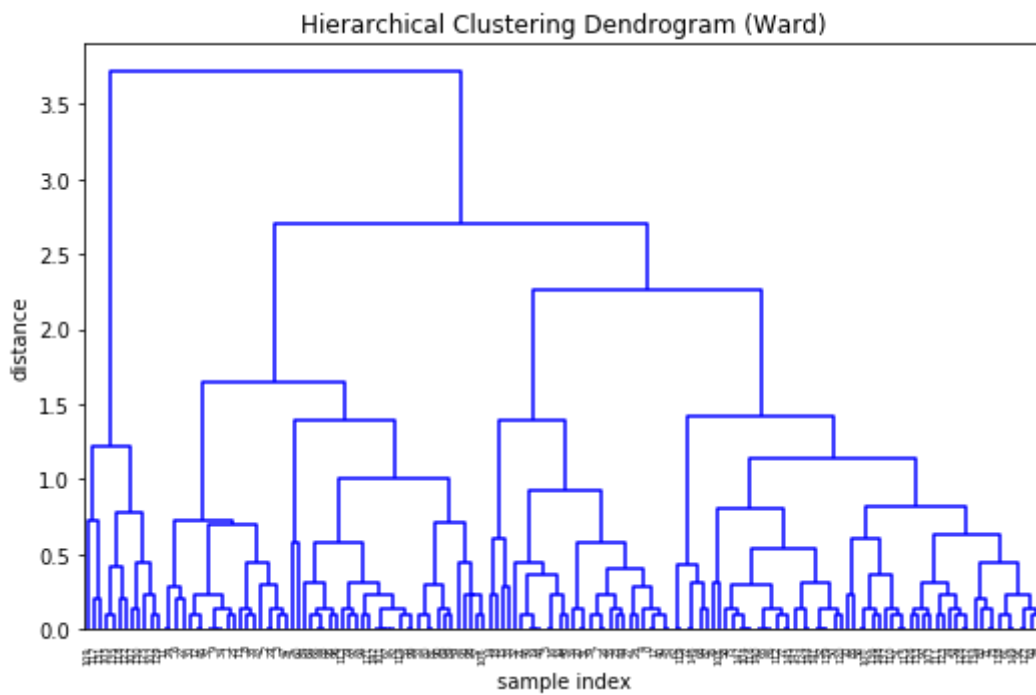
In [4]:

```python
from scipy.cluster.hierarchy import dendrogram, linkage

linkage_matrix = linkage(X, 'complete')
figure = plt.figure(figsize=(7.5, 5))
dendrogram(
    linkage_matrix,
    color_threshold=0,
)
plt.title('Hierarchical Clustering Dendrogram (Ward)')
plt.xlabel('sample index')
plt.ylabel('distance')
plt.tight_layout()
plt.show()
```


Hierarchical Clustering Dendrogram (Ward)

In [ ]: