Visual Analytics Homework-03

Nikhil Yadav

27th September 2017

Github link to jupyter notebook: `https://github.com/nikhil15iitd/visual_analytics_hw03/blob/master/hw03.ipynb`. The notebook may freeze in between if parameters are changed at a fast rate since it calls fit_transform for every parameter change, in that case re run the cells.
The notebook has embedded bokeh server, so should run on local machine with default notebook_url = 'localhost:8888' without calling bokeh serve command.
Visualization is done using scatter plots, with selectors & sliders for specifying number of clusters, affinity(distance metric to consider for clustering) & domain transform like PCA & t-SNE(Stochastic Neighbour Embedding).
Five different clustering algorithms were chosen:

- K-Means: performs good clustering, euclidean based metric (cannot be changed), non deterministic, highly reliant on initial placement of centroids.
- Spectral Clustering (slower to compute because rbf kernel transforms are expensive, so plot may not update fast): Performs clustering of points based on radial basis function(rbf) kernel by default, changing metric to "nearest_neighbor" shows similar clusters as KMeans algorithm
- Agglomerative Clustering: Highly stable clustering, changing n_clusters parameter does not have a significant impact on clustering. Metrics l1, l2 perform the same in most cases. The best metric seems to be euclidean in both t-SNE & normal domain.
- DBSCAN: considers all points as noise, only in t-SNE domain does it assign some points as clusters, does not have n_clusters parameter, hard to tune, the reason for its bad clustering is because data is **homogeneous** so no good clustering can be figured out based on density (all come in the same cluster). However, clustering can be seen if in t-SNE domain, epsilon value is changed to around 3.50 (which is maximum distance between points for them to be considered in same cluster)
- AffinityPropagation: performs good clustering, euclidean based, cannot be changed, however it requires no cluster parameter like K-Means or Agglomerative Clustering, so is easy to use, also robust & deterministic

Out of all affinities(euclidean, l1, l2, manhattan, cosine), euclidean works the best in t-SNE domain, & almost all clustering algorithms by default use euclidean as distance metric (affinity) for clustering data points. In PCA domain or default domain of 6 attributes, it is difficult to visualize the performance of clustering algorithms, because it is difficult to display clusters in 6 dimensions. PCA suffers from dimensionality reduction because its first principal component preserves the maximum variance & second holds no information, so again comparison is difficult. However, t-SNE embeds all dimensions in 2D, preserving information compared to PCA, so visualization can be done easily, compared to other domains.
Log normalize, x-axis, y-axis & point size selectors work only when domain transform = "None". By default, points are displayed after applying log transform, but clicking on Log normalize button will undo the operation & display clusters in normal domain
Snapshots of visualization:

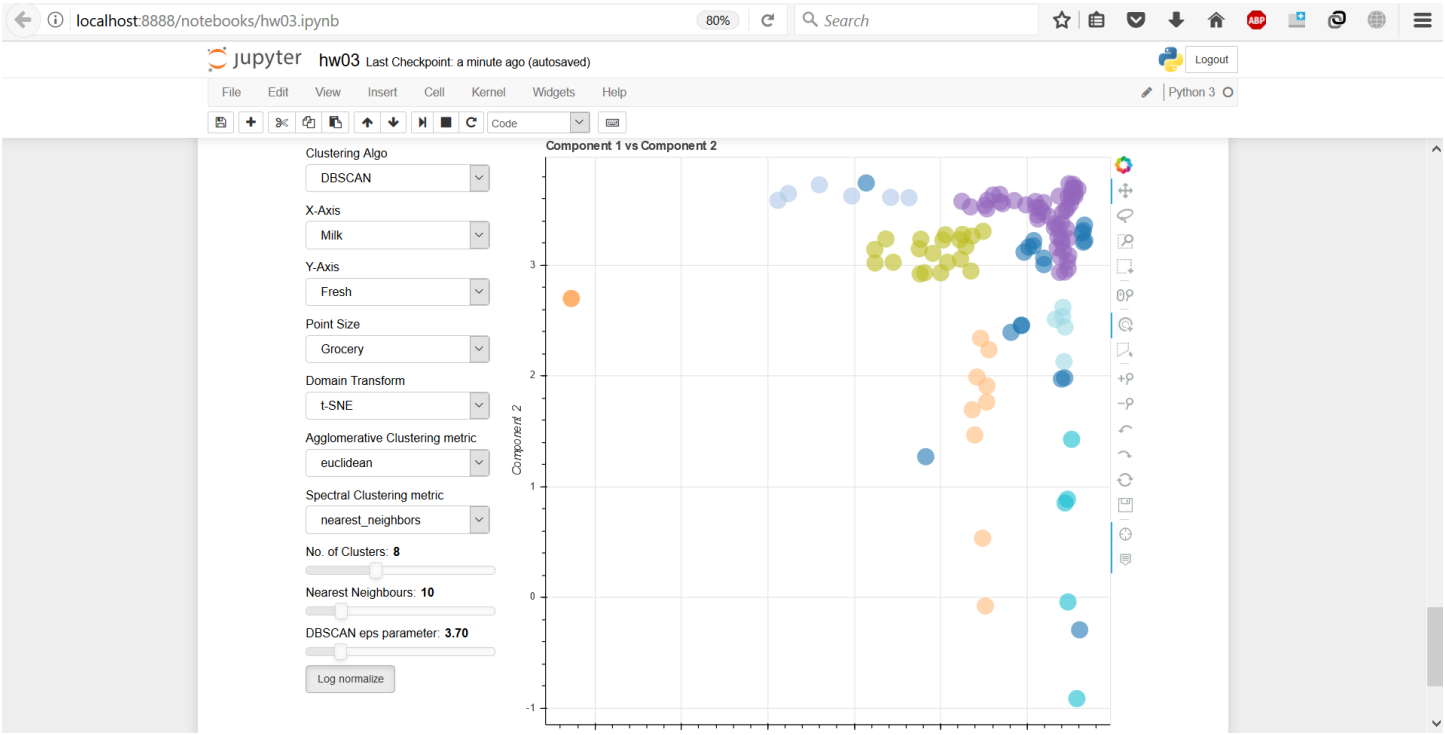

Figure 1: Scatter plot: Spectral clustering, t-SNE domain

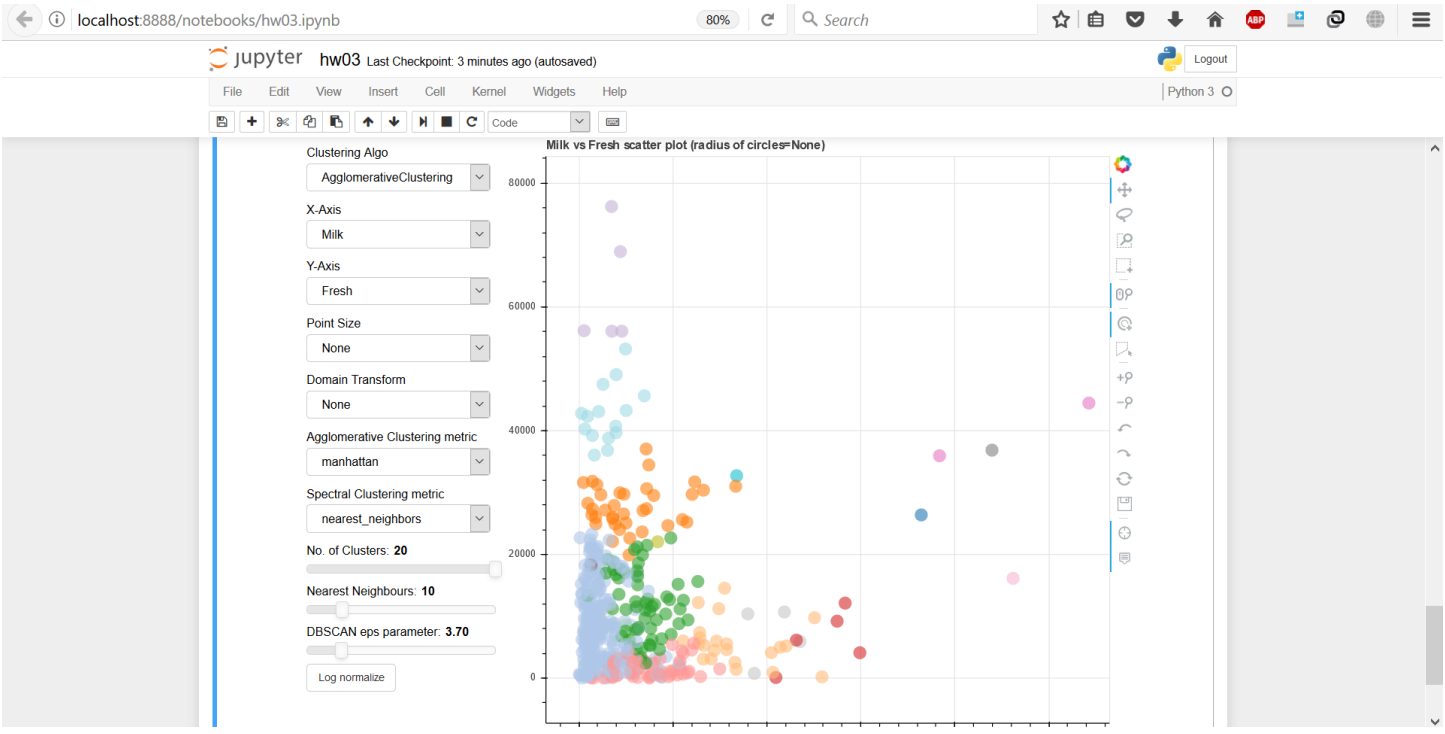

Figure 2: Scatter plot: Agglomerative clustering, PCA domain