IT350 - Data Analytics

Annam Indhu Lekha

191IT207

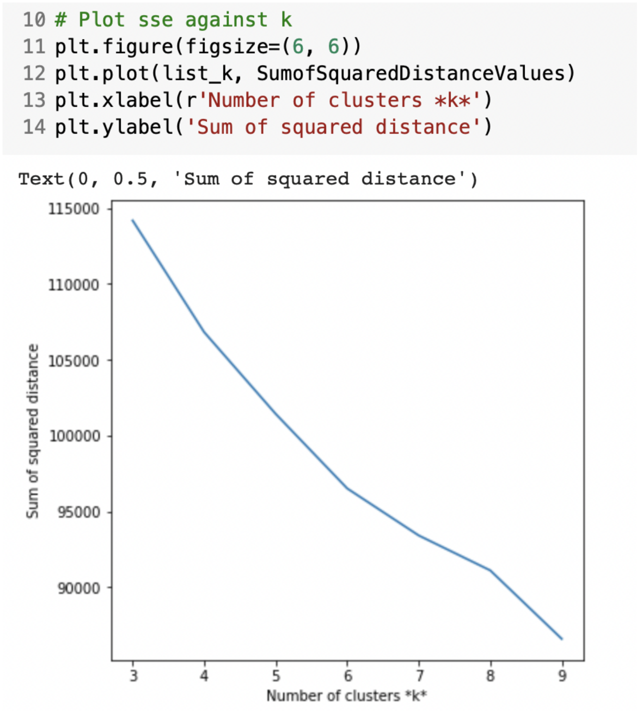
1) Find the clusters in the given dataset based on the content similarity and image similarity using k-means clustering and hierarchical clustering methods.

Data preprocessing - feature extraction

We have used the VGG model and removed the output layer manually. The new final layer is a fully-connected layer with 4,096 output nodes. This vector of 4,096 numbers is the feature vector that can be used to cluster the images.

K means clustering

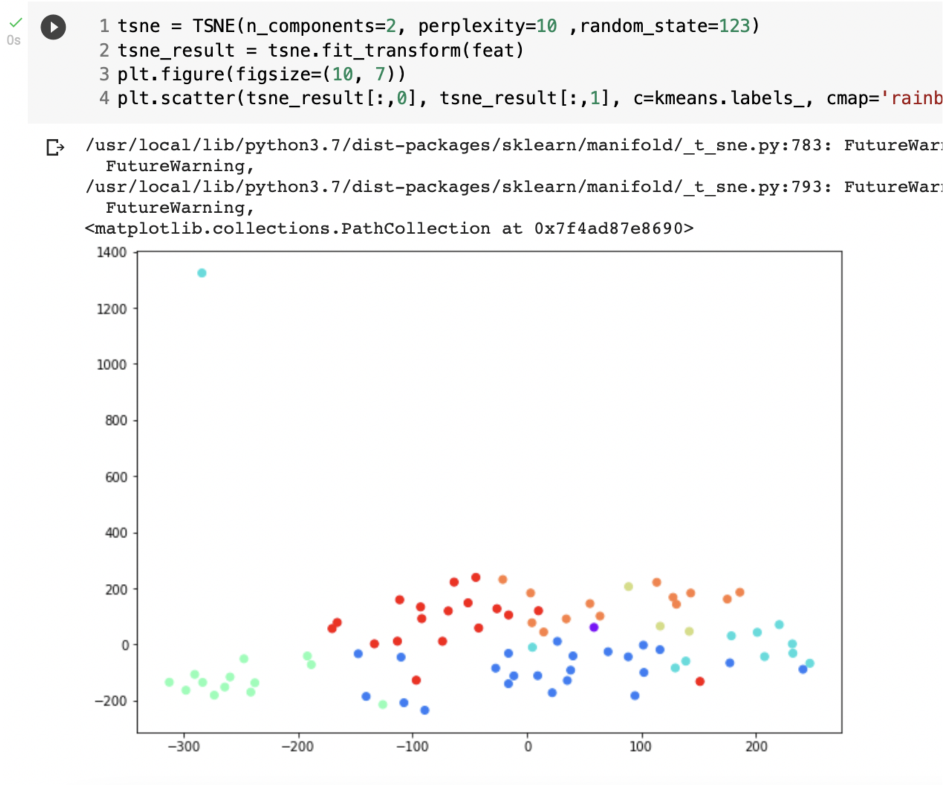
We chose a target number k, which refers to the number of centroids we wanted in the dataset. K means algorithm will allow us to group our feature vectors into k clusters. Each cluster contains images that are visually similar.



```
# Plot sse against k
plt.figure(figsize=(6, 6))
plt.plot(list_k, SumofSquaredDistanceValues)
plt.xlabel(r'Number of clusters *k*')
plt.ylabel('Sum of squared distance')
```

```
Text(0, 0.5, 'Sum of squared distance')
```

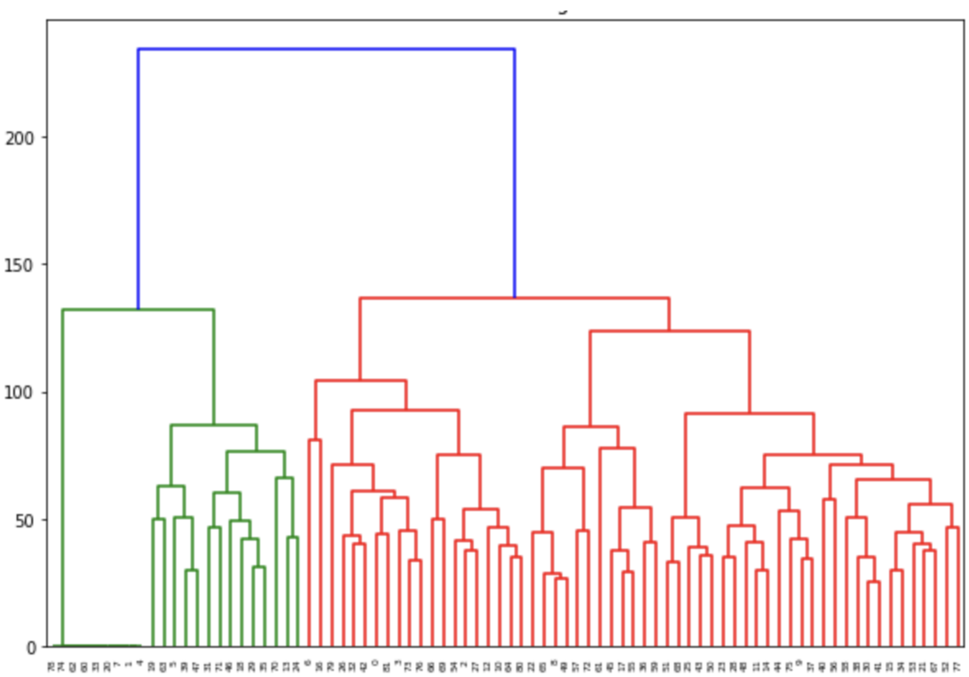
2D visualisation of Kmeans clusters



```
tsne = TSNE(n_components=2, perplexity=10 ,random_state=123)
tsne_result = tsne.fit_transform(feat)
plt.figure(figsize=(10, 7))
plt.scatter(tsne_result[:,0], tsne_result[:,1], c=kmeans.labels_, cmap='rainb
```

```
/usr/local/lib/python3.7/dist-packages/sklearn/manifold/_t_sne.py:783: FutureWar
  FutureWarning,
/usr/local/lib/python3.7/dist-packages/sklearn/manifold/_t_sne.py:793: FutureWar
  FutureWarning,
<matplotlib.collections.PathCollection at 0x7f4ad87e8690>
```

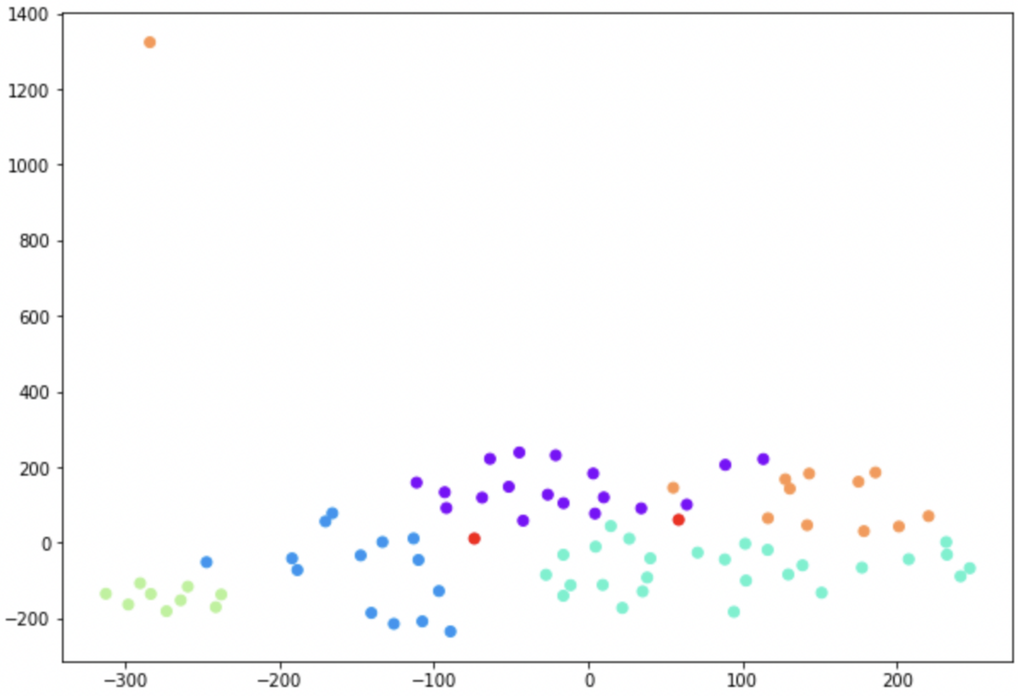
Hierarchical Clustering



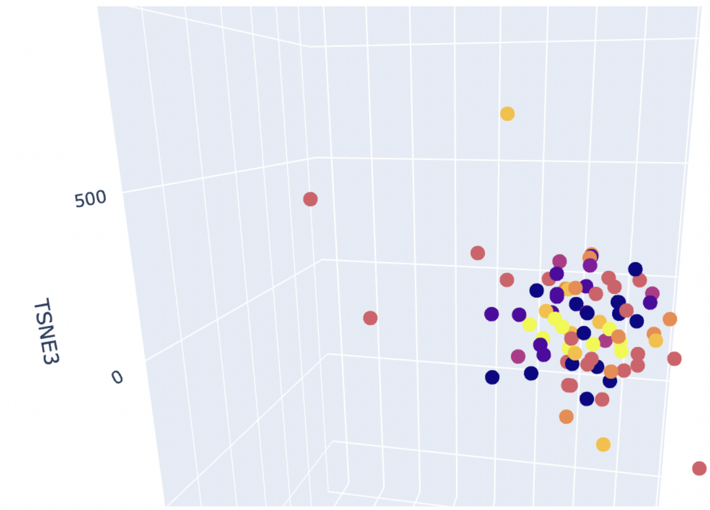
2) Visualization using TSNE

2 D visualization

```
<matplotlib.collections.PathCollection at 0x7f4ad1e1e610>
```



3 - D visualization



3) Evaluation

Clusters obtained by KMeans are justified by using the sum of squared distance from the nearest cluster. When K value is 6, this error is less, and it is a breakpoint from the plot.

In Clusters obtained by a hierarchical method, I had chosen the line in such a way that this horizontal line passes through the longest distance without a horizontal line. From this point I found 8 clusters were appropriate to use.

Link to colab -
https://colab.research.google.com/drive/1Iydx-fOmtTuME97BFKZNe510a753Fmeg?usp=sharing