In [1]:

```
import random
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.preprocessing import StandardScaler
from IPython.display import display
from sklearn.cluster import KMeans
from sklearn.decomposition import PCA
from mpl_toolkits.mplot3d import Axes3D
from sklearn.metrics import homogeneity_score, completeness_score, \
v_measure_score, adjusted_rand_score, adjusted_mutual_info_score, silhouette_score
%matplotlib inline
```

In [47]:

```
train = pd.read_csv('C:\\Users\\vedan\\Desktop\\ML ASSIGNMENT\\unsupervised\\train.csv')
train1=train
test = pd.read_csv('C:\\Users\\vedan\\Desktop\\ML ASSIGNMENT\\unsupervised\\test.csv')
```

In [3]:

```
train.columns
```

Out[3]:

```
Index(['rn', 'activity', 'tBodyAcc.mean.X', 'tBodyAcc.mean.Y',
       'tBodyAcc.mean.Z', 'tBodyAcc.std.X', 'tBodyAcc.std.Y', 'tBodyAcc.std.Z',
       'tBodyAcc.mad.X', 'tBodyAcc.mad.Y',
       ...
       'fBodyBodyGyroJerkMag.meanFreq', 'fBodyBodyGyroJerkMag.skewness',
       'fBodyBodyGyroJerkMag.kurtosis', 'angle.tBodyAccMean.gravity',
       'angle.tBodyAccJerkMean.gravityMean', 'angle.tBodyGyroMean.gravityMean',
       'angle.tBodyGyroJerkMean.gravityMean', 'angle.X.gravityMean',
       'angle.Y.gravityMean', 'angle.Z.gravityMean'],
      dtype='object', length=563)
```

In [4]:

```
print('Shape of the train set: ' + str(train.shape))
print('Shape of the test set: ' + str(test.shape))
```

```
Shape of the train set: (3609, 563)
Shape of the test set: (1541, 562)
```

In [5]:

```
print('Number of duplicates in train : ',sum(train.duplicated()))
print('Number of duplicates in test : ',sum(test.duplicated()))
```

```
Number of duplicates in train :  0
Number of duplicates in test :  0
```

In [6]:

```
print('Total number of missing values in train : ', train.isna().values.sum())
print('Total number of missing values in test : ', test.isna().values.sum())
```

```
Total number of missing values in train :  0
Total number of missing values in test :  0
```

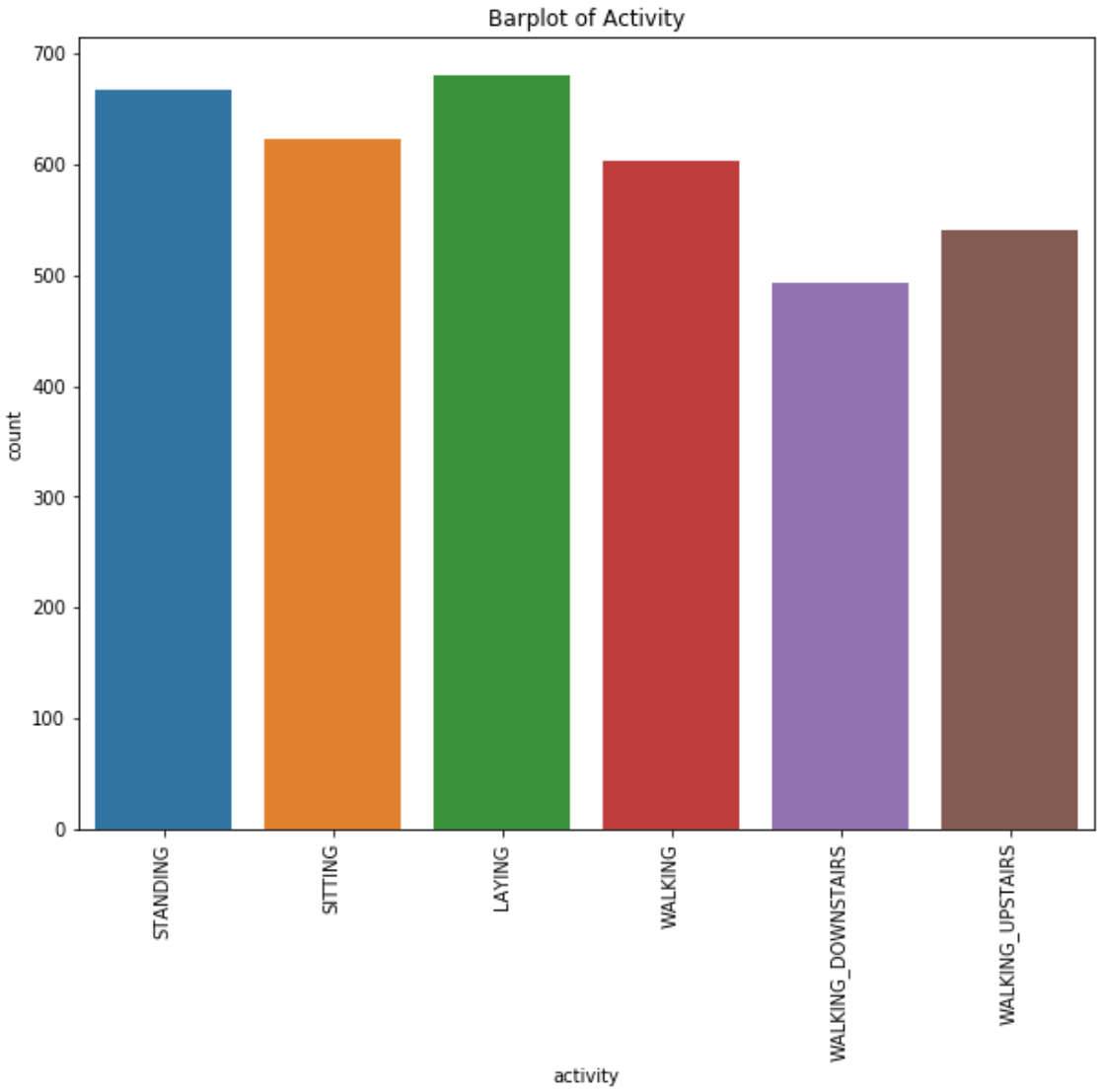
In [7]:

```
#checking for class imbalance
plt.figure(figsize=(10,8))
plt.title('Barplot of Activity')
```

```
sns.countplot(train.activity)
plt.xticks(rotation=90)
```

Out[7]:

```
(array([0, 1, 2, 3, 4, 5]), <a list of 6 Text xticklabel objects>)
```



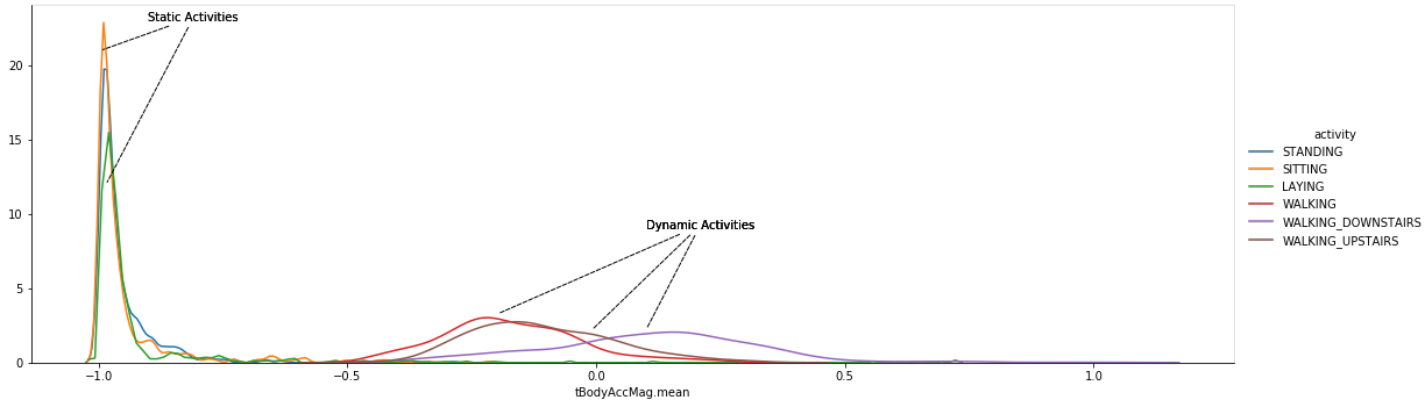
Static and dynamic activities : SITTING, STANDING, LAYING can be considered as static activities with no motion involved WALKING, WALKING_DOWNSTAIRS, WALKING_UPSTAIRS can be considered as dynamic activities with significant amount of motion involved tBodyAccMag-mean() feature used to differentiate among these two broader set of activities. robability density function(PDF) is very helpful to assess importance of a continuous variable.

In [8]:

```
facetgrid = sns.FacetGrid(train, hue='activity', height=5,aspect=3)
facetgrid.map(sns.distplot,'tBodyAccMag.mean', hist=False).add_legend()
plt.annotate("Static Activities", xy=(-.996,21), xytext=(-0.9, 23),arrowprops={'arrowstyle': '-', 'ls': 'dashed'})
plt.annotate("Static Activities", xy=(-.999,26), xytext=(-0.9, 23),arrowprops={'arrowstyle': '-', 'ls': 'dashed'})
plt.annotate("Static Activities", xy=(-0.985,12), xytext=(-0.9, 23),arrowprops={'arrowstyle': '-', 'ls': 'dashed'})
plt.annotate("Dynamic Activities", xy=(-0.2,3.25), xytext=(0.1, 9),arrowprops={'arrowstyle': '-', 'ls': 'dashed'})
plt.annotate("Dynamic Activities", xy=(0.1,2.18), xytext=(0.1, 9),arrowprops={'arrowstyle': '-', 'ls': 'dashed'})
plt.annotate("Dynamic Activities", xy=(-0.01,2.15), xytext=(0.1, 9),arrowprops={'arrowstyle': '-', 'ls': 'dashed'})
```

Out[8]:

```
Text(0.1, 9, 'Dynamic Activities')
```



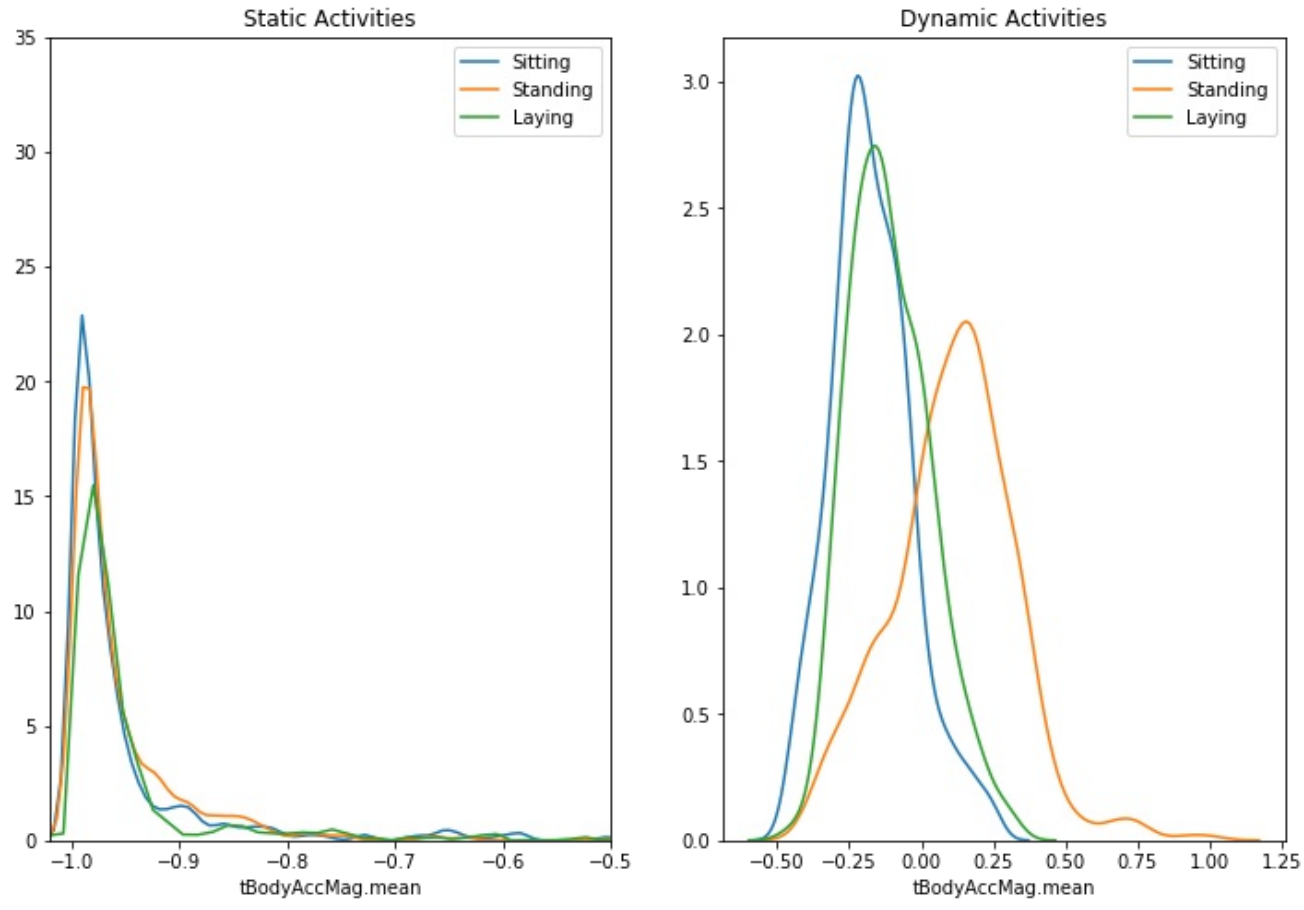
closer view on the PDFs of each activity under static and dynamic categorization

In [9]:

```
plt.figure(figsize=(12,8))
plt.subplot(1,2,1)
plt.title("Static Activities")
sns.distplot(train[train["activity"]=="SITTING"]['tBodyAccMag.mean'],hist = False, label
= 'Sitting')
sns.distplot(train[train["activity"]=="STANDING"]['tBodyAccMag.mean'],hist = False,label
= 'Standing')
sns.distplot(train[train["activity"]=="LAYING"]['tBodyAccMag.mean'],hist = False, label =
'Laying')
plt.axis([-1.02, -0.5, 0, 35])
plt.subplot(1,2,2)
plt.title("Dynamic Activities")
sns.distplot(train[train["activity"]=="WALKING"]['tBodyAccMag.mean'],hist = False, label
= 'Sitting')
sns.distplot(train[train["activity"]=="WALKING_DOWNSTAIRS"]['tBodyAccMag.mean'],hist = Fa
lse,label = 'Standing')
sns.distplot(train[train["activity"]=="WALKING_UPSTAIRS"]['tBodyAccMag.mean'],hist = Fals
e, label = 'Laying')
```

Out[9]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x285734d15c8>
```



plot the boxplot of Body Accelartion Magnitude mean(tBodyAccMag-mean()) across all the six categories.

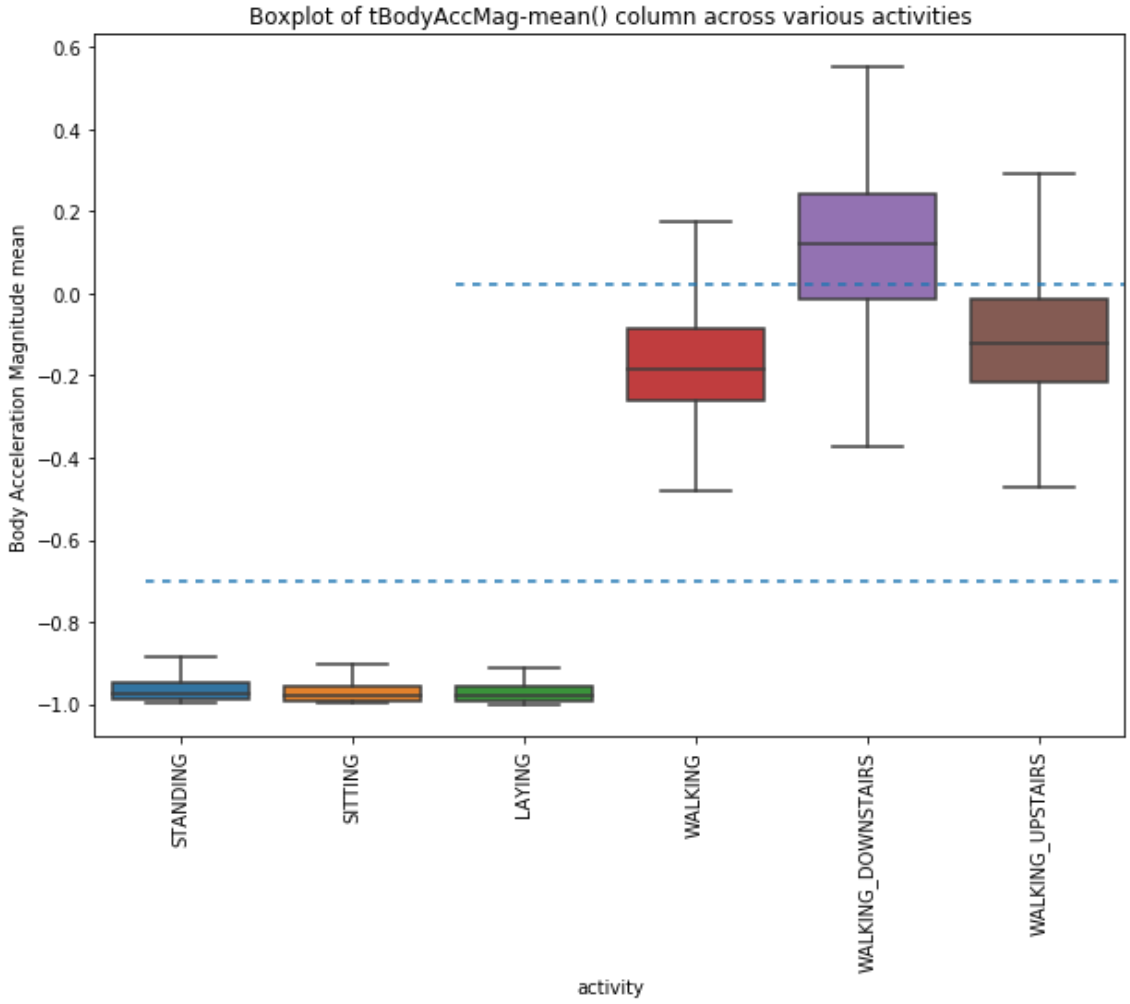
tBodyAccMag-mean()<=-0.8---------->Static tBodyAccMag-mean()>=-0.6---------->dynamic

In [10]:

```
plt.figure(figsize=(10,7))
sns.boxplot(x='activity', y='tBodyAccMag.mean',data=train, showfliers=False)
plt.ylabel('Body Acceleration Magnitude mean')
plt.title("Boxplot of tBodyAccMag-mean() column across various activities")
plt.axhline(y=-0.7, xmin=0.05,dashes=(3,3))
plt.axhline(y=0.020, xmin=0.35, dashes=(3,3))
plt.xticks(rotation=90)
```

Out[10]:

```
(array([0, 1, 2, 3, 4, 5]), <a list of 6 Text xticklabel objects>)
```



Analysing Angle between X-axis and gravityMean feature

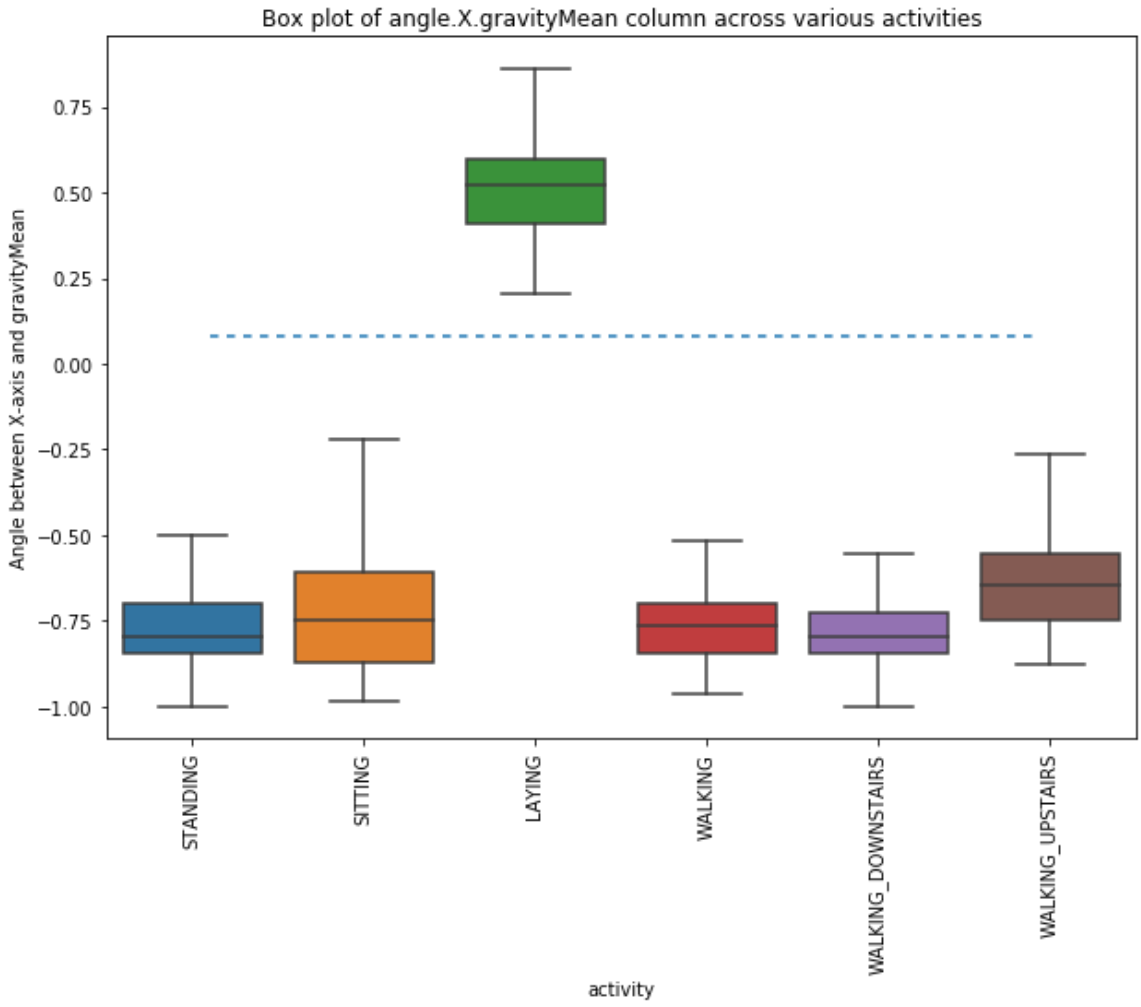
From the boxplot we can observe that angle(X,gravityMean) perfectly seperates LAYING from other activities.

In [11]:

```
plt.figure(figsize=(10,7))
sns.boxplot(x='activity', y='angle.X.gravityMean', data=train, showfliers=False)
plt.axhline(y=0.08, xmin=0.1, xmax=0.9,dashes=(3,3))
plt.ylabel("Angle between X-axis and gravityMean")
plt.title('Box plot of angle.X.gravityMean column across various activities')
plt.xticks(rotation = 90)
```

Out[11]:

```
(array([0, 1, 2, 3, 4, 5]), <a list of 6 Text xticklabel objects>)
```



Using t-SNE data can be visualized from a extremely high dimensional space to a low dimensional space and still it retains lots of actual information. Given training data has 561 unqiue features, using t-SNE let's visualize it to a 2D space.

In [12]:

```
from sklearn.manifold import TSNE
```

In [13]:

```
X_for_tsne = train.drop(['rn', 'activity'], axis=1)
```

In [14]:

```
%time
tsne = TSNE(random_state = 42, n_components=2, verbose=1, perplexity=50, n_iter=1000).fi
t_transform(X_for_tsne)
```

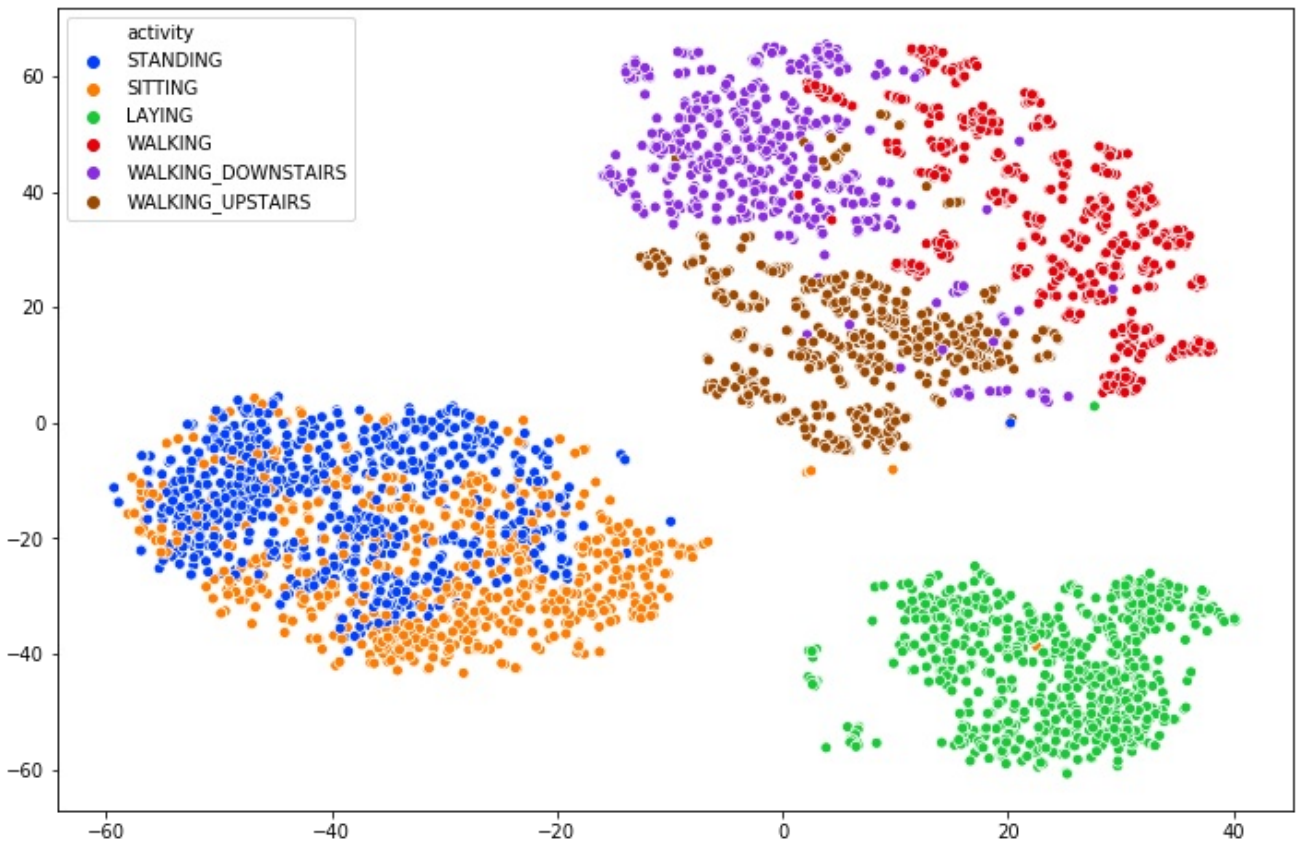
```
Wall time: 0 ns
[t-SNE] Computing 151 nearest neighbors...
[t-SNE] Indexed 3609 samples in 0.078s...
[t-SNE] Computed neighbors for 3609 samples in 8.232s...
[t-SNE] Computed conditional probabilities for sample 1000 / 3609
[t-SNE] Computed conditional probabilities for sample 2000 / 3609
[t-SNE] Computed conditional probabilities for sample 3000 / 3609
[t-SNE] Computed conditional probabilities for sample 3609 / 3609
[t-SNE] Mean sigma: 1.502022
[t-SNE] KL divergence after 250 iterations with early exaggeration: 67.530914
[t-SNE] KL divergence after 1000 iterations: 1.166608
```

In [15]:

```
plt.figure(figsize=(12,8))
sns.scatterplot(x =tsne[:, 0], y = tsne[:, 1], hue = train["activity"],palette="bright")
```

Out[15]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x28572324408>
```



k-means

In [16]:

```
Labels = train['activity']
train = train.drop(['rn', 'activity'], axis = 1)
test = test.drop(['rn'], axis = 1)
Labels_keys = Labels.unique().tolist()
Labels = np.array(Labels)
print('Activity labels: ' + str(Labels_keys))
```

```
Activity labels: ['STANDING', 'SITTING', 'LAYING', 'WALKING', 'WALKING_DOWNSTAIRS', 'WALK
ING_UPSTAIRS']
```

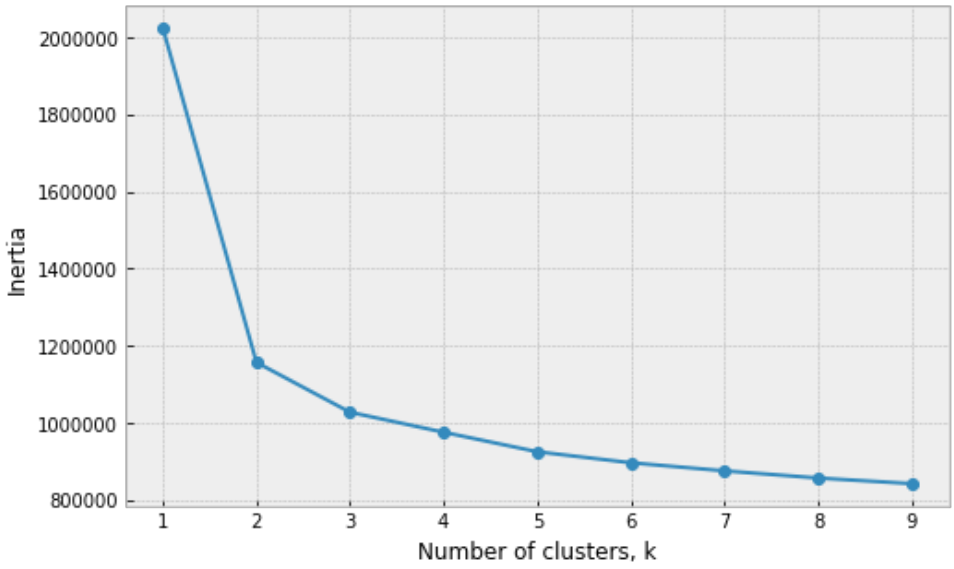
In [17]:

```
#normalize the dataset
scaler = StandardScaler()
train = scaler.fit_transform(train)
```

In [18]:

```
ks = range(1, 10)
inertias = []

for k in ks:
    model = KMeans(n_clusters=k)
    model.fit(train)
    inertias.append(model.inertia_)

plt.figure(figsize=(8,5))
plt.style.use('bmh')
plt.plot(ks, inertias, '-o')
plt.xlabel('Number of clusters, k')
plt.ylabel('Inertia')
plt.xticks(ks)
```

```
plt.show()
```



In [19]:

```
def k_means(n_clust, data_frame, true_labels):
    k_means = KMeans(n_clusters = n_clust, random_state=123, n_init=30)
    k_means.fit(data_frame)
    c_labels = k_means.labels_
    df = pd.DataFrame({'clust_label': c_labels, 'orig_label': true_labels.tolist()})
    ct = pd.crosstab(df['clust_label'], df['orig_label'])
    y_clust = k_means.predict(data_frame)
    #y_clust_1 = k_means.predict(test)
    display(ct)
```

In [20]:

```
k_means(n_clust=2, data_frame=train, true_labels=Labels)
```

orig_label	LAYING	SITTING	STANDING	WALKING	WALKING_DOWNSTAIRS	WALKING_UPSTAIRS
clust_label						
0	680	622	668	0	0	6
1	1	1	0	603	493	535

In [21]:

```
k_means(n_clust=6, data_frame=train, true_labels=Labels)
```

orig_label	LAYING	SITTING	STANDING	WALKING	WALKING_DOWNSTAIRS	WALKING_UPSTAIRS
clust_label						
0	554	21	0	0	0	0
1	0	0	0	248	311	97
2	1	0	0	329	107	438
3	20	445	479	0	0	0
4	0	0	0	26	75	4
5	106	157	189	0	0	2

In [22]:

```
Labels_binary = Labels.copy()
for i in range(len(Labels_binary)):
    if (Labels_binary[i] == 'STANDING' or Labels_binary[i] == 'SITTING' or Labels_binary
[i] == 'LAYING'):
        Labels_binary[i] = 0
```

```
    else:
        Labels_binary[i] = 1
Labels_binary = np.array(Labels_binary.astype(int))
```

In [23]:

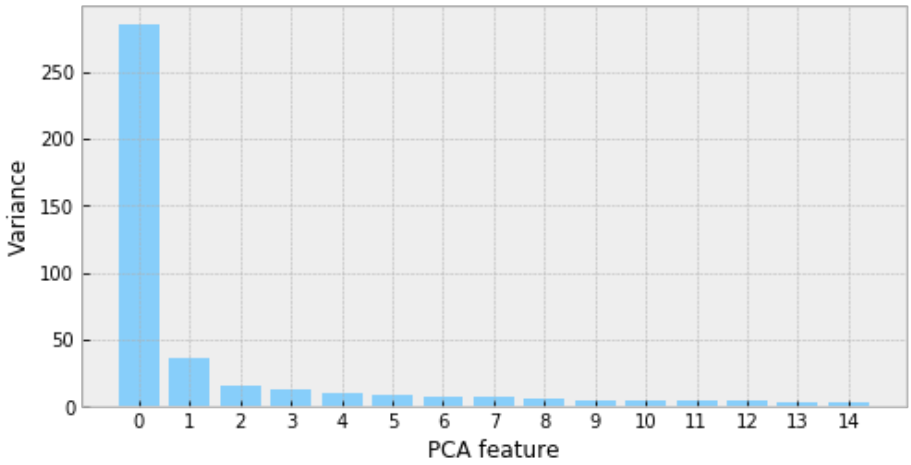
```
k_means(n_clust=2, data_frame=train, true_labels=Labels_binary)
```

orig_label	0	1
clust_label		
0	1970	6
1	2	1631

PCA

In [24]:

```
#check for optimal number of features
pca = PCA(random_state=123)
pca.fit(train)
features = range(pca.n_components_)

plt.figure(figsize=(8,4))
plt.bar(features[:15], pca.explained_variance_[:15], color='lightskyblue')
plt.xlabel('PCA feature')
plt.ylabel('Variance')
plt.xticks(features[:15])
plt.show()
```



In [25]:

```
def pca_transform(n_comp):
    pca = PCA(n_components=n_comp, random_state=123)
    global Data_reduced
    Data_reduced = pca.fit_transform(train)
    print('Shape of the new Data df: ' + str(Data_reduced.shape))
```

In [26]:

```
pca_transform(n_comp=1)
k_means(n_clust=2, data_frame=Data_reduced, true_labels=Labels_binary)
```

```
Shape of the new Data df: (3609, 1)
```

orig_label	0	1
clust_label		
0	1971	8
1	1	1629

In [27]:

```
pca_transform(n_comp=2)
k_means(n_clust=2, data_frame=Data_reduced, true_labels=Labels_binary)
```

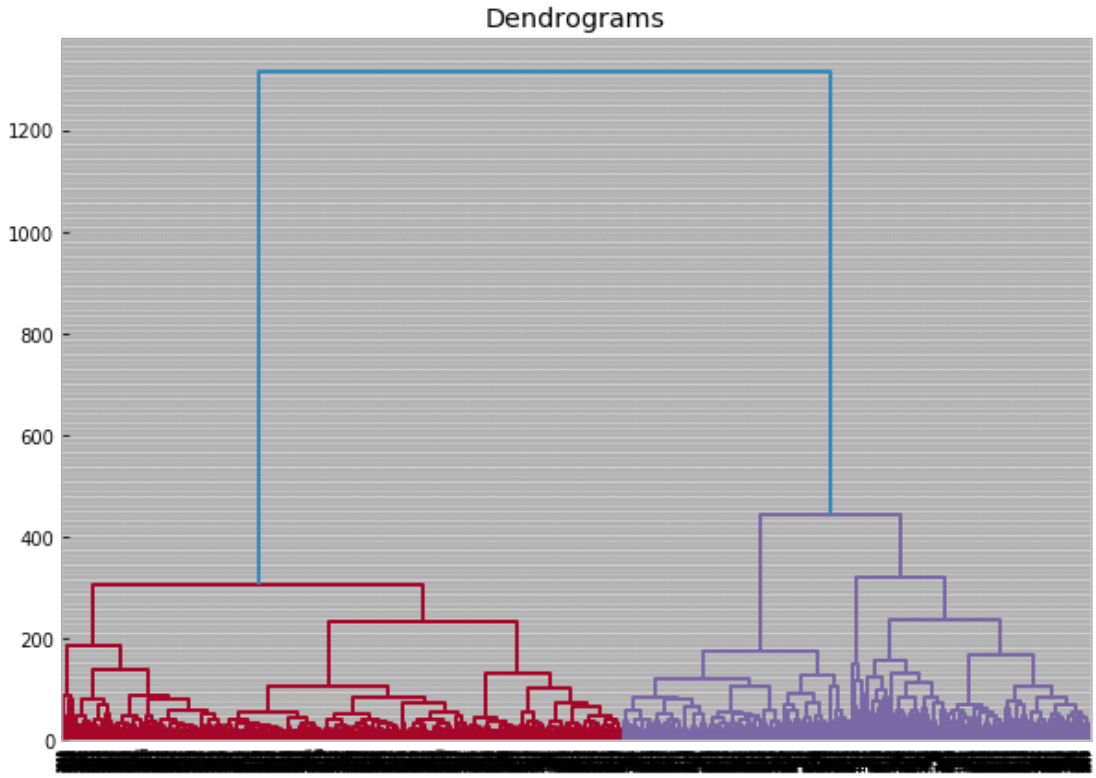
```
Shape of the new Data df: (3609, 2)
```

orig_label	0	1
clust_label		
0	1969	6
1	3	1631

Hierarchical Clustering

In [28]:

```
import scipy.cluster.hierarchy as shc
plt.figure(figsize=(10, 7))
plt.title("Dendrograms")
dend = shc.dendrogram(shc.linkage(train, method='ward'))
```



In [29]:

```
from sklearn.cluster import AgglomerativeClustering
hc = AgglomerativeClustering(n_clusters=6, affinity = 'euclidean', linkage = 'ward')
hc.fit(train)
c_labels = hc.labels_
df = pd.DataFrame({'clust_label': c_labels, 'orig_label': Labels.tolist()})
ct = pd.crosstab(df['clust_label'], df['orig_label'])
y_clust = hc.fit_predict(train)
display(ct)
```

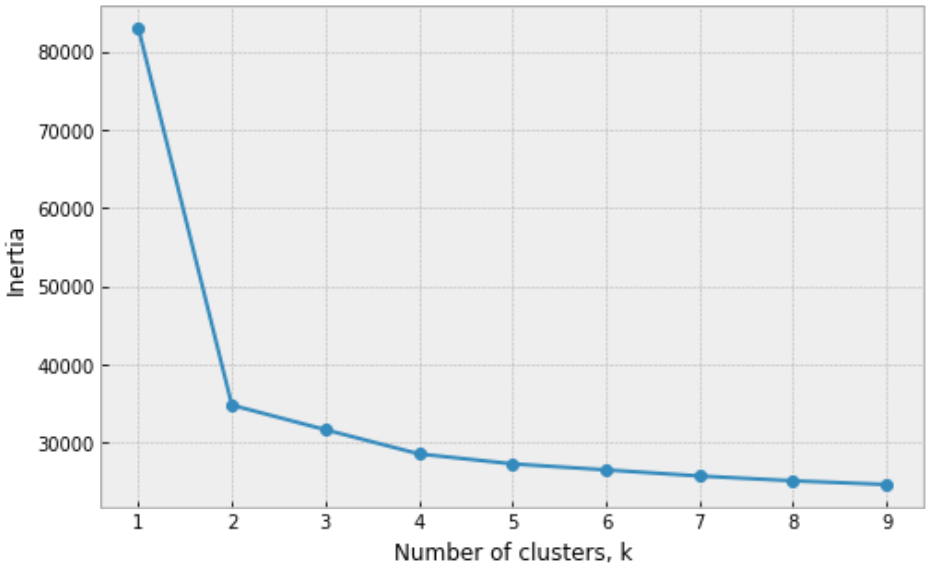
orig_label	LAYING	SITTING	STANDING	WALKING	WALKING_DOWNSTAIRS	WALKING_UPSTAIRS
clust_label						
0	559	437	407	0	0	0
1	121	186	261	0	0	0

orig_label	LAYING	SITTING	STANDING	WALKING	WALKING_DOWNSTAIRS	WALKING_UPSTAIRS
2	1	0	0	242	76	487
3	0	0	0	0	40	0
clust_label						
4	0	0	0	225	243	36
5	0	0	0	136	134	18

TEST DATA

In [30]:

```
ks = range(1, 10)
inertias = []

for k in ks:
    model = KMeans(n_clusters=k)
    model.fit(test)
    inertias.append(model.inertia_)

plt.figure(figsize=(8,5))
plt.style.use('bmh')
plt.plot(ks, inertias, '-o')
plt.xlabel('Number of clusters, k')
plt.ylabel('Inertia')
plt.xticks(ks)
plt.show()
```



2- CLUSTERS

In [31]:

```
k_means = KMeans(n_clusters = 2, random_state=123, n_init=30)
k_means.fit(test)
c_labels = k_means.labels_
y_clust = k_means.predict(test)
print(y_clust)
test['cluster_k_means']=y_clust
print(test)
```

```
[1 1 1 ... 0 0 0]
      tBodyAcc.mean.X  tBodyAcc.mean.Y  tBodyAcc.mean.Z  tBodyAcc.std.X  \
0               0.280         -0.01950          -0.1130         -0.9950
1               0.277         -0.01660          -0.1150         -0.9980
2               0.277         -0.02180          -0.1210         -0.9970
3               0.279         -0.01480          -0.1170         -0.9970
4               0.279         -0.01450          -0.1070         -0.9980
...               ...              ...              ...             ...
1536            0.289         -0.02810          -0.0943         -0.0623
1537            0.377         -0.01810          -0.1100         -0.3140
1538            0.253         -0.02490          -0.1700         -0.3080
```

```
1539           0.277          0.00108          -0.0740         -0.0685
1540           0.192         -0.03360          -0.1060         -0.3550

      tBodyAcc.std.Y  tBodyAcc.std.Z  tBodyAcc.mad.X  tBodyAcc.mad.Y  \
0            -0.9670          -0.979          -0.997         -0.9640
1            -0.9810          -0.990          -0.998         -0.9800
2            -0.9610          -0.984          -0.998         -0.9570
3            -0.9820          -0.983          -0.997         -0.9820
4            -0.9860          -0.993          -0.998         -0.9850
...              ...             ...             ...             ...
1536          0.1140          -0.190          -0.114          0.0393
1537         -0.1520          -0.214          -0.394         -0.1810
1538         -0.1890          -0.141          -0.377         -0.2260
1539         -0.2450          -0.145          -0.149         -0.3030
1540         -0.0925          -0.313          -0.434         -0.0887

      tBodyAcc.mad.Z  tBodyAcc.max.X  ...  fBodyBodyGyroJerkMag.skewness  \
0             -0.977         -0.9390  ...                         -0.391
1             -0.990         -0.9420  ...                         -0.351
2             -0.984         -0.9410  ...                         -0.269
3             -0.981         -0.9420  ...                         -0.779
4             -0.995         -0.9430  ...                         -0.715
...              ...             ...  ...                            ...
1536          -0.207          0.3300  ...                         -0.237
1537          -0.266         -0.0726  ...                         -0.323
1538          -0.221          0.0920  ...                         -0.142
1539          -0.199          0.4030  ...                          0.161
1540          -0.336         -0.0416  ...                         -0.630

      fBodyBodyGyroJerkMag.kurtosis  angle.tBodyAccMean.gravity  \
0                           -0.760                    -0.11900
1                           -0.699                     0.12300
2                           -0.573                     0.01300
3                           -0.940                    -0.00145
4                           -0.937                     0.02570
...                            ...                         ...
1536                        -0.607                    -0.19600
1537                        -0.753                    -0.82900
1538                        -0.564                     0.00451
1539                        -0.126                     0.13400
1540                        -0.916                     0.53600

      angle.tBodyAccJerkMean.gravityMean  angle.tBodyGyroMean.gravityMean  \
0                                 0.1780                            0.101
1                                 0.1230                            0.694
2                                 0.0809                           -0.234
3                                -0.0481                           -0.340
4                                 0.0665                           -0.226
...                                  ...                              ...
1536                              0.6980                            0.990
1537                              0.0483                            0.913
1538                              0.3570                           -0.946
1539                              0.8830                           -0.994
1540                              0.6890                           -0.937

      angle.tBodyGyroJerkMean.gravityMean  angle.X.gravityMean  \
0                                   0.809              -0.849
1                                  -0.616              -0.848
2                                   0.118              -0.848
3                                  -0.229              -0.759
4                                  -0.225              -0.762
...                                   ...                 ...
1536                               -0.108              -0.806
1537                               -0.904              -0.695
1538                                0.614              -0.695
1539                                0.475              -0.804
1540                                0.562              -0.647

      angle.Y.gravityMean  angle.Z.gravityMean  cluster_k_means
0                   0.181              -0.0491                1
1                   0.185              -0.0439                1
2                   0.189              -0.0374                1
```

```
3                  0.264                0.0270                 1
4                  0.262                0.0294                 1
...                  ...                   ...               ...
1536               0.190                0.1200                 0
1537               0.246                0.1730                 0
1538               0.259                0.1580                 0
1539               0.197                0.1140                 0
1540               0.282                0.1810                 0

[1541 rows x 562 columns]
```

In [32]:

```
k_means.cluster_centers_
```

Out[32]:

```
array([[ 0.27332564, -0.01963102, -0.11027506, ..., -0.72086182,
         0.23169051,  0.07847158],
       [ 0.27790691, -0.01672017, -0.10833092, ..., -0.30261144,
        -0.08169381, -0.16577687]])
```

In [33]:

```
from sklearn.cluster import AgglomerativeClustering
hc1 = AgglomerativeClustering(n_clusters=6, affinity = 'euclidean', linkage = 'ward')
Y_hc = hc1.fit_predict(test)
print(Y_hc)
test['cluster_h1']=Y_hc
print(test)
```

```
[0 4 0 ... 3 5 3]
      tBodyAcc.mean.X  tBodyAcc.mean.Y  tBodyAcc.mean.Z  tBodyAcc.std.X  \
0               0.280         -0.01950          -0.1130         -0.9950
1               0.277         -0.01660          -0.1150         -0.9980
2               0.277         -0.02180          -0.1210         -0.9970
3               0.279         -0.01480          -0.1170         -0.9970
4               0.279         -0.01450          -0.1070         -0.9980
...               ...              ...              ...             ...
1536            0.289         -0.02810          -0.0943         -0.0623
1537            0.377         -0.01810          -0.1100         -0.3140
1538            0.253         -0.02490          -0.1700         -0.3080
1539            0.277          0.00108          -0.0740         -0.0685
1540            0.192         -0.03360          -0.1060         -0.3550

      tBodyAcc.std.Y  tBodyAcc.std.Z  tBodyAcc.mad.X  tBodyAcc.mad.Y  \
0            -0.9670          -0.979          -0.997         -0.9640
1            -0.9810          -0.990          -0.998         -0.9800
2            -0.9610          -0.984          -0.998         -0.9570
3            -0.9820          -0.983          -0.997         -0.9820
4            -0.9860          -0.993          -0.998         -0.9850
...              ...             ...             ...             ...
1536          0.1140          -0.190          -0.114          0.0393
1537         -0.1520          -0.214          -0.394         -0.1810
1538         -0.1890          -0.141          -0.377         -0.2260
1539         -0.2450          -0.145          -0.149         -0.3030
1540         -0.0925          -0.313          -0.434         -0.0887

      tBodyAcc.mad.Z  tBodyAcc.max.X  ...  fBodyBodyGyroJerkMag.kurtosis  \
0            -0.977         -0.9390  ...                         -0.760
1            -0.990         -0.9420  ...                         -0.699
2            -0.984         -0.9410  ...                         -0.573
3            -0.981         -0.9420  ...                         -0.940
4            -0.995         -0.9430  ...                         -0.937
...             ...             ...  ...                            ...
1536         -0.207          0.3300  ...                         -0.607
1537         -0.266         -0.0726  ...                         -0.753
1538         -0.221          0.0920  ...                         -0.564
1539         -0.199          0.4030  ...                         -0.126
1540         -0.336         -0.0416  ...                         -0.916

      angle.tBodyAccMean.gravity  angle.tBodyAccJerkMean.gravityMean  \
```

```
0                      -0.11900                                0.1780
1                       0.12300                                0.1230
2                       0.01300                                0.0809
3                      -0.00145                               -0.0481
4                       0.02570                                0.0665
...                         ...                                   ...
1536                   -0.19600                                0.6980
1537                   -0.82900                                0.0483
1538                    0.00451                                0.3570
1539                    0.13400                                0.8830
1540                    0.53600                                0.6890

      angle.tBodyGyroMean.gravityMean  angle.tBodyGyroJerkMean.gravityMean  \
0                               0.101                                0.809
1                               0.694                               -0.616
2                              -0.234                                0.118
3                              -0.340                               -0.229
4                              -0.226                               -0.225
...                               ...                                  ...
1536                            0.990                               -0.108
1537                            0.913                               -0.904
1538                           -0.946                                0.614
1539                           -0.994                                0.475
1540                           -0.937                                0.562

      angle.X.gravityMean  angle.Y.gravityMean  angle.Z.gravityMean  \
0                  -0.849                0.181              -0.0491
1                  -0.848                0.185              -0.0439
2                  -0.848                0.189              -0.0374
3                  -0.759                0.264               0.0270
4                  -0.762                0.262               0.0294
...                   ...                  ...                  ...
1536               -0.806                0.190               0.1200
1537               -0.695                0.246               0.1730
1538               -0.695                0.259               0.1580
1539               -0.804                0.197               0.1140
1540               -0.647                0.282               0.1810

      cluster_k_means  cluster_h1
0                   1           0
1                   1           4
2                   1           0
3                   1           4
4                   1           4
...               ...         ...
1536                0           5
1537                0           3
1538                0           3
1539                0           5
1540                0           3

[1541 rows x 563 columns]
```

In [34]:

```
test['cluster_h1'].value_counts()
```

Out[34]:

```
0    365
3    352
2    287
4    186
5    181
1    170
Name: cluster_h1, dtype: int64
```

6-CLUSTERS

In [35]:

```
k_means = KMeans(n_clusters = 6, random_state=123, n_init=30)
```

```
k_means.fit(test)
c_labels = k_means.labels_
y_clust = k_means.predict(test)
print(y_clust)
test['cluster_k_means_6']=y_clust
print(test)
```

```
[5 1 5 ... 0 2 0]
      tBodyAcc.mean.X  tBodyAcc.mean.Y  tBodyAcc.mean.Z  tBodyAcc.std.X  \
0               0.280         -0.01950          -0.1130         -0.9950
1               0.277         -0.01660          -0.1150         -0.9980
2               0.277         -0.02180          -0.1210         -0.9970
3               0.279         -0.01480          -0.1170         -0.9970
4               0.279         -0.01450          -0.1070         -0.9980
...               ...              ...              ...             ...
1536            0.289         -0.02810          -0.0943         -0.0623
1537            0.377         -0.01810          -0.1100         -0.3140
1538            0.253         -0.02490          -0.1700         -0.3080
1539            0.277          0.00108          -0.0740         -0.0685
1540            0.192         -0.03360          -0.1060         -0.3550

      tBodyAcc.std.Y  tBodyAcc.std.Z  tBodyAcc.mad.X  tBodyAcc.mad.Y  \
0            -0.9670          -0.979          -0.997         -0.9640
1            -0.9810          -0.990          -0.998         -0.9800
2            -0.9610          -0.984          -0.998         -0.9570
3            -0.9820          -0.983          -0.997         -0.9820
4            -0.9860          -0.993          -0.998         -0.9850
...              ...             ...             ...             ...
1536          0.1140          -0.190          -0.114          0.0393
1537         -0.1520          -0.214          -0.394         -0.1810
1538         -0.1890          -0.141          -0.377         -0.2260
1539         -0.2450          -0.145          -0.149         -0.3030
1540         -0.0925          -0.313          -0.434         -0.0887

      tBodyAcc.mad.Z  tBodyAcc.max.X  ...  angle.tBodyAccMean.gravity  \
0            -0.977         -0.9390  ...                    -0.11900
1            -0.990         -0.9420  ...                     0.12300
2            -0.984         -0.9410  ...                     0.01300
3            -0.981         -0.9420  ...                    -0.00145
4            -0.995         -0.9430  ...                     0.02570
...             ...             ...  ...                         ...
1536         -0.207          0.3300  ...                    -0.19600
1537         -0.266         -0.0726  ...                    -0.82900
1538         -0.221          0.0920  ...                     0.00451
1539         -0.199          0.4030  ...                     0.13400
1540         -0.336         -0.0416  ...                     0.53600

      angle.tBodyAccJerkMean.gravityMean  angle.tBodyGyroMean.gravityMean  \
0                                 0.1780                            0.101
1                                 0.1230                            0.694
2                                 0.0809                           -0.234
3                                -0.0481                           -0.340
4                                 0.0665                           -0.226
...                                  ...                              ...
1536                              0.6980                            0.990
1537                              0.0483                            0.913
1538                              0.3570                           -0.946
1539                              0.8830                           -0.994
1540                              0.6890                           -0.937

      angle.tBodyGyroJerkMean.gravityMean  angle.X.gravityMean  \
0                                   0.809              -0.849
1                                  -0.616              -0.848
2                                   0.118              -0.848
3                                  -0.229              -0.759
4                                  -0.225              -0.762
...                                   ...                 ...
1536                               -0.108              -0.806
1537                               -0.904              -0.695
1538                                0.614              -0.695
1539                                0.475              -0.804
1540                                0.562              -0.647
```

```
      angle.Y.gravityMean  angle.Z.gravityMean  cluster_k_means  cluster_h1  \
0                   0.181              -0.0491                1           0
1                   0.185              -0.0439                1           4
2                   0.189              -0.0374                1           0
3                   0.264               0.0270                1           4
4                   0.262               0.0294                1           4
...                   ...                  ...              ...         ...
1536                0.190               0.1200                0           5
1537                0.246               0.1730                0           3
1538                0.259               0.1580                0           3
1539                0.197               0.1140                0           5
1540                0.282               0.1810                0           3

      cluster_k_means_6
0                     5
1                     1
2                     5
3                     1
4                     1
...                 ...
1536                  2
1537                  0
1538                  0
1539                  2
1540                  0

[1541 rows x 564 columns]
```

In [36]:

```
from sklearn.cluster import AgglomerativeClustering
#hc = AgglomerativeClustering(n_clusters=2, affinity = 'euclidean', linkage = 'ward')
Y_hc = hc.fit_predict(test)
print(Y_hc)
test['cluster_h_6']=y_clust
print(test)
```

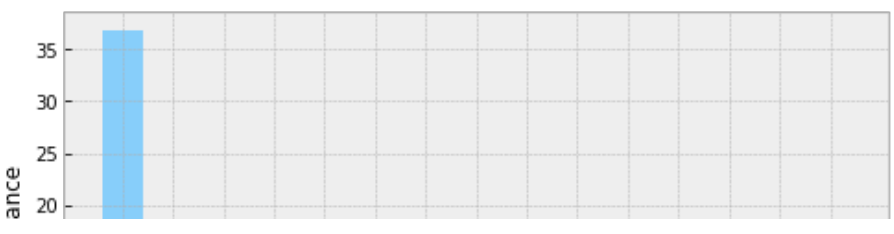
```
[2 4 2 ... 3 5 3]
      tBodyAcc.mean.X  tBodyAcc.mean.Y  tBodyAcc.mean.Z  tBodyAcc.std.X  \
0               0.280         -0.01950          -0.1130         -0.9950
1               0.277         -0.01660          -0.1150         -0.9980
2               0.277         -0.02180          -0.1210         -0.9970
3               0.279         -0.01480          -0.1170         -0.9970
4               0.279         -0.01450          -0.1070         -0.9980
...               ...              ...              ...             ...
1536            0.289         -0.02810          -0.0943         -0.0623
1537            0.377         -0.01810          -0.1100         -0.3140
1538            0.253         -0.02490          -0.1700         -0.3080
1539            0.277          0.00108          -0.0740         -0.0685
1540            0.192         -0.03360          -0.1060         -0.3550

      tBodyAcc.std.Y  tBodyAcc.std.Z  tBodyAcc.mad.X  tBodyAcc.mad.Y  \
0            -0.9670          -0.979          -0.997         -0.9640
1            -0.9810          -0.990          -0.998         -0.9800
2            -0.9610          -0.984          -0.998         -0.9570
3            -0.9820          -0.983          -0.997         -0.9820
4            -0.9860          -0.993          -0.998         -0.9850
...              ...             ...             ...             ...
1536          0.1140          -0.190          -0.114          0.0393
1537         -0.1520          -0.214          -0.394         -0.1810
1538         -0.1890          -0.141          -0.377         -0.2260
1539         -0.2450          -0.145          -0.149         -0.3030
1540         -0.0925          -0.313          -0.434         -0.0887

      tBodyAcc.mad.Z  tBodyAcc.max.X  ...  angle.tBodyAccJerkMean.gravityMean  \
0             -0.977         -0.9390  ...                              0.1780
1             -0.990         -0.9420  ...                              0.1230
2             -0.984         -0.9410  ...                              0.0809
3             -0.981         -0.9420  ...                             -0.0481
4             -0.995         -0.9430  ...                              0.0665
...              ...             ...  ...                                 ...
```

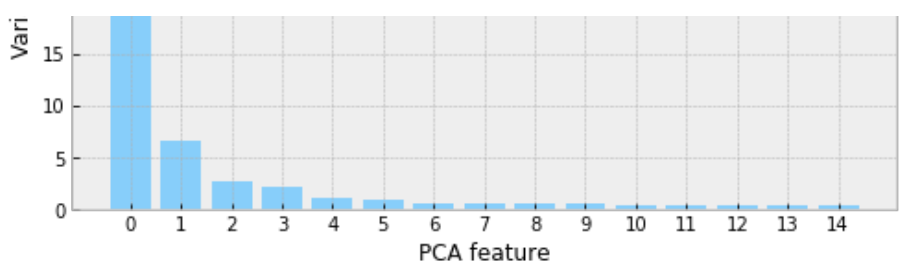
```
1536          -0.207          0.3300  ...                          0.6980
1537          -0.266         -0.0726  ...                          0.0483
1538          -0.221          0.0920  ...                          0.3570
1539          -0.199          0.4030  ...                          0.8830
1540          -0.336         -0.0416  ...                          0.6890

      angle.tBodyGyroMean.gravityMean  angle.tBodyGyroJerkMean.gravityMean  \
0                               0.101                                0.809
1                               0.694                               -0.616
2                              -0.234                                0.118
3                              -0.340                               -0.229
4                              -0.226                               -0.225
...                               ...                                  ...
1536                            0.990                               -0.108
1537                            0.913                               -0.904
1538                           -0.946                                0.614
1539                           -0.994                                0.475
1540                           -0.937                                0.562

      angle.X.gravityMean  angle.Y.gravityMean  angle.Z.gravityMean  \
0                  -0.849                0.181              -0.0491
1                  -0.848                0.185              -0.0439
2                  -0.848                0.189              -0.0374
3                  -0.759                0.264               0.0270
4                  -0.762                0.262               0.0294
...                   ...                  ...                  ...
1536               -0.806                0.190               0.1200
1537               -0.695                0.246               0.1730
1538               -0.695                0.259               0.1580
1539               -0.804                0.197               0.1140
1540               -0.647                0.282               0.1810

      cluster_k_means  cluster_h1  cluster_k_means_6  cluster_h_6
0                   1           0                  5            5
1                   1           4                  1            1
2                   1           0                  5            5
3                   1           4                  1            1
4                   1           4                  1            1
...               ...         ...                ...          ...
1536                0           5                  2            2
1537                0           3                  0            0
1538                0           3                  0            0
1539                0           5                  2            2
1540                0           3                  0            0

[1541 rows x 565 columns]
```

PCA-TEST

In [37]:

```
#check for optimal number of features
pca = PCA(random_state=123)
pca.fit(test)
features = range(pca.n_components_)

plt.figure(figsize=(8,4))
plt.bar(features[:15], pca.explained_variance_[:15], color='lightskyblue')
plt.xlabel('PCA feature')
plt.ylabel('Variance')
plt.xticks(features[:15])
plt.show()
```





In [38]:

```
def pca_transform(n_comp):
    pca = PCA(n_components=n_comp, random_state=123)
    global Data_reduced_test
    Data_reduced_test = pca.fit_transform(test)
    print('Shape of the new Data df: ' + str(Data_reduced_test.shape))
```

In [39]:

```
pca_transform(n_comp=1)
k_means.fit(Data_reduced_test)
c_labels = k_means.labels_
y_clust = k_means.predict(Data_reduced_test)
print(y_clust)
test['cluster_k_means_pca']=y_clust
print(test)
```

```
Shape of the new Data df: (1541, 1)
[3 0 3 ... 5 5 2]
      tBodyAcc.mean.X  tBodyAcc.mean.Y  tBodyAcc.mean.Z  tBodyAcc.std.X  \
0               0.280         -0.01950          -0.1130         -0.9950
1               0.277         -0.01660          -0.1150         -0.9980
2               0.277         -0.02180          -0.1210         -0.9970
3               0.279         -0.01480          -0.1170         -0.9970
4               0.279         -0.01450          -0.1070         -0.9980
...               ...              ...              ...             ...
1536            0.289         -0.02810          -0.0943         -0.0623
1537            0.377         -0.01810          -0.1100         -0.3140
1538            0.253         -0.02490          -0.1700         -0.3080
1539            0.277          0.00108          -0.0740         -0.0685
1540            0.192         -0.03360          -0.1060         -0.3550

      tBodyAcc.std.Y  tBodyAcc.std.Z  tBodyAcc.mad.X  tBodyAcc.mad.Y  \
0            -0.9670          -0.979          -0.997         -0.9640
1            -0.9810          -0.990          -0.998         -0.9800
2            -0.9610          -0.984          -0.998         -0.9570
3            -0.9820          -0.983          -0.997         -0.9820
4            -0.9860          -0.993          -0.998         -0.9850
...              ...             ...             ...             ...
1536          0.1140          -0.190          -0.114          0.0393
1537         -0.1520          -0.214          -0.394         -0.1810
1538         -0.1890          -0.141          -0.377         -0.2260
1539         -0.2450          -0.145          -0.149         -0.3030
1540         -0.0925          -0.313          -0.434         -0.0887

      tBodyAcc.mad.Z  tBodyAcc.max.X  ...  angle.tBodyGyroMean.gravityMean  \
0            -0.977         -0.9390  ...                            0.101
1            -0.990         -0.9420  ...                            0.694
2            -0.984         -0.9410  ...                           -0.234
3            -0.981         -0.9420  ...                           -0.340
4            -0.995         -0.9430  ...                           -0.226
...             ...             ...  ...                              ...
1536         -0.207          0.3300  ...                            0.990
1537         -0.266         -0.0726  ...                            0.913
1538         -0.221          0.0920  ...                           -0.946
1539         -0.199          0.4030  ...                           -0.994
1540         -0.336         -0.0416  ...                           -0.937

      angle.tBodyGyroJerkMean.gravityMean  angle.X.gravityMean  \
0                                   0.809               -0.849
1                                  -0.616               -0.848
```

```
2                                  0.118                -0.848
3                                 -0.229                -0.759
4                                 -0.225                -0.762
...                                  ...                   ...
1536                              -0.108                -0.806
1537                              -0.904                -0.695
1538                               0.614                -0.695
1539                               0.475                -0.804
1540                               0.562                -0.647

      angle.Y.gravityMean  angle.Z.gravityMean  cluster_k_means  cluster_h1  \
0                   0.181              -0.0491                1           0
1                   0.185              -0.0439                1           4
2                   0.189              -0.0374                1           0
3                   0.264               0.0270                1           4
4                   0.262               0.0294                1           4
...                   ...                  ...              ...         ...
1536                0.190               0.1200                0           5
1537                0.246               0.1730                0           3
1538                0.259               0.1580                0           3
1539                0.197               0.1140                0           5
1540                0.282               0.1810                0           3

      cluster_k_means_6  cluster_h_6  cluster_k_means_pca
0                     5            5                    3
1                     1            1                    0
2                     5            5                    3
3                     1            1                    0
4                     1            1                    0
...                 ...          ...                  ...
1536                  2            2                    1
1537                  0            0                    5
1538                  0            0                    5
1539                  2            2                    5
1540                  0            0                    2

[1541 rows x 566 columns]
```

In [40]:

```
Y_hc = hc.fit_predict(Data_reduced_test)
print(Y_hc)
test['cluster_h_6_pca']=y_clust
print(test)
```

```
[3 1 3 ... 0 0 4]
      tBodyAcc.mean.X  tBodyAcc.mean.Y  tBodyAcc.mean.Z  tBodyAcc.std.X  \
0               0.280         -0.01950          -0.1130         -0.9950
1               0.277         -0.01660          -0.1150         -0.9980
2               0.277         -0.02180          -0.1210         -0.9970
3               0.279         -0.01480          -0.1170         -0.9970
4               0.279         -0.01450          -0.1070         -0.9980
...               ...              ...              ...             ...
1536            0.289         -0.02810          -0.0943         -0.0623
1537            0.377         -0.01810          -0.1100         -0.3140
1538            0.253         -0.02490          -0.1700         -0.3080
1539            0.277          0.00108          -0.0740         -0.0685
1540            0.192         -0.03360          -0.1060         -0.3550

      tBodyAcc.std.Y  tBodyAcc.std.Z  tBodyAcc.mad.X  tBodyAcc.mad.Y  \
0            -0.9670          -0.979          -0.997         -0.9640
1            -0.9810          -0.990          -0.998         -0.9800
2            -0.9610          -0.984          -0.998         -0.9570
3            -0.9820          -0.983          -0.997         -0.9820
4            -0.9860          -0.993          -0.998         -0.9850
...              ...             ...             ...             ...
1536          0.1140          -0.190          -0.114          0.0393
1537         -0.1520          -0.214          -0.394         -0.1810
1538         -0.1890          -0.141          -0.377         -0.2260
1539         -0.2450          -0.145          -0.149         -0.3030
1540         -0.0925          -0.313          -0.434         -0.0887
```

```
      tBodyAcc.mad.Z  tBodyAcc.max.X  ...  \
0             -0.977         -0.9390  ...
1             -0.990         -0.9420  ...
2             -0.984         -0.9410  ...
3             -0.981         -0.9420  ...
4             -0.995         -0.9430  ...
...              ...             ...  ...
1536          -0.207          0.3300  ...
1537          -0.266         -0.0726  ...
1538          -0.221          0.0920  ...
1539          -0.199          0.4030  ...
1540          -0.336         -0.0416  ...

      angle.tBodyGyroJerkMean.gravityMean  angle.X.gravityMean  \
0                                   0.809               -0.849
1                                  -0.616               -0.848
2                                   0.118               -0.848
3                                  -0.229               -0.759
4                                  -0.225               -0.762
...                                   ...                  ...
1536                               -0.108               -0.806
1537                               -0.904               -0.695
1538                                0.614               -0.695
1539                                0.475               -0.804
1540                                0.562               -0.647

      angle.Y.gravityMean  angle.Z.gravityMean  cluster_k_means  cluster_h1  \
0                   0.181              -0.0491                1           0
1                   0.185              -0.0439                1           4
2                   0.189              -0.0374                1           0
3                   0.264               0.0270                1           4
4                   0.262               0.0294                1           4
...                   ...                  ...              ...         ...
1536                0.190               0.1200                0           5
1537                0.246               0.1730                0           3
1538                0.259               0.1580                0           3
1539                0.197               0.1140                0           5
1540                0.282               0.1810                0           3

      cluster_k_means_6  cluster_h_6  cluster_k_means_pca  cluster_h_6_pca
0                     5            5                    3                3
1                     1            1                    0                0
2                     5            5                    3                3
3                     1            1                    0                0
4                     1            1                    0                0
...                 ...          ...                  ...              ...
1536                  2            2                    1                1
1537                  0            0                    5                5
1538                  0            0                    5                5
1539                  2            2                    5                5
1540                  0            0                    2                2

[1541 rows x 567 columns]
```

In [44]:

```
from apyori import apriori
```

In [49]:

```
data = pd.read_csv('C:\\Users\\vedan\\Desktop\\ML ASSIGNMENT\\unsupervised\\my_movies.csv')
```

In [51]:

```
data
```

Out[51]:

	V1	V2	V3	V4	V5	Sixth Sense	Gladiator	LOTR1	Harry Potter1	Patriot	LOTR2	Harry Potter2	LOTR	Bravehea

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14
0	Sixth Sense	LOTR	Harry Potter1	Green Mile	LOTR1	Sixth Sense	Gladiator	LOTR1	Harry Potter1	Patriot	LOTR2	Harry Potter2	LOTR	Braveheart
1	Gladiator	Patriot	Braveheart	NaN	NaN	0	1	0	0	1	0	0	0	NaN
2	LOTR1	LOTR2	NaN	NaN	NaN	0	0	1	0	0	1	0	0	NaN
3	Gladiator	Patriot	Sixth Sense	NaN	NaN	1	1	0	0	1	0	0	0	NaN
4	Gladiator	Patriot	Sixth Sense	NaN	NaN	1	1	0	0	1	0	0	0	NaN
5	Gladiator	Patriot	Sixth Sense	NaN	NaN	1	1	0	0	1	0	0	0	NaN
6	Harry Potter1	Harry Potter2	NaN	NaN	NaN	0	0	0	1	0	0	1	0	NaN
7	Gladiator	Patriot	NaN	NaN	NaN	0	1	0	0	1	0	0	0	NaN
8	Gladiator	Patriot	Sixth Sense	NaN	NaN	1	1	0	0	1	0	0	0	NaN
9	Sixth Sense	LOTR	Gladiator	Green Mile	NaN	1	1	0	0	0	0	0	1	NaN

In [55]:

```
records = []
for i in range(len(data)):
    records.append([str(data.values[i,j]) for j in range(0, 15)])
```

In [56]:

```
association_rules = apriori(records, min_support=0.0045, min_confidence=0.2, min_lift=3,
min_length=2)
association_results = list(association_rules)
```

In [59]:

```
print(len(association_results))
```

```
112
```

In [61]:

```
print(association_results[0])
```

```
RelationRecord(items=frozenset({'Green Mile', 'LOTR'}), support=0.1, ordered_statistics=[
OrderedStatistic(items_base=frozenset({'Green Mile'}), items_add=frozenset({'LOTR'}), con
fidence=0.5, lift=5.0), OrderedStatistic(items_base=frozenset({'LOTR'}), items_add=frozen
set({'Green Mile'}), confidence=1.0, lift=5.0)])
```

In [70]:

```
df= pd.DataFrame(columns=["Rule1","Rule2","Support","Confidence","Lift"])

for item in association_results:
    # first index of the inner list
    # Contains base item and add item
    pair = item[0]
    items = [x for x in pair]
    print("Rule: " + items[0] + " -> " + items[1])

    #second index of the inner list
    print("Support: " + str(item[1]))

    #third index of the list located at 0th
    #of the third index of the inner list

    print("Confidence: " + str(item[2][0][2]))
    print("Lift: " + str(item[2][0][3]))
```

```
    df = df.append({'Rule1' : items[0], 'Rule2' : items[1], 'Support' : item[1], 'Confid
ence' : item[2][0][2], 'Lift' : item[2][0][3]},  ignore_index = True)
    print("====================================")
```

```
Rule: Green Mile -> LOTR
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: Harry Potter1 -> Harry Potter2
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: LOTR1 -> LOTR2
Support: 0.2
Confidence: 1.0
Lift: 5.0
====================================
Rule: 0 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 0 -> Harry Potter1
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 0 -> LOTR2
Support: 0.2
Confidence: 1.0
Lift: 5.0
====================================
Rule: 1 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> Harry Potter1
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> LOTR1
Support: 0.2
Confidence: 1.0
Lift: 5.0
====================================
Rule: Gladiator -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: LOTR1 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: Green Mile -> Harry Potter1
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: Green Mile -> Harry Potter1
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: Green Mile -> LOTR
Support: 0.1
Confidence: 0.5
```

```
Lift: 5.0
====================================
Rule: Green Mile -> LOTR
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: LOTR1 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: LOTR1 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: Green Mile -> LOTR2
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: Harry Potter1 -> Harry Potter2
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: LOTR1 -> Harry Potter1
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: LOTR1 -> Harry Potter1
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: Harry Potter1 -> LOTR2
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: LOTR1 -> LOTR2
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: LOTR1 -> nan
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> 0
Support: 0.2
Confidence: 1.0
Lift: 5.0
====================================
Rule: 0 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
```

```
Rule: 0 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 0 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 0 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 0 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 0 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 0 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 0 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 0 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 0 -> Harry Potter1
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 0 -> Harry Potter1
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 0 -> Harry Potter1
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 0 -> Harry Potter1
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 0 -> LOTR2
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 0 -> nan
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 1 -> Gladiator
Support: 0.1
```

```
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> LOTR1
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> LOTR1
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> LOTR1
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> Harry Potter1
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> LOTR1
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> LOTR1
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> Harry Potter1
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> LOTR1
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> LOTR1
Support: 0.1
Confidence: 0.5
Lift: 5.0
```

```
=====================================
Rule: Gladiator -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: Gladiator -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: LOTR1 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: LOTR1 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: Green Mile -> Harry Potter1
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: nan -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: LOTR1 -> Green Mile
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: LOTR1 -> Harry Potter1
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 1 -> 0
```

```
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 0 -> Gladiator
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 0 -> Gladiator
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 0 -> LOTR2
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 0 -> LOTR1
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 0 -> LOTR2
Support: 0.1
Confidence: 0.5
Lift: 5.0
=====================================
Rule: 0 -> Green Mile
Support: 0.1
Confidence: 0.5
```

```
Lift: 5.0
====================================
Rule: 0 -> LOTR2
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 0 -> LOTR2
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> Gladiator
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> Gladiator
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> LOTR2
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> LOTR1
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> LOTR2
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> Sixth Sense
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> LOTR2
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> LOTR2
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> LOTR2
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: Green Mile -> Gladiator
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: LOTR2 -> LOTR1
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
```

```
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 0 -> nan
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 0 -> LOTR2
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> nan
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> LOTR2
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
Rule: 1 -> 0
Support: 0.1
Confidence: 0.5
Lift: 5.0
====================================
```

In [71]:

```
df
```

Out[71]:

	Rule1	Rule2	Support	Confidence	Lift
0	Green Mile	LOTR	0.1	0.5	5.0
1	Harry Potter1	Harry Potter2	0.1	0.5	5.0

	Rule1	Rule2	Support	Confidence	Lift
2	LOTR1	LOTR2	0.2	1.0	5.0
3	0	Green Mile	0.1	0.5	5.0
4	0	Harry Potter1	0.1	0.5	5.0
...	...	...	...	...	...
107	0	LOTR2	0.1	0.5	5.0
108	1	nan	0.1	0.5	5.0
109	1	LOTR2	0.1	0.5	5.0
110	1	0	0.1	0.5	5.0
111	1	0	0.1	0.5	5.0

112 rows × 5 columns

In [76]:

```
df1=df.sort_values(['Confidence', 'Lift'], ascending=False)
df1.head()
```

Out[76]:

	Rule1	Rule2	Support	Confidence	Lift
2	LOTR1	LOTR2	0.2	1.0	5.0
5	0	LOTR2	0.2	1.0	5.0
8	1	LOTR1	0.2	1.0	5.0
26	1	0	0.2	1.0	5.0
0	Green Mile	LOTR	0.1	0.5	5.0

In [ ]:

```

```