t-SNE-MNIST-Digits-Dataset

November 8, 2018

1 t-SNE on MNIST Handwritten Digits Dataset

```
In [1]: # import required modules
        import numpy as np # for matrix multiplication
        import matplotlib.pyplot as plt # for plotting
        import seaborn as sns # for plotting
        import pandas as pd          # for data handling
        from sklearn.preprocessing import StandardScaler # for column standardization
        from sklearn.manifold import TSNE # for t-SNE
        import time # for time measurement
```

```
In [2]: # Read dataset
        d0 = pd.read_csv('./../../AAIC-Course/datasets/mnist-digits-dataset/train.csv')
        d0.head()
```

```
Out[2]:    label  pixel0  pixel1  pixel2  pixel3  pixel4  pixel5  pixel6  pixel7  \
        0      1       0       0       0       0       0       0       0       0
        1      0       0       0       0       0       0       0       0       0
        2      1       0       0       0       0       0       0       0       0
        3      4       0       0       0       0       0       0       0       0
        4      0       0       0       0       0       0       0       0       0

           pixel8   ...    pixel774  pixel775  pixel776  pixel777  pixel778  \
        0       0   ...           0         0         0         0         0
        1       0   ...           0         0         0         0         0
        2       0   ...           0         0         0         0         0
        3       0   ...           0         0         0         0         0
        4       0   ...           0         0         0         0         0

           pixel779  pixel780  pixel781  pixel782  pixel783
        0         0         0         0         0         0
        1         0         0         0         0         0
        2         0         0         0         0         0
        3         0         0         0         0         0
        4         0         0         0         0         0

        [5 rows x 785 columns]
```

```
In [3]: # split the dataset into features and labels
        d_labels_all = d0['label']
        d_data_all = d0.drop('label',axis=1)
        print(d_labels_all.shape,d_data_all.shape)
```

(42000,) (42000, 784)

1.1 Training Size - 15000

```
In [4]: # Taking 15000 images for training
        training_size = 15000

        d_train_labels = d_labels_all.head(training_size)
        d_train_data = d_data_all.head(training_size)
        print('Shape of Training Data {0}, Label {1}'.format(d_train_data.shape, d_train_labels.

        # standardize the data
        standardized_data = StandardScaler().fit_transform(d_train_data.astype(np.float64))
        standardized_data.shape
```

Shape of Training Data (15000, 784), Label (15000,)

Out[4]: (15000, 784)

1.1.1 t-SNE over 1000 data points

```
In [5]: no_of_data_points = 1000
```

```
In [6]: p_data = standardized_data[0:no_of_data_points,:]
        p_labels = d_train_labels[0:no_of_data_points]
        print('t-SNE Data Points {0} and its Labels {1}'.format(p_data.shape, p_labels.shape))
```

t-SNE Data Points (1000, 784) and its Labels (1000,)

Perplexity = default (30)

   Iterations = default (1000)

```
In [7]: time_start = time.time()

        # Compute t-SNE (784-D to 2-D) for visualization
        model = TSNE(n_components=2,random_state=0,perplexity=30,n_iter=1000)

        tsne_data = model.fit_transform(p_data)
        print('t-SNE done! Time elapsed: {} seconds'.format(time.time() - time_start))
```
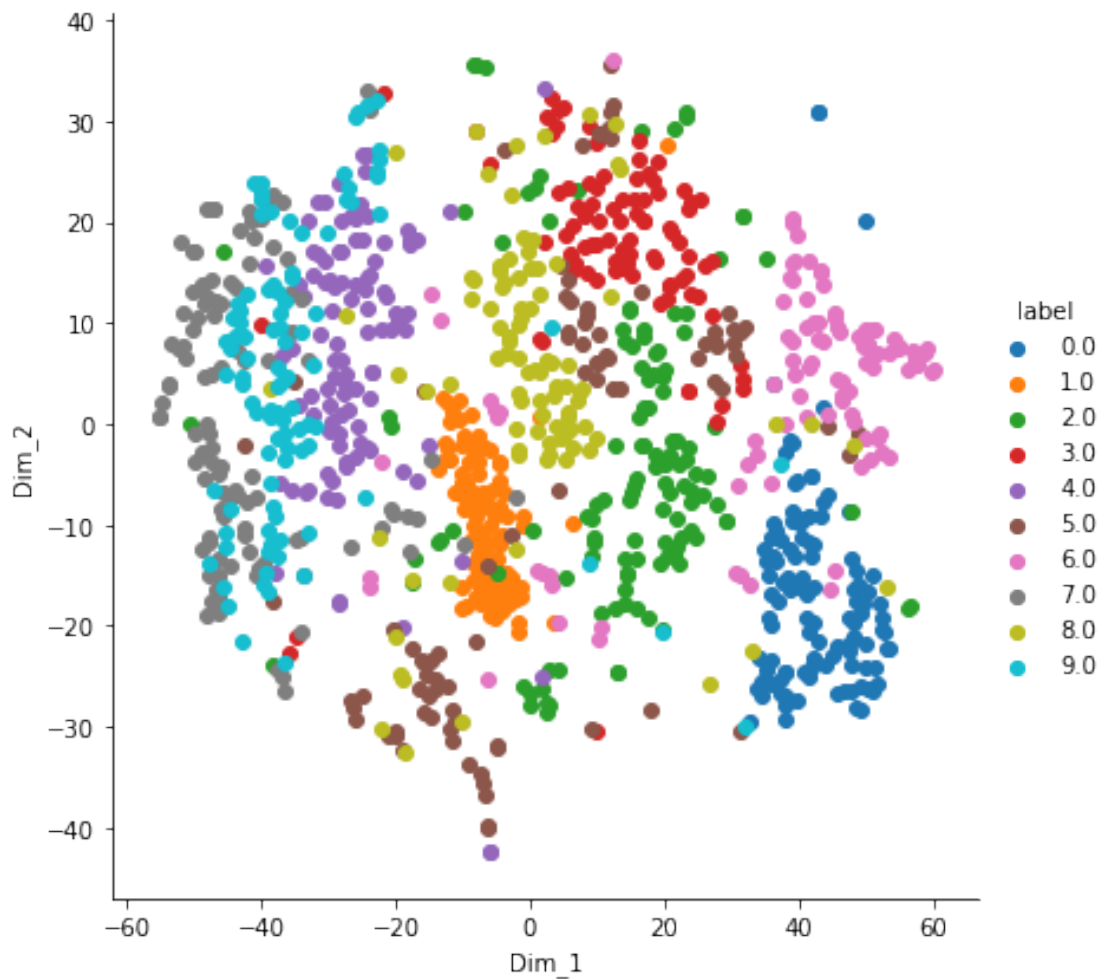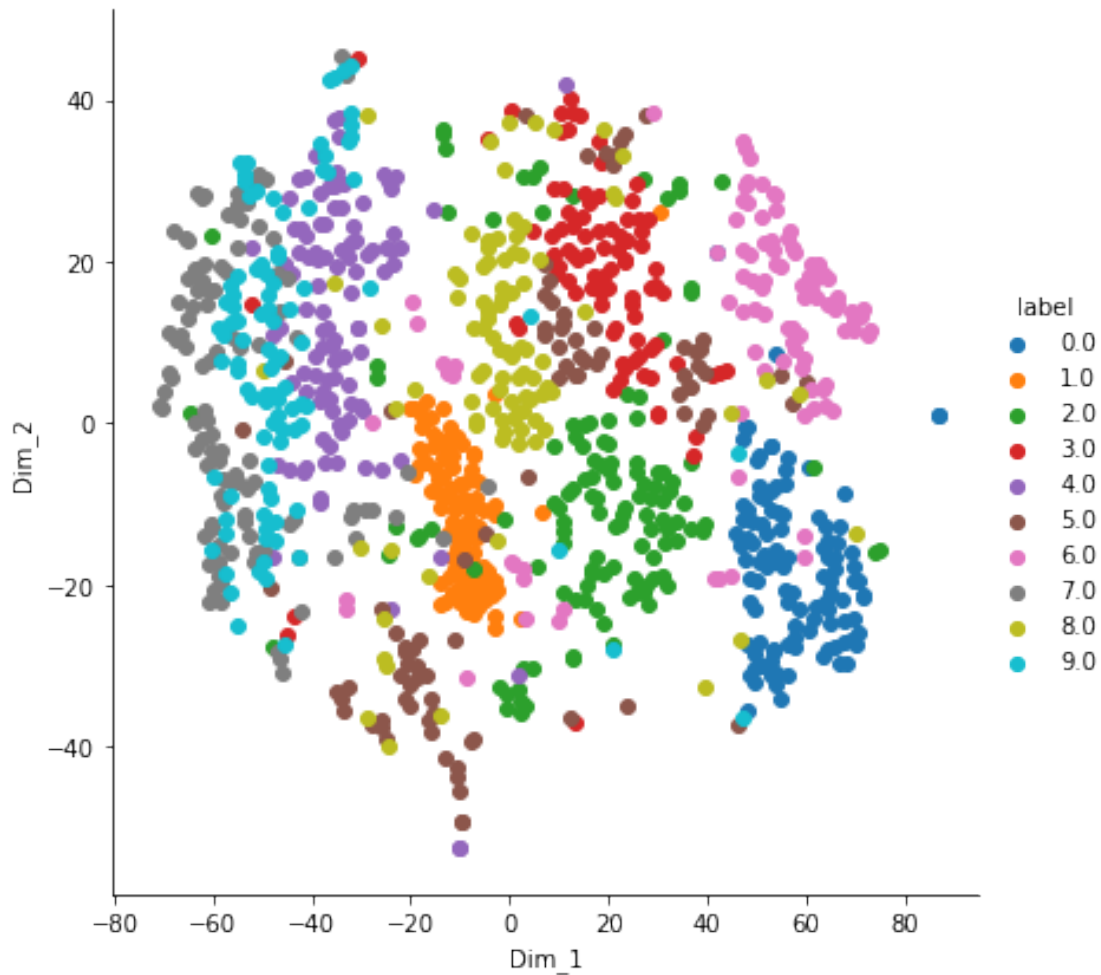
```
tsne_data = np.vstack((tsne_data.T,p_labels)).T
#print(tsne_data.shape)
tsne_data[:4]

tsne_df = pd.DataFrame(tsne_data,columns=['Dim_1','Dim_2','label'])
tsne_df.head()

sns.FacetGrid(tsne_df,hue='label',height=6).map(plt.scatter, 'Dim_1', 'Dim_2').add_legen
```

t-SNE done! Time elapsed: 12.665777444839478 seconds

Iterations = 5000

```
In [8]: time_start = time.time()

        # Compute t-SNE (784-D to 2-D) for visualization
```

3

```
model = TSNE(n_components=2,random_state=0,perplexity=30,n_iter=5000)

tsne_data = model.fit_transform(p_data)
print('t-SNE done! Time elapsed: {} seconds'.format(time.time() - time_start))

tsne_data = np.vstack((tsne_data.T,p_labels)).T
#print(tsne_data.shape)
tsne_data[:4]

tsne_df = pd.DataFrame(tsne_data,columns=['Dim_1','Dim_2','label'])
tsne_df.head()

sns.FacetGrid(tsne_df,hue='label',height=6).map(plt.scatter, 'Dim_1', 'Dim_2').add_legen
```
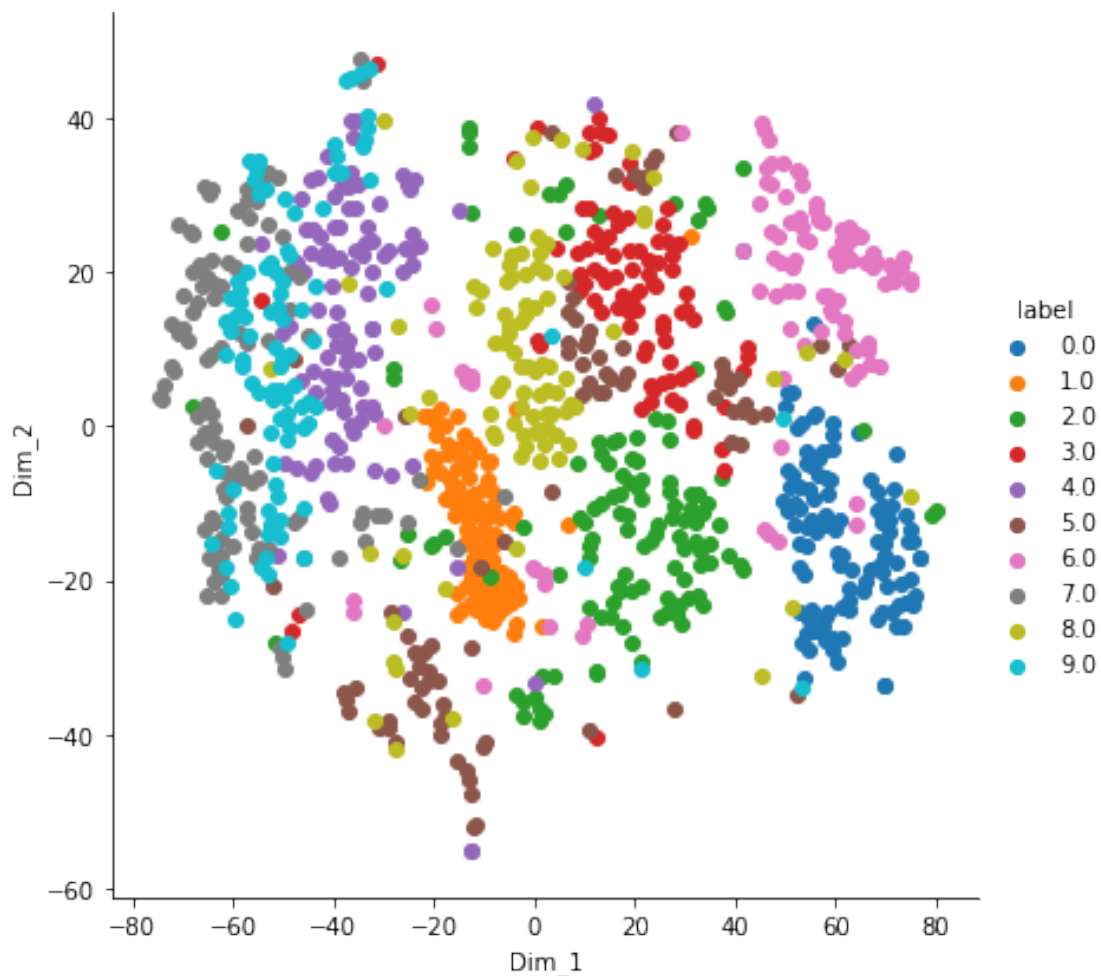
t-SNE done! Time elapsed: 48.392478704452515 seconds

Iterations = 10000

```
In [9]: time_start = time.time()

        # Compute t-SNE (784-D to 2-D) for visualization
        model = TSNE(n_components=2,random_state=0,perplexity=30,n_iter=10000)

        tsne_data = model.fit_transform(p_data)
        print('t-SNE done! Time elapsed: {} seconds'.format(time.time() - time_start))

        tsne_data = np.vstack((tsne_data.T,p_labels)).T
        #print(tsne_data.shape)
        tsne_data[:4]

        tsne_df = pd.DataFrame(tsne_data,columns=['Dim_1','Dim_2','label'])
        tsne_df.head()

        sns.FacetGrid(tsne_df,hue='label',height=6).map(plt.scatter, 'Dim_1', 'Dim_2').add_legen
```
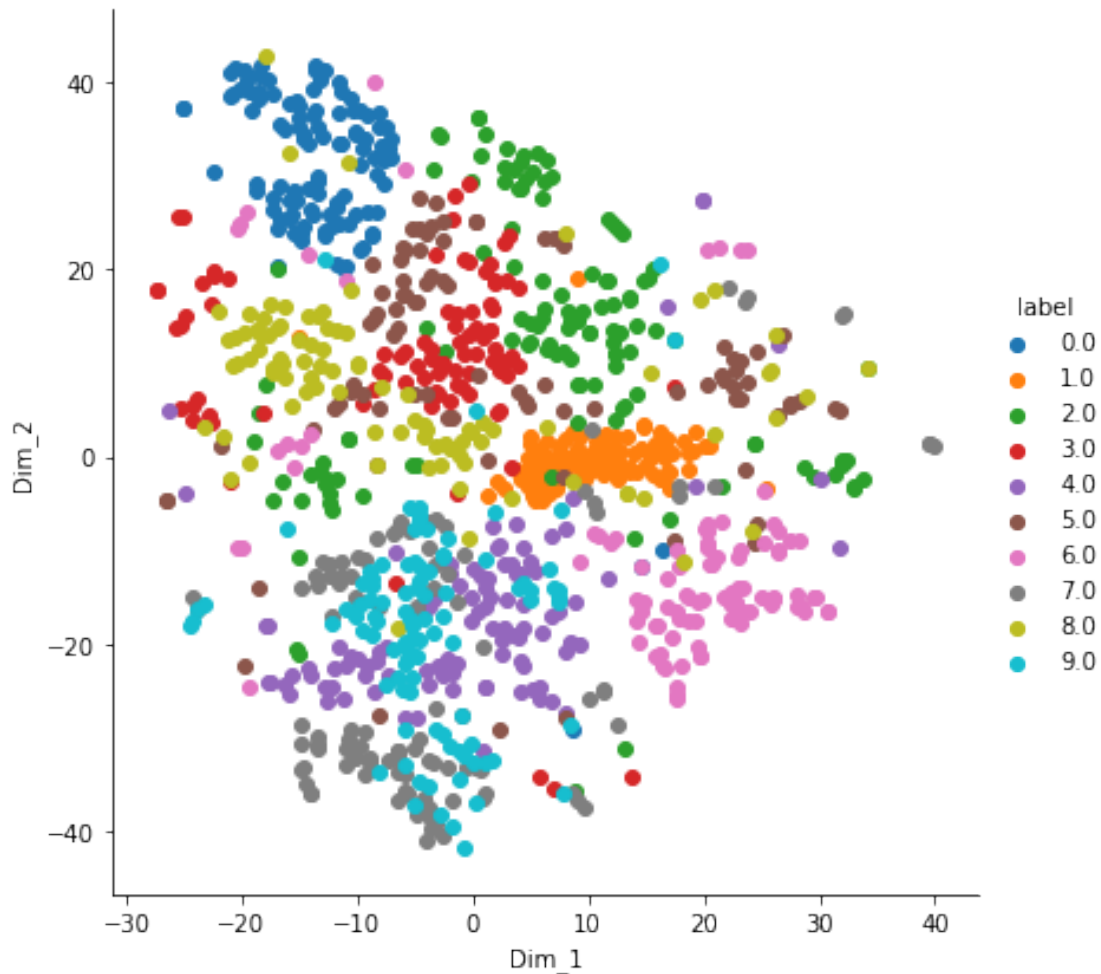
t-SNE done! Time elapsed: 87.9329731464386 seconds

Perplexity = 50

Iterations = 1000

```
In [10]: time_start = time.time()

         # Compute t-SNE (784-D to 2-D) for visualization
         model = TSNE(n_components=2,random_state=0,perplexity=50,n_iter=1000)

         tsne_data = model.fit_transform(p_data)
         print('t-SNE done! Time elapsed: {} seconds'.format(time.time() - time_start))

         tsne_data = np.vstack((tsne_data.T,p_labels)).T
         #print(tsne_data.shape)
         tsne_data[:4]

         tsne_df = pd.DataFrame(tsne_data,columns=['Dim_1','Dim_2','label'])
         tsne_df.head()

         sns.FacetGrid(tsne_df,hue='label',height=6).map(plt.scatter, 'Dim_1', 'Dim_2').add_lege

t-SNE done! Time elapsed: 14.85374641418457 seconds
```
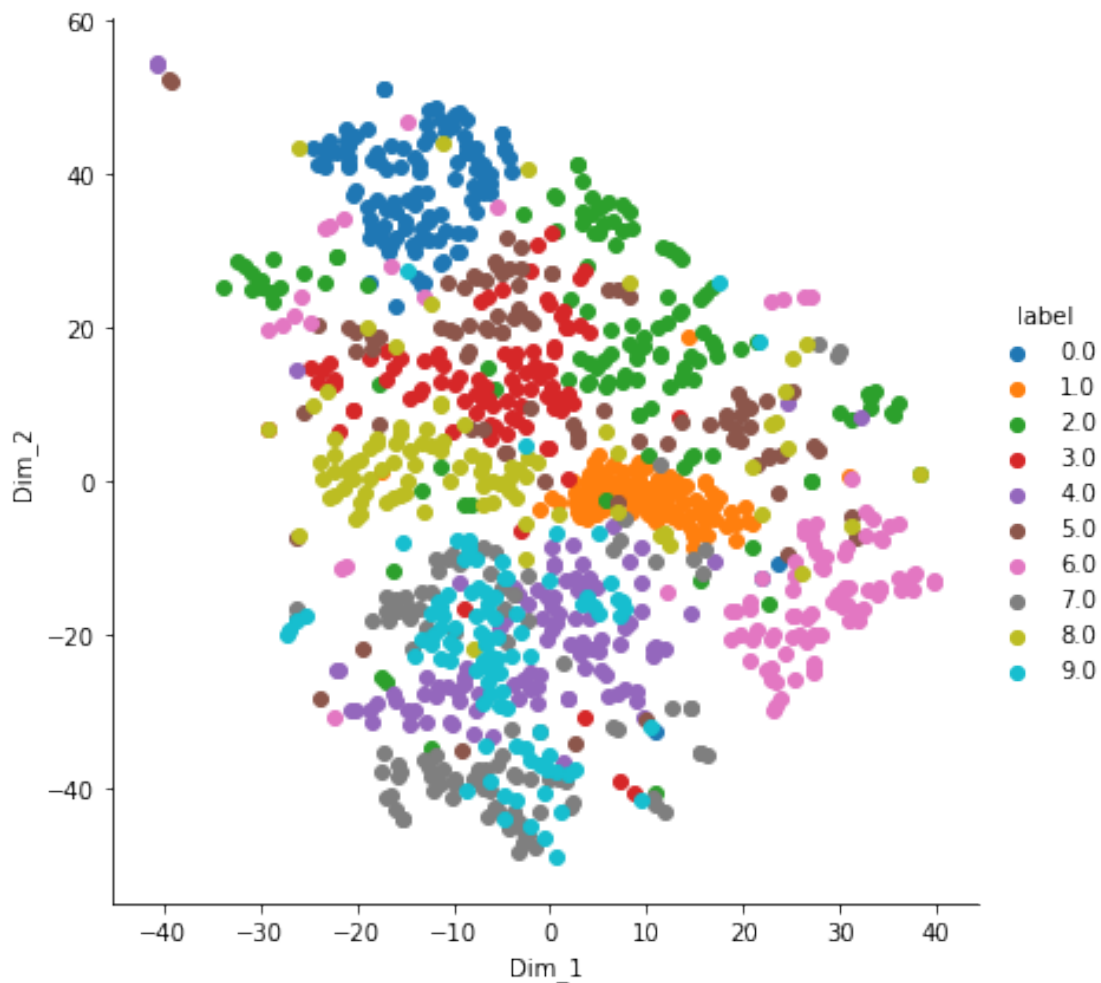
Iterations = 5000

```
In [11]: time_start = time.time()

         # Compute t-SNE (784-D to 2-D) for visualization
         model = TSNE(n_components=2,random_state=0,perplexity=50,n_iter=5000)

         tsne_data = model.fit_transform(p_data)
         print('t-SNE done! Time elapsed: {} seconds'.format(time.time() - time_start))

         tsne_data = np.vstack((tsne_data.T,p_labels)).T
         #print(tsne_data.shape)
         tsne_data[:4]

         tsne_df = pd.DataFrame(tsne_data,columns=['Dim_1','Dim_2','label'])
         tsne_df.head()
```

```
sns.FacetGrid(tsne_df,hue='label',height=6).map(plt.scatter, 'Dim_1', 'Dim_2').add_lege
```

t-SNE done! Time elapsed: 55.796788930892944 seconds

Iterations = 10000

```
In [12]: time_start = time.time()

         # Compute t-SNE (784-D to 2-D) for visualization
         model = TSNE(n_components=2,random_state=0,perplexity=50,n_iter=10000)

         tsne_data = model.fit_transform(p_data)
         print('t-SNE done! Time elapsed: {} seconds'.format(time.time() - time_start))

         tsne_data = np.vstack((tsne_data.T,p_labels)).T
```

```
#print(tsne_data.shape)
tsne_data[:4]

tsne_df = pd.DataFrame(tsne_data,columns=['Dim_1','Dim_2','label'])
tsne_df.head()

sns.FacetGrid(tsne_df,hue='label',height=6).map(plt.scatter, 'Dim_1', 'Dim_2').add_lege
```

t-SNE done! Time elapsed: 70.5398519039154 seconds
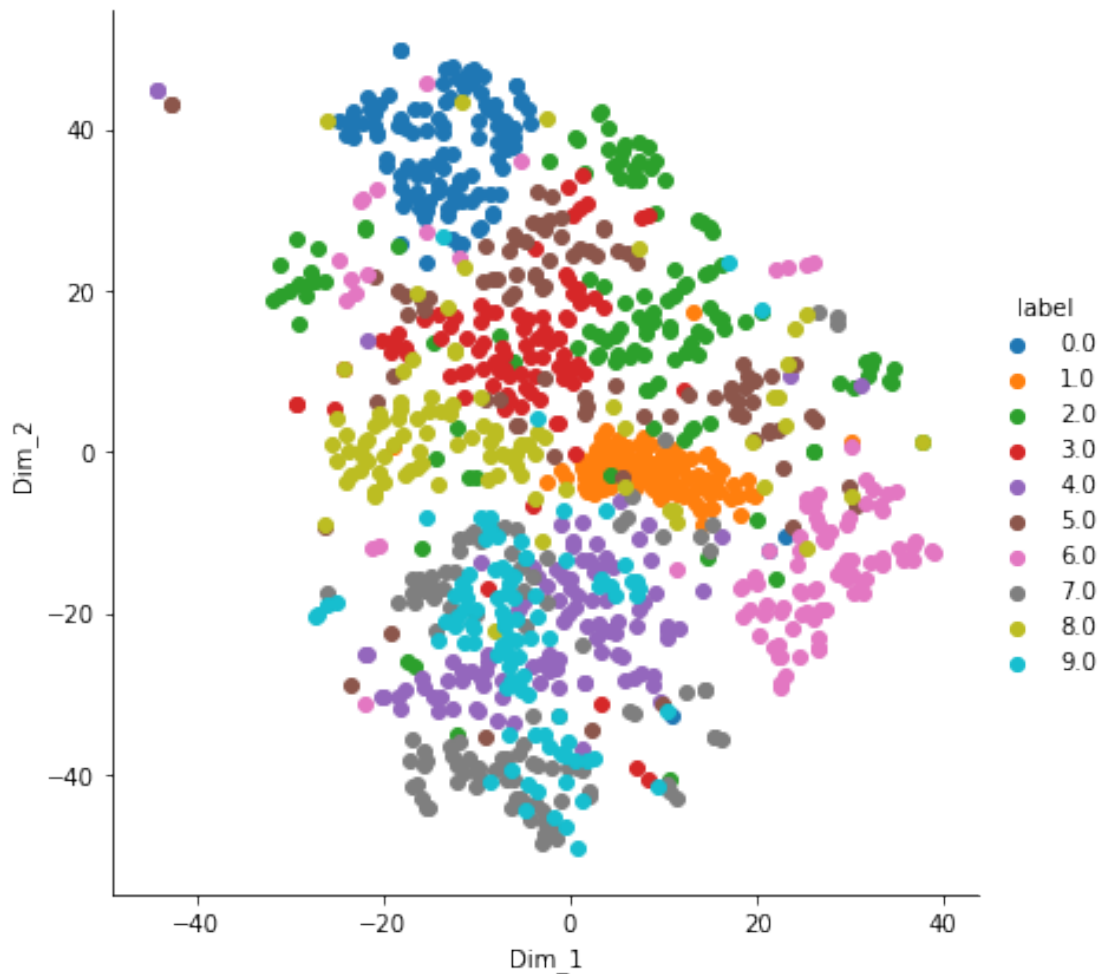
1.1.2   t-SNE over 5000 data points

In [13]: `no_of_data_points = 15000`

In [16]:
```
p_data = standardized_data[0:no_of_data_points,:]
p_labels = d_train_labels[0:no_of_data_points]
print('t-SNE Data Points {0} and its Labels {1}'.format(p_data.shape, p_labels.shape))
```

9

```
t-SNE Data Points (15000, 784) and its Labels (15000,)
```

Perplexity = default (30)

Iterations = 1000

```python
In [17]: time_start = time.time()

         # Compute t-SNE (784-D to 2-D) for visualization
         model = TSNE(n_components=2,random_state=0,perplexity=30,n_iter=1000)

         tsne_data = model.fit_transform(p_data)
         print('t-SNE done! Time elapsed: {} seconds'.format(time.time() - time_start))

         tsne_data = np.vstack((tsne_data.T,p_labels)).T
         #print(tsne_data.shape)
         tsne_data[:4]

         tsne_df = pd.DataFrame(tsne_data,columns=['Dim_1','Dim_2','label'])
         tsne_df.head()

         sns.FacetGrid(tsne_df,hue='label',height=6).map(plt.scatter, 'Dim_1', 'Dim_2').add_lege
```
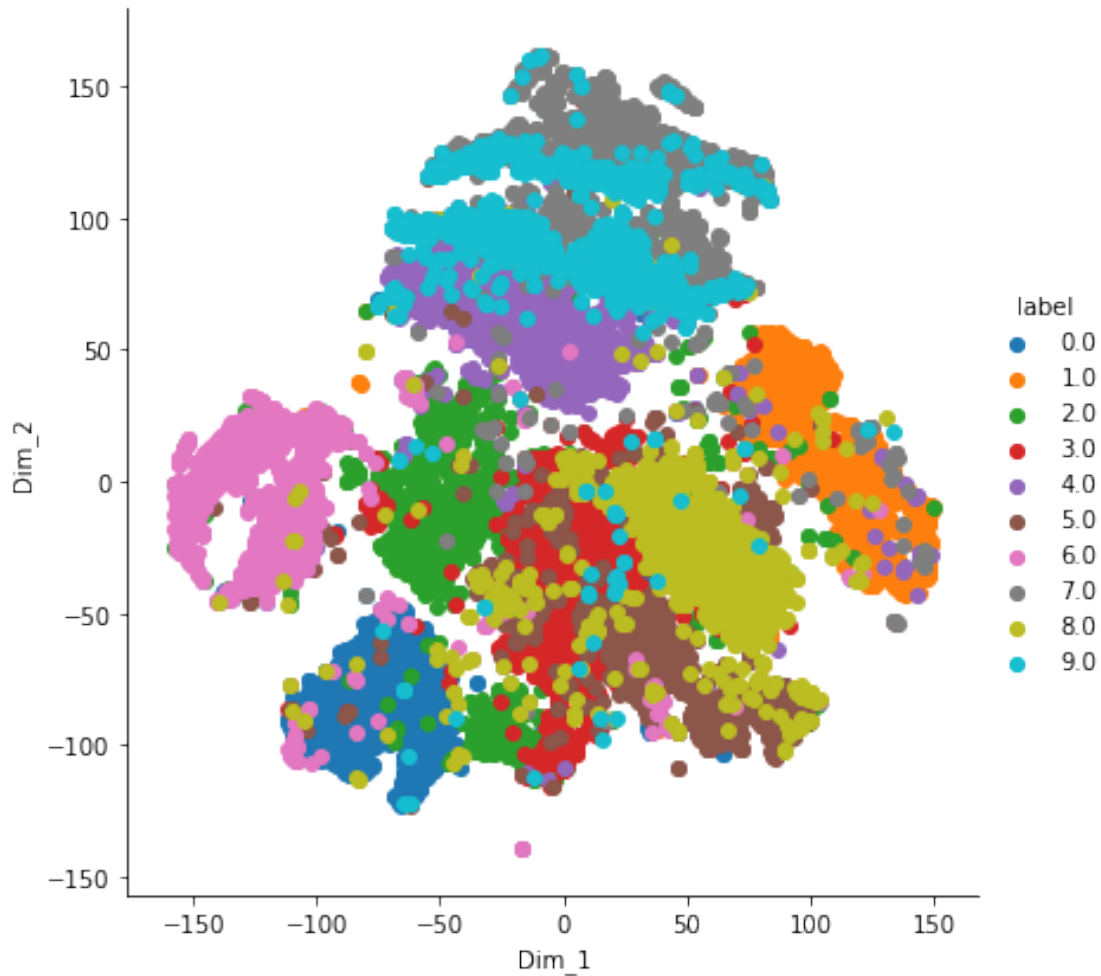
```
t-SNE done! Time elapsed: 500.472599029541 seconds
```
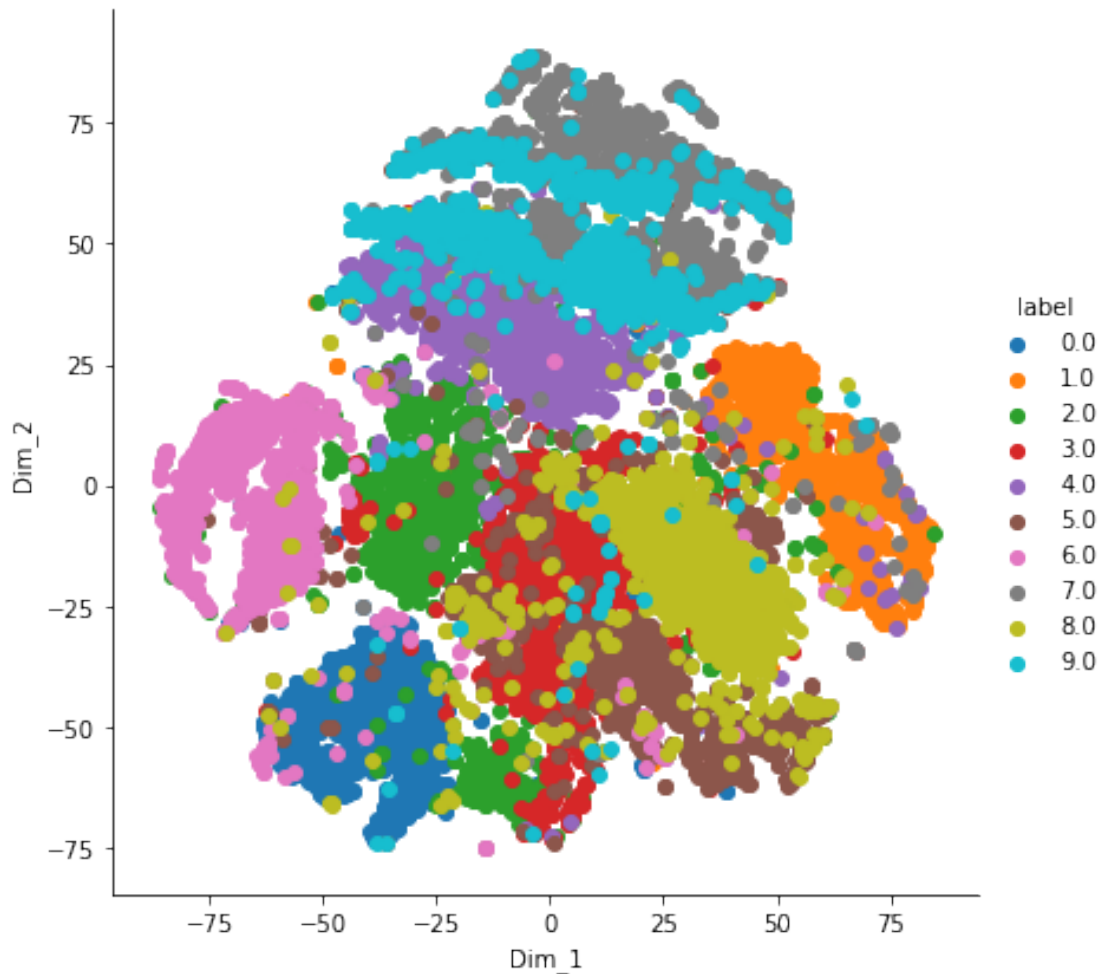
Iterations = 5000

```
In [18]: time_start = time.time()

         # Compute t-SNE (784-D to 2-D) for visualization
         model = TSNE(n_components=2,random_state=0,perplexity=30,n_iter=5000)

         tsne_data = model.fit_transform(p_data)
         print('t-SNE done! Time elapsed: {} seconds'.format(time.time() - time_start))

         tsne_data = np.vstack((tsne_data.T,p_labels)).T
         #print(tsne_data.shape)
         tsne_data[:4]

         tsne_df = pd.DataFrame(tsne_data,columns=['Dim_1','Dim_2','label'])
         tsne_df.head()
```
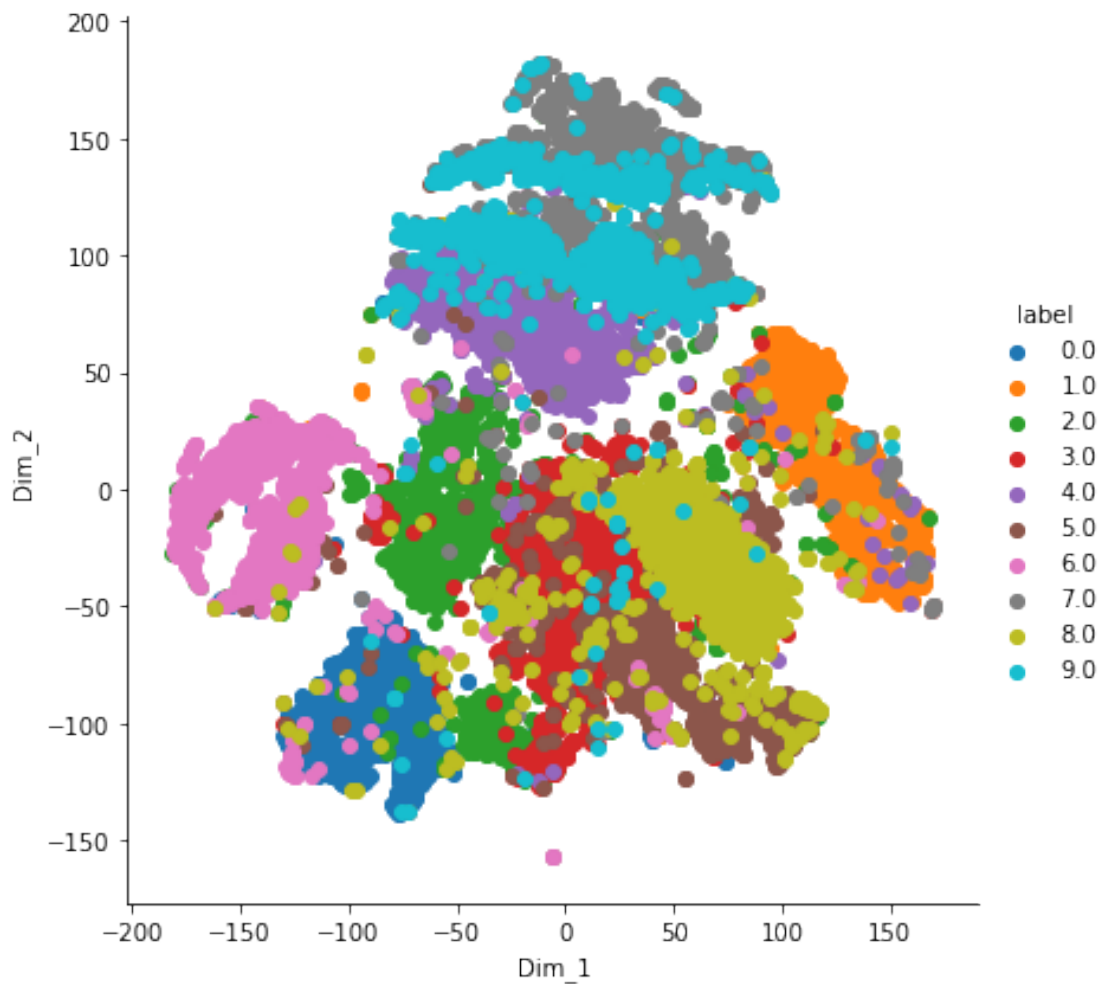
```
sns.FacetGrid(tsne_df,hue='label',height=6).map(plt.scatter, 'Dim_1', 'Dim_2').add_lege
```

t-SNE done! Time elapsed: 1315.6844096183777 seconds

Iterations = 10000

```
In [19]: time_start = time.time()

         # Compute t-SNE (784-D to 2-D) for visualization
         model = TSNE(n_components=2,random_state=0,perplexity=30,n_iter=10000)

         tsne_data = model.fit_transform(p_data)
         print('t-SNE done! Time elapsed: {} seconds'.format(time.time() - time_start))

         tsne_data = np.vstack((tsne_data.T,p_labels)).T
```

```
#print(tsne_data.shape)
tsne_data[:4]

tsne_df = pd.DataFrame(tsne_data,columns=['Dim_1','Dim_2','label'])
tsne_df.head()

sns.FacetGrid(tsne_df,hue='label',height=6).map(plt.scatter, 'Dim_1', 'Dim_2').add_lege
```

t-SNE done! Time elapsed: 2347.5857632160187 seconds

Perplexity = 50

Iterations = 1000

```
In [20]: time_start = time.time()
```

```
# Compute t-SNE (784-D to 2-D) for visualization
model = TSNE(n_components=2,random_state=0,perplexity=50,n_iter=1000)

tsne_data = model.fit_transform(p_data)
print('t-SNE done! Time elapsed: {} seconds'.format(time.time() - time_start))

tsne_data = np.vstack((tsne_data.T,p_labels)).T
#print(tsne_data.shape)
tsne_data[:4]

tsne_df = pd.DataFrame(tsne_data,columns=['Dim_1','Dim_2','label'])
tsne_df.head()

sns.FacetGrid(tsne_df,hue='label',height=6).map(plt.scatter, 'Dim_1', 'Dim_2').add_lege
```
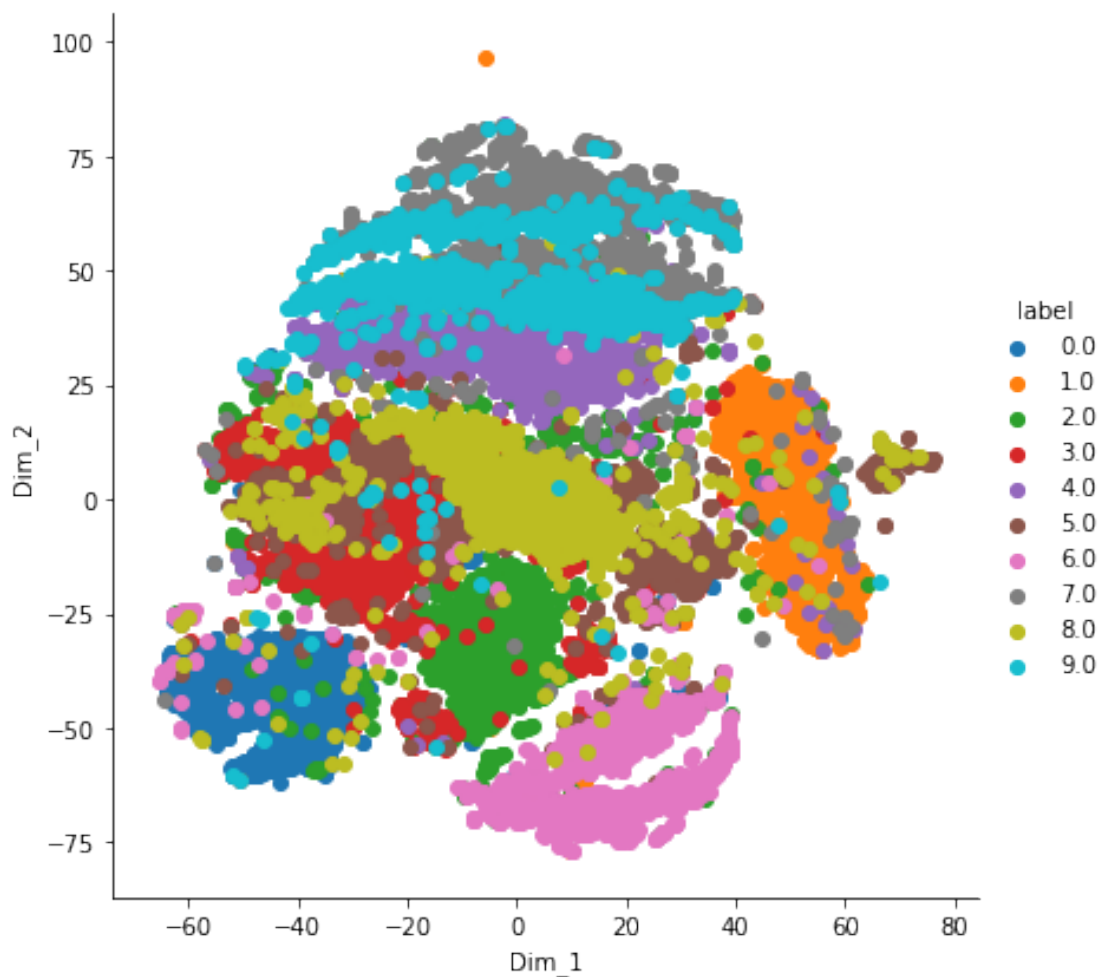
t-SNE done! Time elapsed: 660.0550405979156 seconds

Iterations = 5000

```
In [21]: time_start = time.time()

         # Compute t-SNE (784-D to 2-D) for visualization
         model = TSNE(n_components=2,random_state=0,perplexity=50,n_iter=5000)

         tsne_data = model.fit_transform(p_data)
         print('t-SNE done! Time elapsed: {} seconds'.format(time.time() - time_start))

         tsne_data = np.vstack((tsne_data.T,p_labels)).T
         #print(tsne_data.shape)
         tsne_data[:4]

         tsne_df = pd.DataFrame(tsne_data,columns=['Dim_1','Dim_2','label'])
         tsne_df.head()

         sns.FacetGrid(tsne_df,hue='label',height=6).map(plt.scatter, 'Dim_1', 'Dim_2').add_lege
```
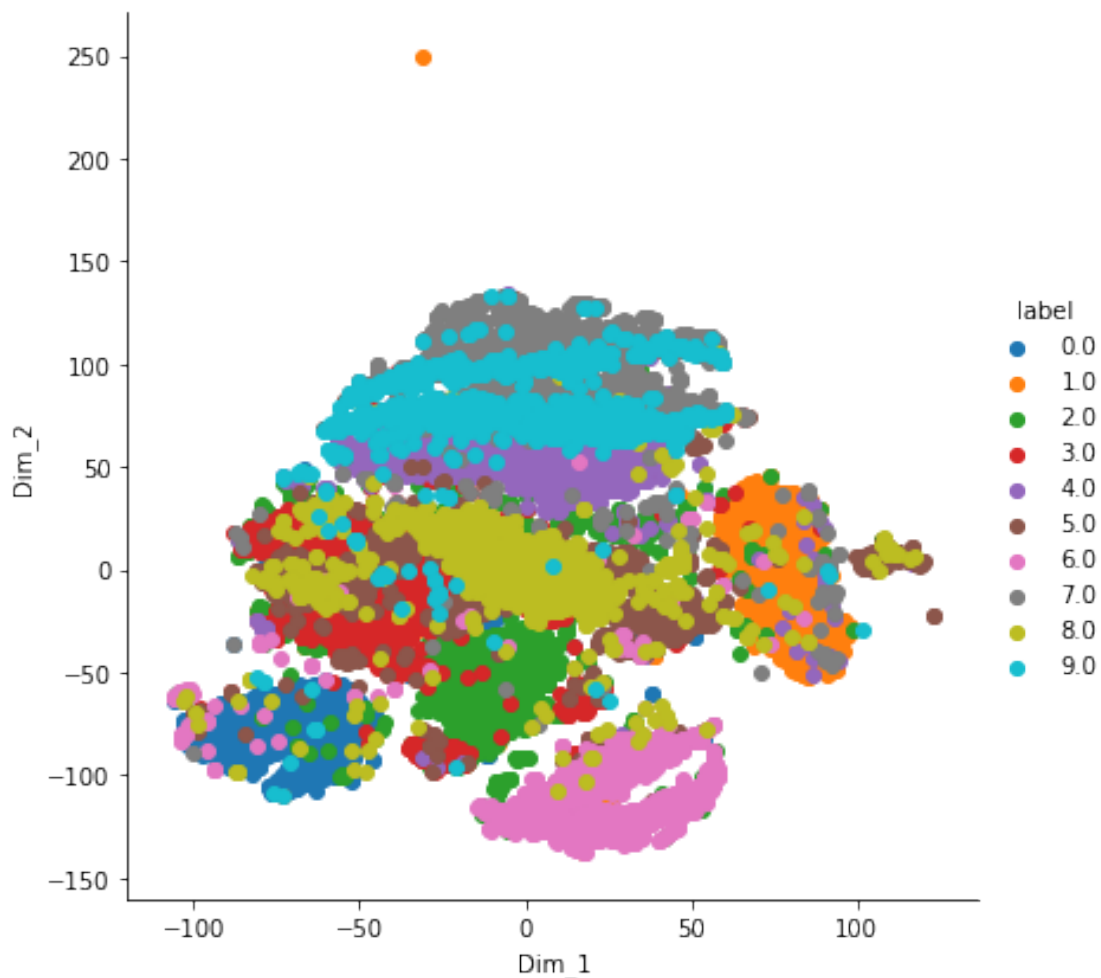
t-SNE done! Time elapsed: 1783.4331126213074 seconds

15

Iterations = 10000

```
In [22]: time_start = time.time()

         # Compute t-SNE (784-D to 2-D) for visualization
         model = TSNE(n_components=2,random_state=0,perplexity=50,n_iter=10000)

         tsne_data = model.fit_transform(p_data)
         print('t-SNE done! Time elapsed: {} seconds'.format(time.time() - time_start))

         tsne_data = np.vstack((tsne_data.T,p_labels)).T
         #print(tsne_data.shape)
         tsne_data[:4]

         tsne_df = pd.DataFrame(tsne_data,columns=['Dim_1','Dim_2','label'])
         tsne_df.head()

         sns.FacetGrid(tsne_df,hue='label',height=6).map(plt.scatter, 'Dim_1', 'Dim_2').add_lege
```
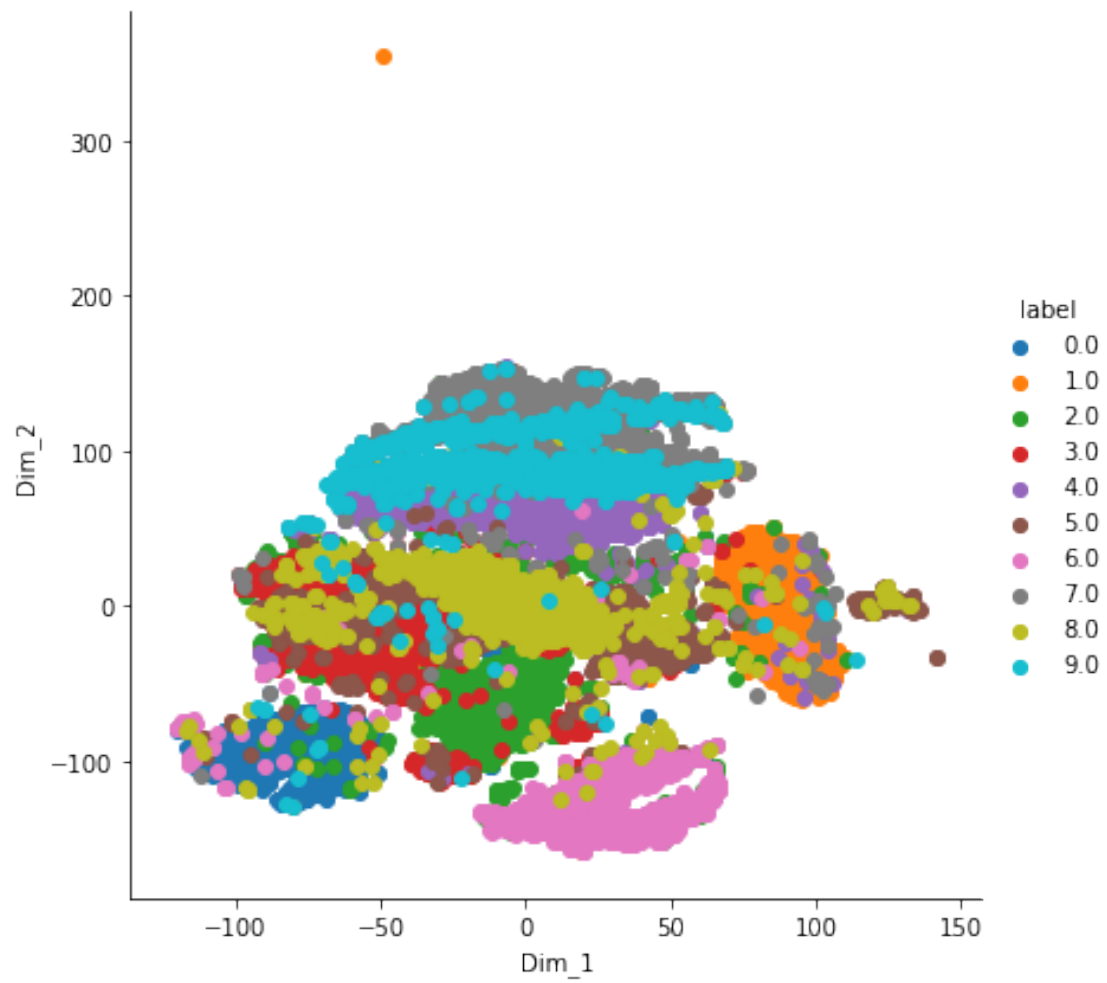
t-SNE done! Time elapsed: 3410.578033208847 seconds

2 Observation

TO DO