Importing the libraries

```python
In [4]: import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
        %matplotlib inline
        import seaborn as sns
        from scipy.stats import zscore
```

```python
In [6]: data = pd.read_csv("Cust_Spend_Data.csv")
```

```python
In [8]: data.head(10)
```

Out[8]:

	Cust_ID	Name	Avg_Mthly_Spend	No_Of_Visits	Apparel_Items	FnV_Items	St
0	1	A	10000	2	1	1	
1	2	B	7000	3	0	10	
2	3	C	7000	7	1	3	
3	4	D	6500	5	1	1	
4	5	E	6000	6	0	12	
5	6	F	4000	3	0	1	
6	7	G	2500	5	0	11	
7	8	H	2500	3	0	1	
8	9	I	2000	2	0	2	
9	10	J	1000	4	0	1	

```python
In [18]: datAttr = data.iloc[:,2:]
```

```python
In [20]: datAttr.head(10)
```

	Avg_Mthly_Spend	No_Of_Visits	Apparel_Items	FnV_Items	Staples_Items
0	10000	2	1	1	0
1	7000	3	0	10	9
2	7000	7	1	3	4
3	6500	5	1	1	4
4	6000	6	0	12	3
5	4000	3	0	1	8
6	2500	5	0	11	2
7	2500	3	0	1	1
8	2000	2	0	2	2
9	1000	4	0	1	7

```python
In [22]: datscaled = datAttr.apply(zscore)
         datscaled
```
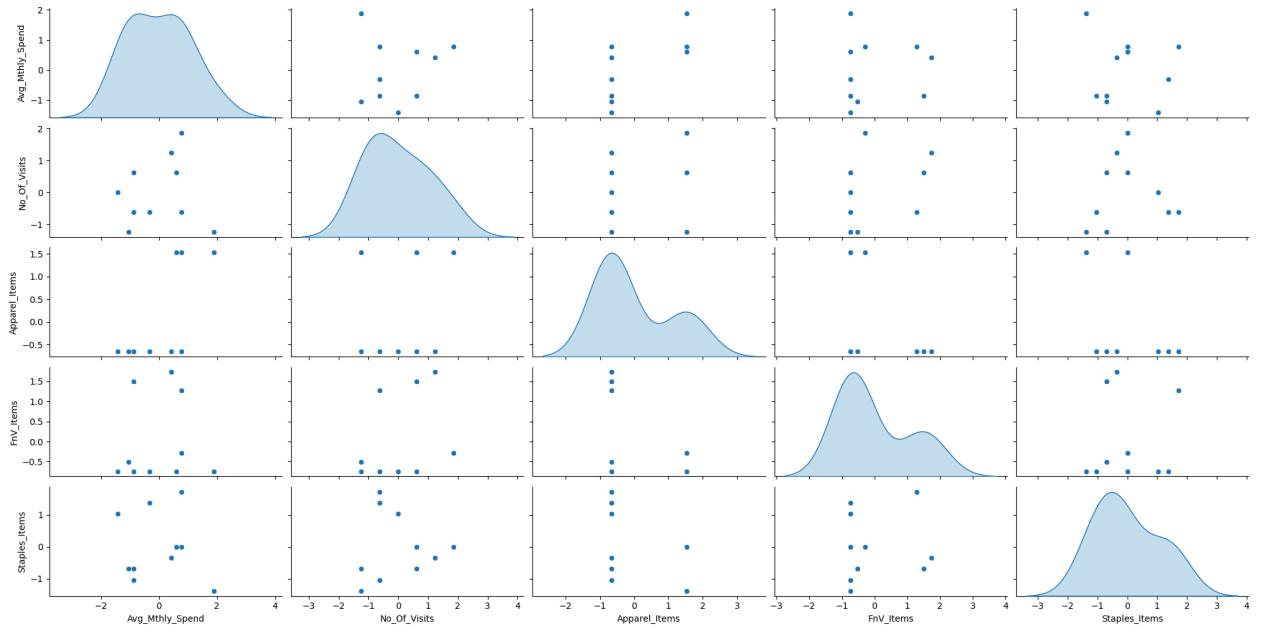
	Avg_Mthly_Spend	No_Of_Visits	Apparel_Items	FnV_Items	Staples_Items
0	1.886498	-1.240347	1.527525	-0.741433	-1.380131
1	0.787567	-0.620174	-0.654654	1.280656	1.725164
2	0.787567	1.860521	1.527525	-0.292080	0.000000
3	0.604412	0.620174	1.527525	-0.741433	0.000000
4	0.421257	1.240347	-0.654654	1.730009	-0.345033
5	-0.311364	-0.620174	-0.654654	-0.741433	1.380131
6	-0.860829	0.620174	-0.654654	1.505333	-0.690066
7	-0.860829	-0.620174	-0.654654	-0.741433	-1.035098
8	-1.043984	-1.240347	-0.654654	-0.516756	-0.690066
9	-1.410294	0.000000	-0.654654	-0.741433	1.035098

```python
In [24]: sns.pairplot(datscaled, height = 2, aspect = 2, diag_kind = "kde")
```

`<seaborn.axisgrid.PairGrid at 0x169aa2a9c50>`

In [34]:
```python
from sklearn.cluster import AgglomerativeClustering
model = AgglomerativeClustering(n_clusters = 3, affinity = "euclidean", linkag
model.fit(datscaled)
```

```
F:\Anaconda\Lib\site-packages\sklearn\cluster\_agglomerative.py:983: FutureWarn
ing: Attribute `affinity` was deprecated in version 1.2 and will be removed in
1.4. Use `metric` instead
  warnings.warn(
```

Out[34]:
▼   AgglomerativeClustering

AgglomerativeClustering(affinity='euclidean', linkage='average', n_cl
usters=3)

In [40]:
```python
datAttr["labels"] = model.labels_
datAttr.head(10)
```

Out[40]:

	Avg_Mthly_Spend	No_Of_Visits	Apparel_Items	FnV_Items	Staples_Items	lab
0	10000	2	1	1	0	
1	7000	3	0	10	9	
2	7000	7	1	3	4	
3	6500	5	1	1	4	
4	6000	6	0	12	3	
5	4000	3	0	1	8	
6	2500	5	0	11	2	
7	2500	3	0	1	1	
8	2000	2	0	2	2	
9	1000	4	0	1	7	

In [42]:
```python
datagrp = datAttr.groupby(["labels"])
datagrp.mean()
```

Out[42]:

	Avg_Mthly_Spend	No_Of_Visits	Apparel_Items	FnV_Items	Staples_Items
labels					
0	7833.333333	4.666667	1.0	1.666667	2.666667
1	5166.666667	4.666667	0.0	11.000000	4.666667
2	2375.000000	3.000000	0.0	1.250000	4.500000

In [46]:
```python
from scipy.cluster.hierarchy import cophenet, dendrogram, linkage
from scipy.spatial.distance import pdist ## pairwise distribution between data
```

In [48]:
```python
z = linkage(datscaled, metric = "euclidean", method = "average")
c, coph_dists = cophenet(z, pdist(datscaled))
c
```
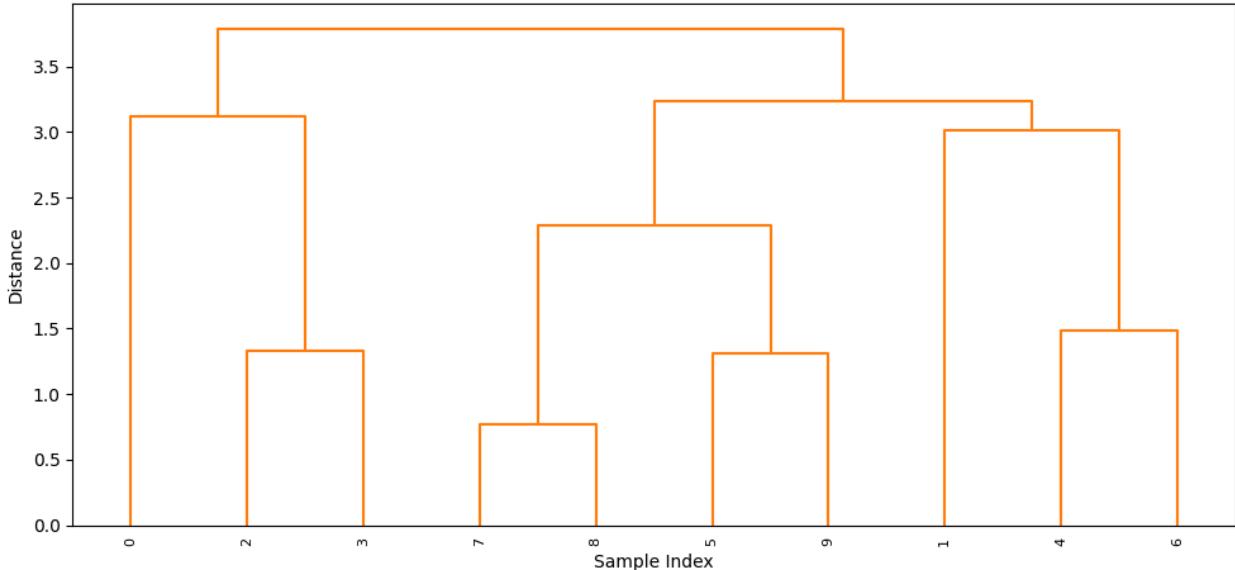
Out[48]: 0.8681149436293064

In [52]:
```python
plt.figure(figsize = (10,5))
plt.title("Agglomerative Hierarchical Clustering Dendrogram")
plt.xlabel("Sample Index")
plt.ylabel("Distance")
dendrogram(z, leaf_rotation = 90, color_threshold = 40, leaf_font_size = 8)
plt.tight_layout()
```
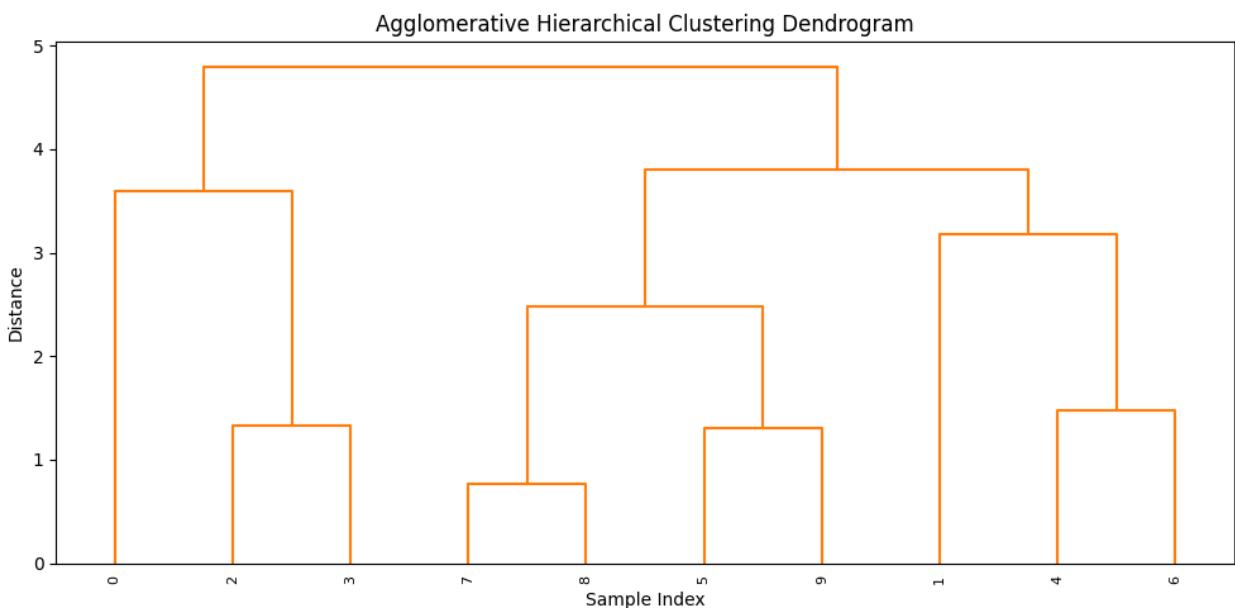
Agglomerative Hierarchical Clustering Dendrogram


```
In [54]: z = linkage(datscaled, metric = "euclidean", method = "complete")
         c, coph_dists = cophenet(z, pdist(datscaled))
         c
```

Out[54]: 0.8606955190809153

```
In [56]: plt.figure(figsize = (10,5))
         plt.title("Agglomerative Hierarchical Clustering Dendrogram")
         plt.xlabel("Sample Index")
         plt.ylabel("Distance")
         dendrogram(z, leaf_rotation = 90, color_threshold = 40, leaf_font_size = 8)
         plt.tight_layout()
```

Agglomerative Hierarchical Clustering Dendrogram
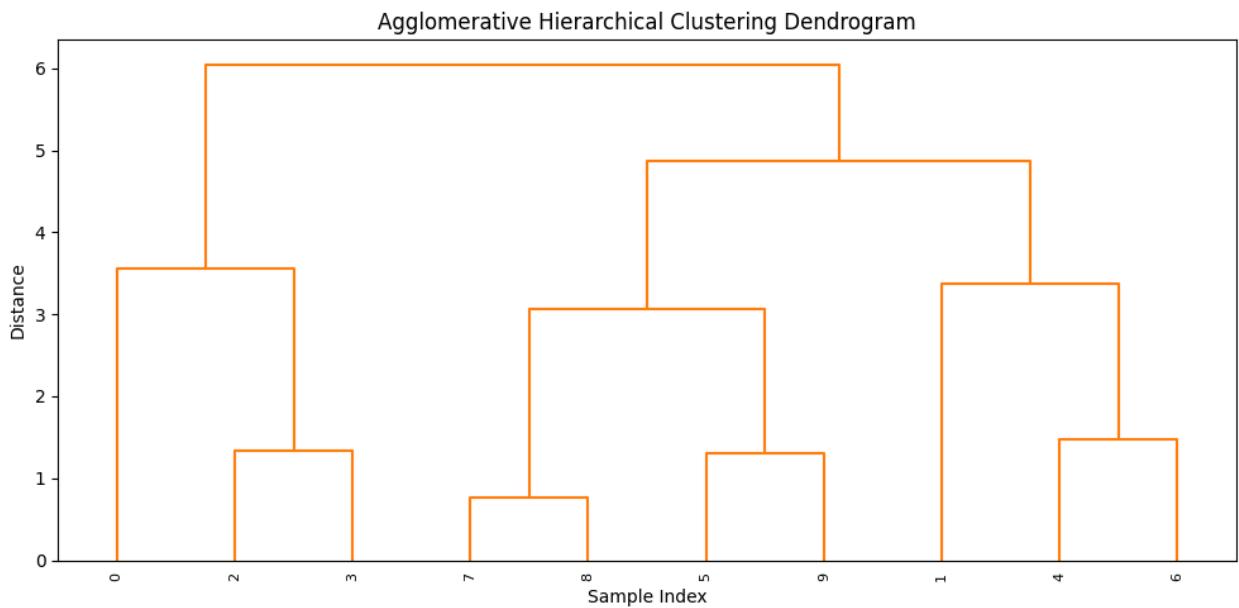

```
In [58]: z = linkage(datscaled, metric = "euclidean", method = "ward")
         c, coph_dists = cophenet(z, pdist(datscaled))
```

```
C
```

Out[58]: 0.8453818941339526

In [60]:
```python
plt.figure(figsize = (10,5))
plt.title("Agglomerative Hierarchical Clustering Dendrogram")
plt.xlabel("Sample Index")
plt.ylabel("Distance")
dendrogram(z, leaf_rotation = 90, color_threshold = 40, leaf_font_size = 8)
plt.tight_layout()
```


In [ ]:

In [ ]:

In [ ]:

In [ ]: