In [1]:
```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from pandas import Series
```

In [2]:
```python
df=pd.read_csv('mnist_train.csv')
```

In [3]:
```python
df.head()
```

Out[3]:

	label	pixel0	pixel1	pixel2	pixel3	pixel4	pixel5	pixel6	pixel7	pixel8	...	pixel774	pixel775	pixel776	pixel777	pixel778	pixel779
0	1	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0
2	1	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0
3	4	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0

5 rows × 785 columns

In [4]:
```python
l=df['label']
```

In [5]:
```python
d=df.drop('label',axis=1)
```

In [6]:
```python
d.head()
```

Out[6]:

	pixel0	pixel1	pixel2	pixel3	pixel4	pixel5	pixel6	pixel7	pixel8	pixel9	...	pixel774	pixel775	pixel776	pixel777	pixel778	pixel77
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	
1	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	
2	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	
3	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	
4	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	

5 rows × 784 columns

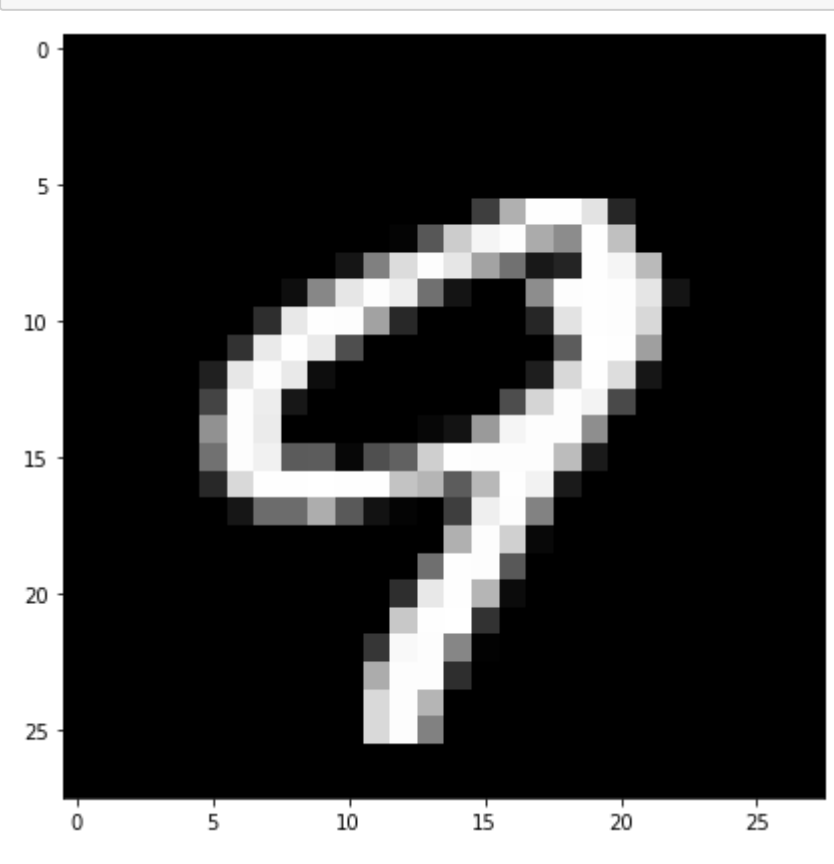
In [7]:
```python
d.shape
```

Out[7]:
```
(42000, 784)
```

In [8]:
```python
l.shape
```

Out[8]:
```
(42000,)
```

In [9]:
```python
plt.figure(figsize=(7,7))
idx=100

grid_plot=d.iloc[idx].to_numpy().reshape(28,28)
plt.imshow(grid_plot,interpolation=None,cmap='gray')
plt.show()


print(l[idx])
```



```
9
```

In [10]:
```python
labels=l.head(15000)
data=d.head(15000)
```

In [11]:
```python
print("Shape of data",data.shape)
print("Shape of labels",labels.shape)
```

```
Shape of data (15000, 784)
Shape of labels (15000,)
```

In [12]:
```python
from sklearn.preprocessing import StandardScaler
standardized_data = StandardScaler().fit_transform(data)
print(standardized_data.shape)
```

```
(15000, 784)
```

In [13]:
```python
sample_data = standardized_data

# matrix multiplication using numpy
covar_matrix = np.matmul(sample_data.T , sample_data)

print ( "The shape of variance matrix = ", covar_matrix.shape)
```

```
The shape of variance matrix =  (784, 784)
```

In [14]:
```python
from scipy.linalg import eigh

# the parameter 'eigvals' is defined (low value to heigh value)
# eigh function will return the eigen values in asending order
# this code generates only the top 2 (782 and 783) eigenvalues.
values, vectors = eigh(covar_matrix, eigvals=(782,783))

print("Shape of eigen vectors = ",vectors.shape)
# converting the eigen vectors into (2,d) shape for easyness of further computations
vectors = vectors.T

print("Updated shape of eigen vectors = ",vectors.shape)
```

```
Shape of eigen vectors =  (784, 2)
Updated shape of eigen vectors =  (2, 784)
```

In [15]:
```python
# projecting the original data sample on the plane
#formed by two principal eigen vectors by vector-vector multiplication.

import matplotlib.pyplot as plt
new_coordinates = np.matmul(vectors, sample_data.T)

print (" resultanat new data points' shape ", vectors.shape, "X", sample_data.T.shape," = ", new_coordi
nates.shape)
```

```
 resultanat new data points' shape  (2, 784) X (784, 15000)  =  (2, 15000)
```

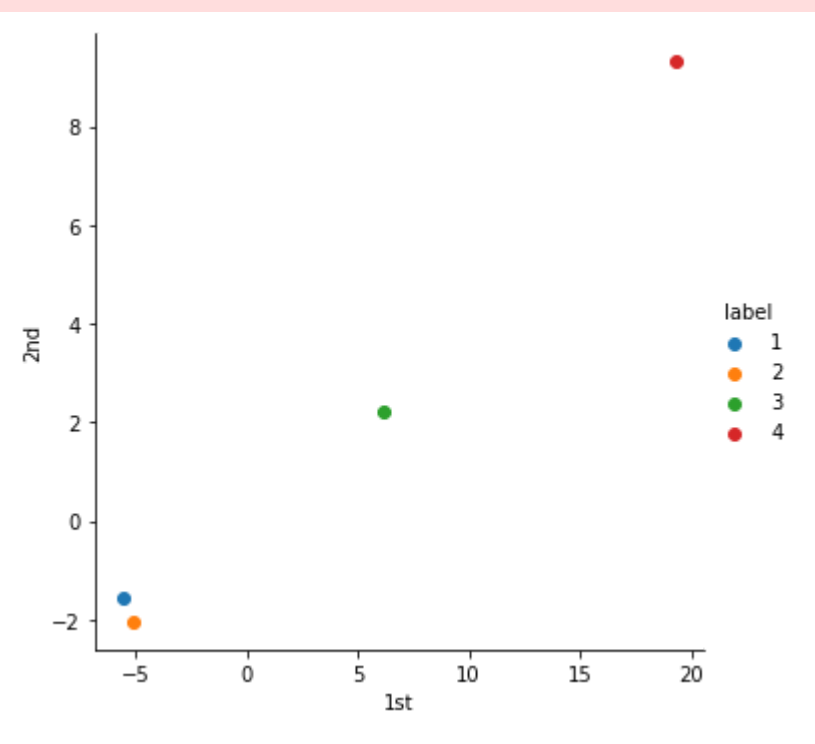
In [16]:
```python
import pandas as pd

# appending label to the 2d projected data
new_coordinates = np.vstack((new_coordinates, labels)).T

# creating a new data frame for ploting the labeled points.
dataframe = pd.DataFrame(data=new_coordinates, columns=("1st_principal", "2nd_principal", "label"))
print(dataframe.head())
```

```
   1st_principal  2nd_principal  label
0      -5.558661      -5.043558    1.0
1       6.193635      19.305278    0.0
2      -1.909878      -7.678775    1.0
3       5.525748      -0.464845    4.0
4       6.366527      26.644289    0.0
```

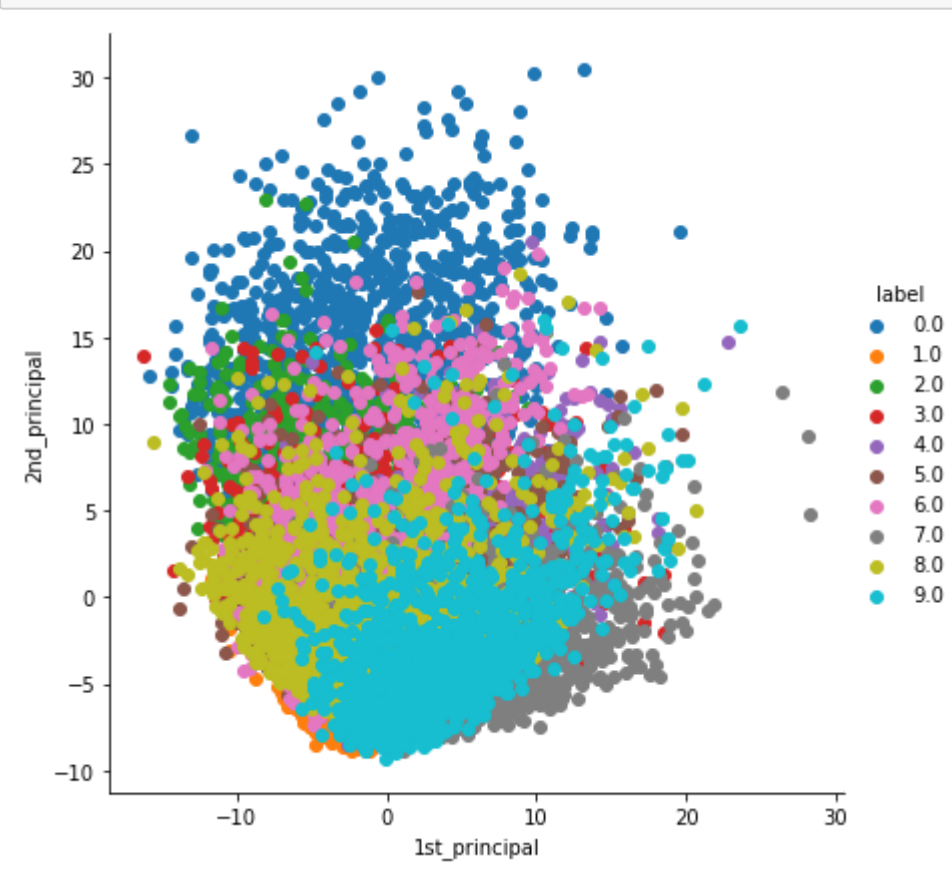
In [17]:
```python
import pandas as pd
df=pd.DataFrame()
df['1st']=[-5.558661,-5.043558,6.193635 ,19.305278]
df['2nd']=[-1.558661,-2.043558,2.193635 ,9.305278]
df['label']=[1,2,3,4]
```

In [18]:
```python
import seaborn as sn
import matplotlib.pyplot as plt
sn.FacetGrid(df, hue="label", size=5).map(plt.scatter, '1st', '2nd').add_legend()
plt.show()
```

```
C:\Users\Bhoomi\anaconda3\lib\site-packages\seaborn\axisgrid.py:243: UserWarning: The `size` paramete
r has been renamed to `height`; please update your code.
  warnings.warn(msg, UserWarning)
```

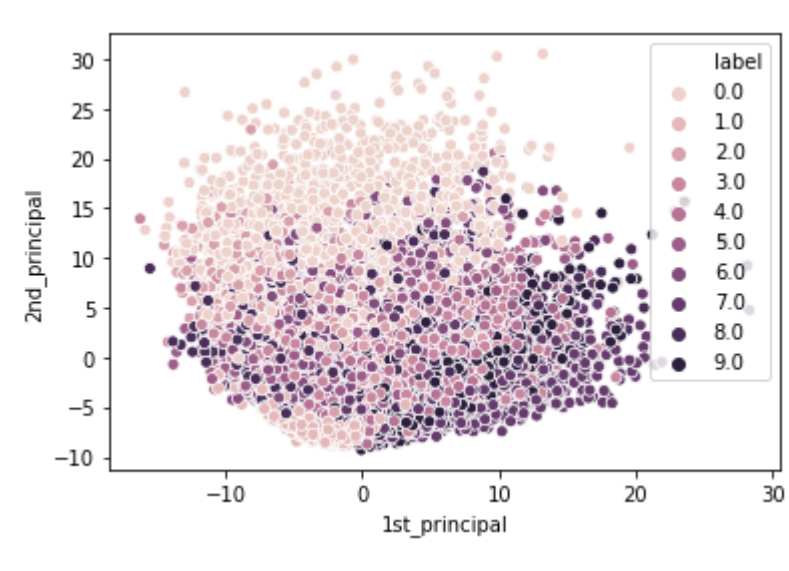


In [19]:
```python
# ploting the 2d data points with seaborn
import seaborn as sn
sn.FacetGrid(dataframe, hue="label", size=6).map(plt.scatter, '1st_principal', '2nd_principal').add_leg
end()
plt.show()
```



In [20]:
```python
sn.scatterplot(x="1st_principal",y="2nd_principal",legend="full",hue="label",data=dataframe)
```

Out[20]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1e425eadd30>
```



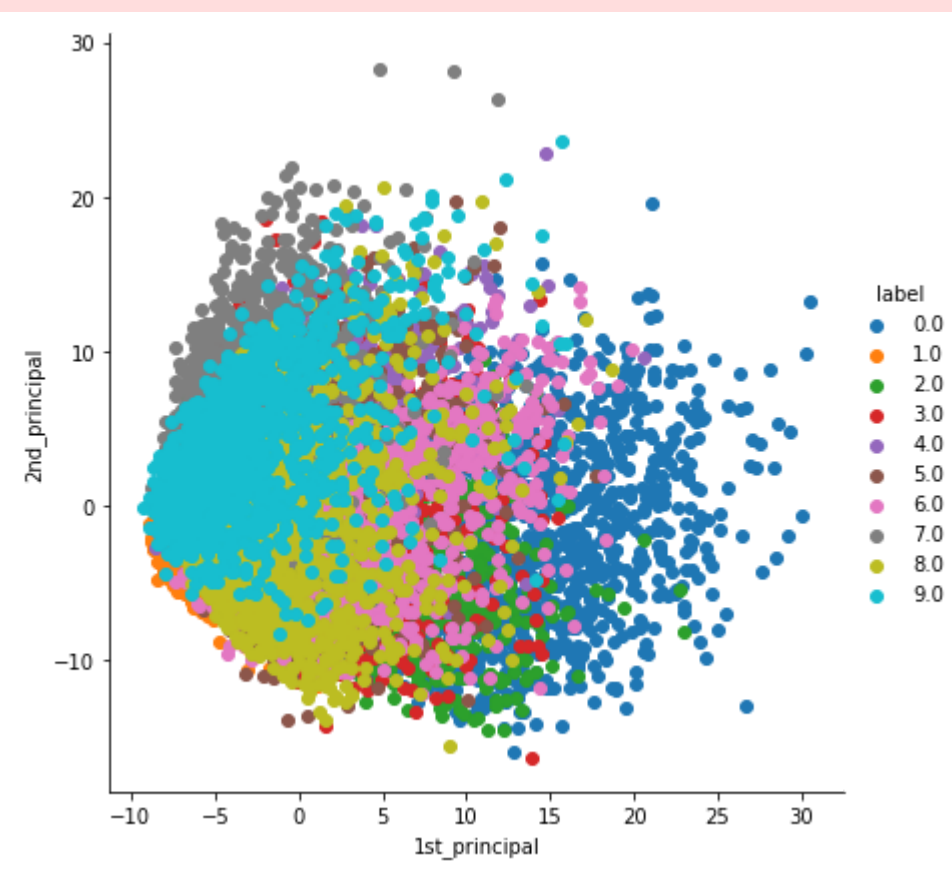
In [21]:
```python
# initializing the pca
from sklearn import decomposition
pca = decomposition.PCA()
```

In [22]:
```python
# configuring the parameteres
# the number of components = 2
pca.n_components = 2
pca_data = pca.fit_transform(sample_data)

# pca_reduced will contain the 2-d projects of simple data
print("shape of pca_reduced.shape = ", pca_data.shape)
```

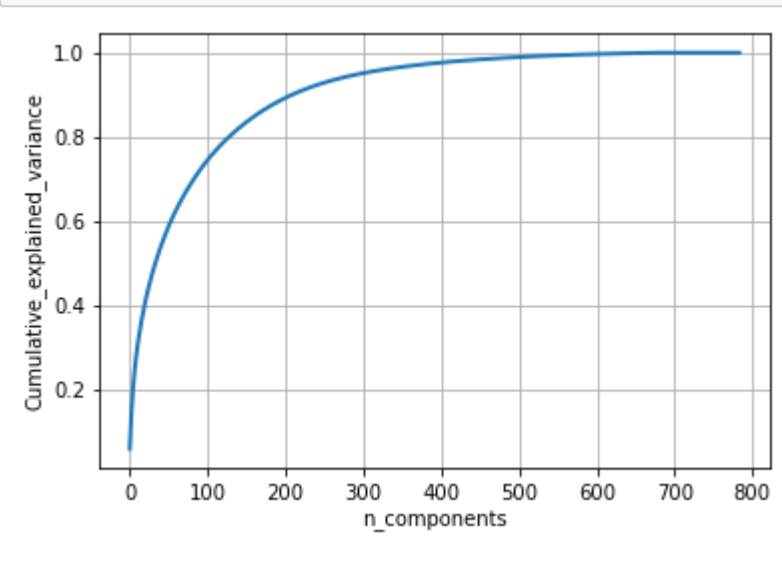
```
shape of pca_reduced.shape =  (15000, 2)
```

In [23]:
```python
# attaching the label for each 2-d data point
pca_data = np.vstack((pca_data.T, labels)).T

# creating a new data fram which help us in ploting the result data
pca_df = pd.DataFrame(data=pca_data, columns=("1st_principal", "2nd_principal", "label"))
sn.FacetGrid(pca_df, hue="label", size=6).map(plt.scatter, '1st_principal', '2nd_principal').add_legend
()
plt.show()
```

```
C:\Users\Bhoomi\anaconda3\lib\site-packages\seaborn\axisgrid.py:243: UserWarning: The `size` paramete
r has been renamed to `height`; please update your code.
  warnings.warn(msg, UserWarning)
```



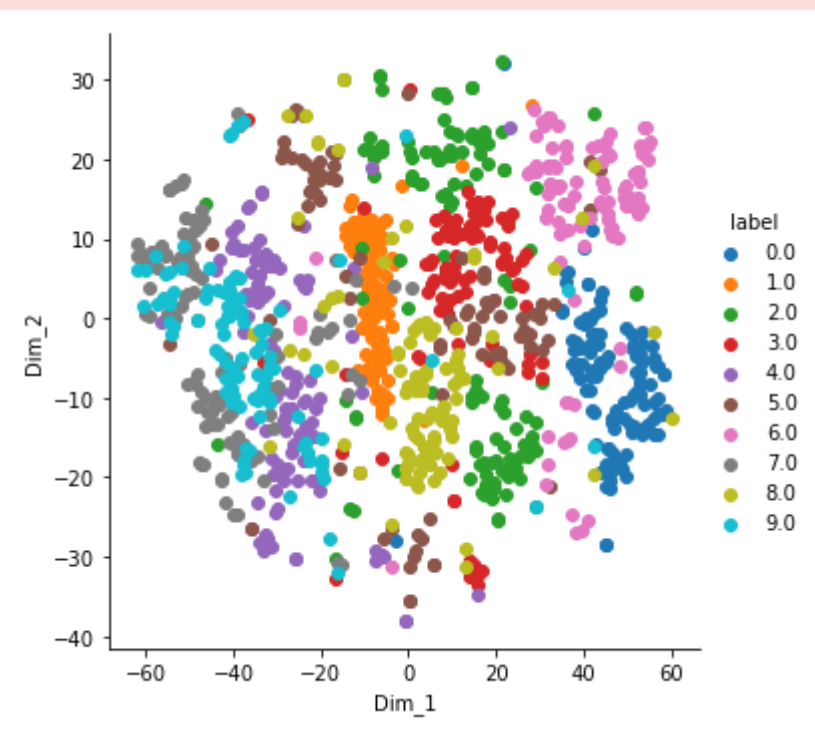
In [24]:
```python
# PCA for dimensionality redcution (non-visualization)

pca.n_components = 784
pca_data = pca.fit_transform(sample_data)

percentage_var_explained = pca.explained_variance_ / np.sum(pca.explained_variance_);

cum_var_explained = np.cumsum(percentage_var_explained)

# Plot the PCA spectrum
plt.figure(1, figsize=(6, 4))

plt.clf()
plt.plot(cum_var_explained, linewidth=2)
plt.axis('tight')
plt.grid()
plt.xlabel('n_components')
plt.ylabel('Cumulative_explained_variance')
plt.show()


# If we take 200-dimensions, approx. 90% of variance is expalined.
```



In [25]:
```python
from sklearn.manifold import TSNE

# Picking the top 1000 points as TSNE takes a lot of time for 15K points
data_1000 = standardized_data[0:1000,:]
labels_1000 = labels[0:1000]
```

In [27]:
```python
model = TSNE(n_components=2, random_state=0)
# the number of components = 2
# default perplexity = 30
# default Maximum number of iterations for the optimization = 1000

tsne_data = model.fit_transform(data_1000)


# creating a new data frame which help us in ploting the result data
tsne_data = np.vstack((tsne_data.T, labels_1000)).T
tsne_df = pd.DataFrame(data=tsne_data, columns=("Dim_1", "Dim_2", "label"))

# Ploting the result of tsne
sn.FacetGrid(tsne_df, hue="label", size=5).map(plt.scatter, 'Dim_1', 'Dim_2').add_legend()
plt.show()
```

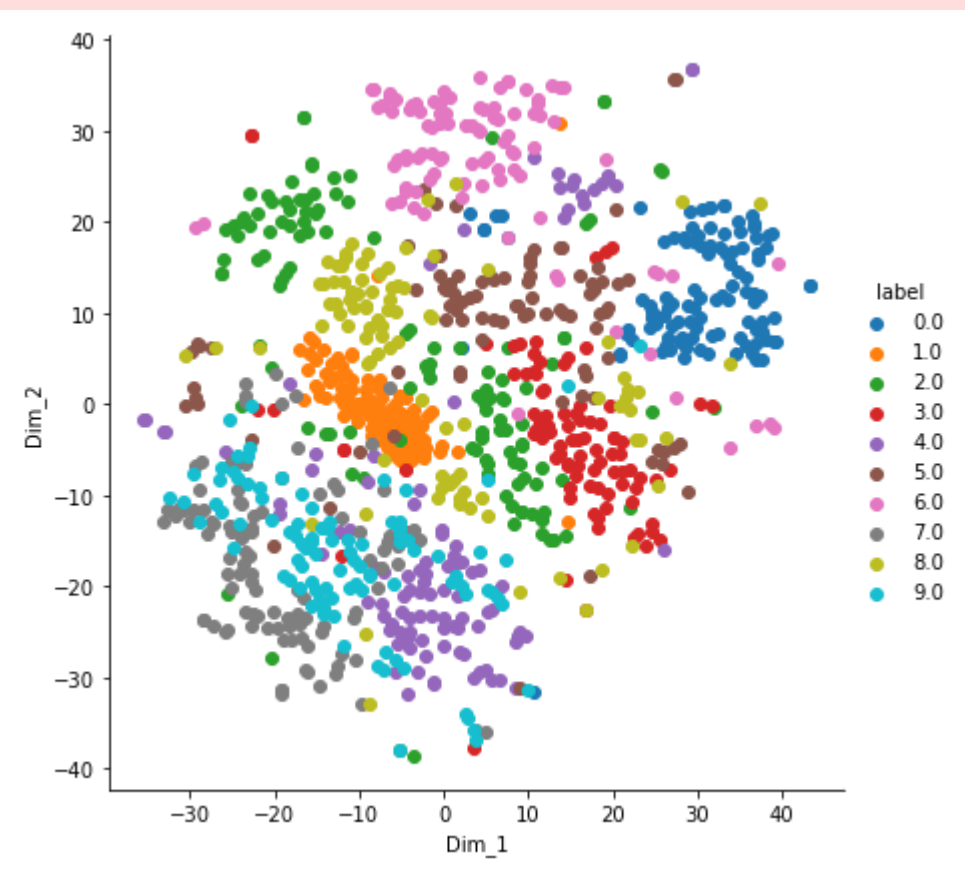
```
C:\Users\Bhoomi\anaconda3\lib\site-packages\seaborn\axisgrid.py:243: UserWarning: The `size` paramete
r has been renamed to `height`; please update your code.
  warnings.warn(msg, UserWarning)
```



Warning can be avoided using, import warnings warnings.filterwarnings("ignore")

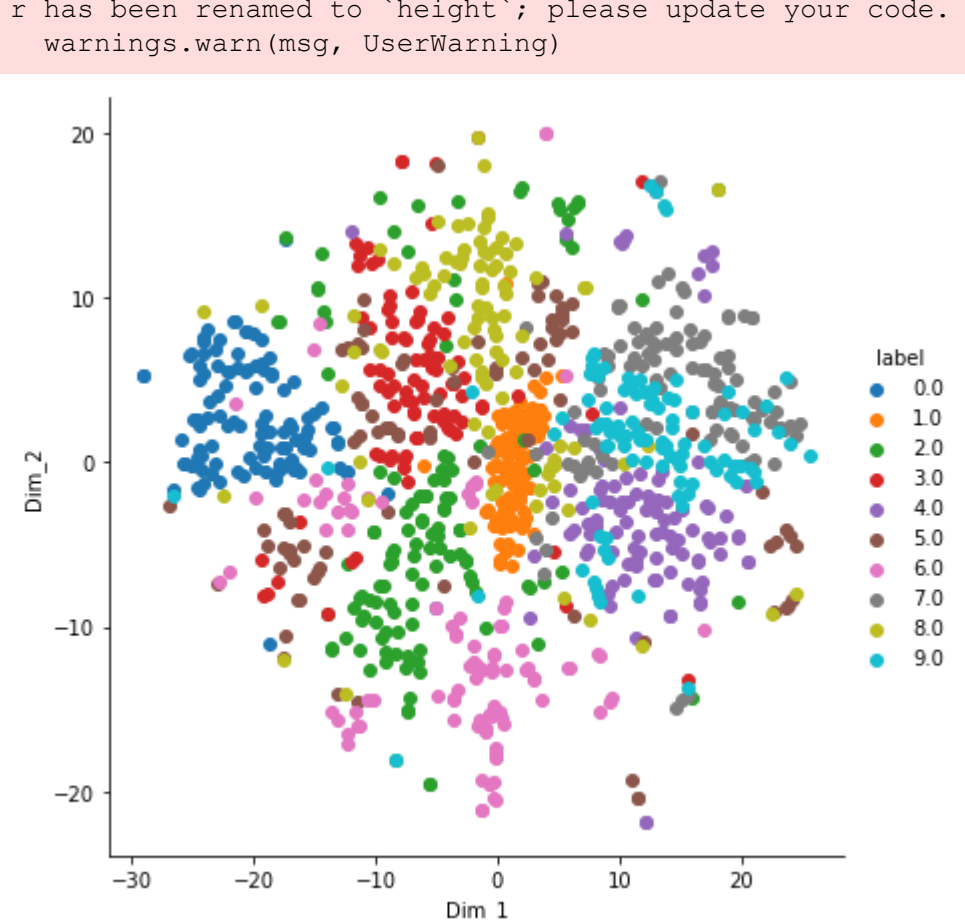
In [28]:
```python
model = TSNE(n_components=2, random_state=0,perplexity=50)
# configuring the parameteres
# the number of components = 2
# default perplexity = 30
# default learning rate = 200
# default Maximum number of iterations for the optimization = 1000

tsne_data = model.fit_transform(data_1000)


# creating a new data frame which help us in ploting the result data
tsne_data = np.vstack((tsne_data.T, labels_1000)).T
tsne_df = pd.DataFrame(data=tsne_data, columns=("Dim_1", "Dim_2", "label"))

# Ploting the result of tsne
sn.FacetGrid(tsne_df, hue="label", size=6).map(plt.scatter, 'Dim_1', 'Dim_2').add_legend()
plt.show()
```

```
C:\Users\Bhoomi\anaconda3\lib\site-packages\seaborn\axisgrid.py:243: UserWarning: The `size` paramete
r has been renamed to `height`; please update your code.
  warnings.warn(msg, UserWarning)
```



In [30]:
```python
model = TSNE(n_components=2, random_state=0,perplexity=100,n_iter=10000)
# configuring the parameteres
# the number of components = 2
# default perplexity = 30
# default learning rate = 200
# default Maximum number of iterations for the optimization = 1000

tsne_data = model.fit_transform(data_1000)


# creating a new data frame which help us in ploting the result data
tsne_data = np.vstack((tsne_data.T, labels_1000)).T
tsne_df = pd.DataFrame(data=tsne_data, columns=("Dim_1", "Dim_2", "label"))

# Ploting the result of tsne
sn.FacetGrid(tsne_df, hue="label", size=6).map(plt.scatter, 'Dim_1', 'Dim_2').add_legend()
plt.show()
```

```
C:\Users\Bhoomi\anaconda3\lib\site-packages\seaborn\axisgrid.py:243: UserWarning: The `size` paramete
r has been renamed to `height`; please update your code.
  warnings.warn(msg, UserWarning)
```



In [ ]: