Program 3
PR3: Text Clustering
By Chaithra Lakshmi Sathyanarayana

Rank: 21
NMI-Score: 0.4188

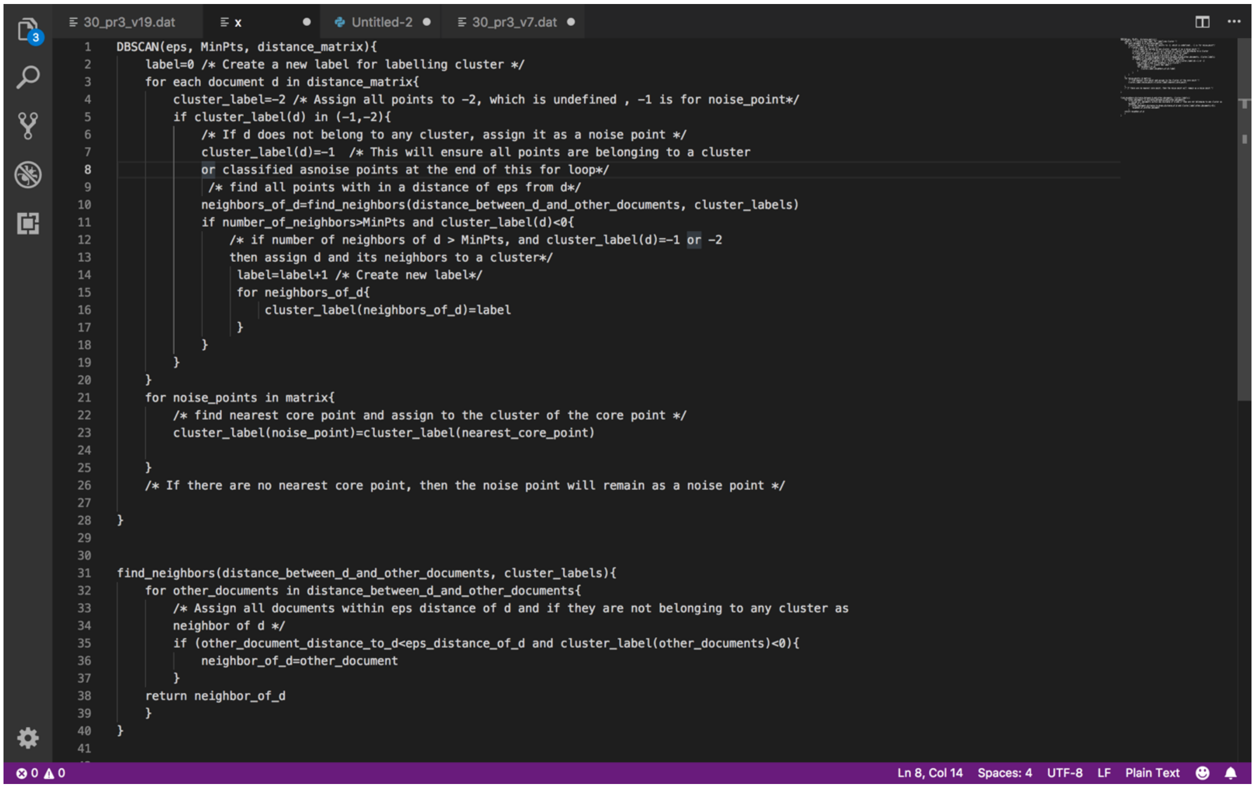
Approach

Training data has 8580 text records in sparse format. Each row represents a document. Each row has term followed by term frequency. The training data has to be clustered using DBSCAN algorithm.

1. From training data, build document matrix
2. Calculate distance between all documents and store it as distance matrix.
3. Determine the best value for eps by calculating kth nearest neighbor to each document and assign the elbow point as eps
4. Use DBSCAN to cluster documents for eps obtained above and varying the MinPts from 3 to 21
5. Calculate Silhouette score for the result obtained in step 4.

Pseudocode for DBSCAN

In dbscan algorithm, points are classified as core points, border points, noise points. A point is core point if the count of points within epsilon distance from the core point is greater than a specified number of point(MinPts). A border point has fewer than MinPts within epsilon distance, but is in close vicinity of a core point. All other points which are neither core point nor border point are called noise point.



```
DBSCAN(eps, MinPts, distance_matrix){
    label=0 /* Create a new label for labelling cluster */
    for each document d in distance_matrix{
        cluster_label=-2 /* Assign all points to -2, which is undefined , -1 is for noise_point*/
        if cluster_label(d) in (-1,-2){
            /* If d does not belong to any cluster, assign it as a noise point */
            cluster_label(d)=-1  /* This will ensure all points are belonging to a cluster
            or classified asnoise points at the end of this for loop*/
             /* find all points with in a distance of eps from d*/
            neighbors_of_d=find_neighbors(distance_between_d_and_other_documents, cluster_labels)
            if number_of_neighbors>MinPts and cluster_label(d)<0{
                /* if number of neighbors of d > MinPts, and cluster_label(d)=-1 or -2
                then assign d and its neighbors to a cluster*/
                 label=label+1 /* Create new label*/
                 for neighbors_of_d{
                     cluster_label(neighbors_of_d)=label
                 }
            }
        }
    }
    for noise_points in matrix{
        /* find nearest core point and assign to the cluster of the core point */
        cluster_label(noise_point)=cluster_label(nearest_core_point)

    }
    /* If there are no nearest core point, then the noise point will remain as a noise point */

}


find_neighbors(distance_between_d_and_other_documents, cluster_labels){
    for other_documents in distance_between_d_and_other_documents{
        /* Assign all documents within eps distance of d and if they are not belonging to any cluster as
        neighbor of d */
        if (other_document_distance_to_d<eps_distance_of_d and cluster_label(other_documents)<0){
            neighbor_of_d=other_document
        }
    return neighbor_of_d
    }
}
```

Fig: Pseudo code

For eps=0.25, below id the graph of silhouette score when MinPts was varied from 3 to 21 in steps of 2:

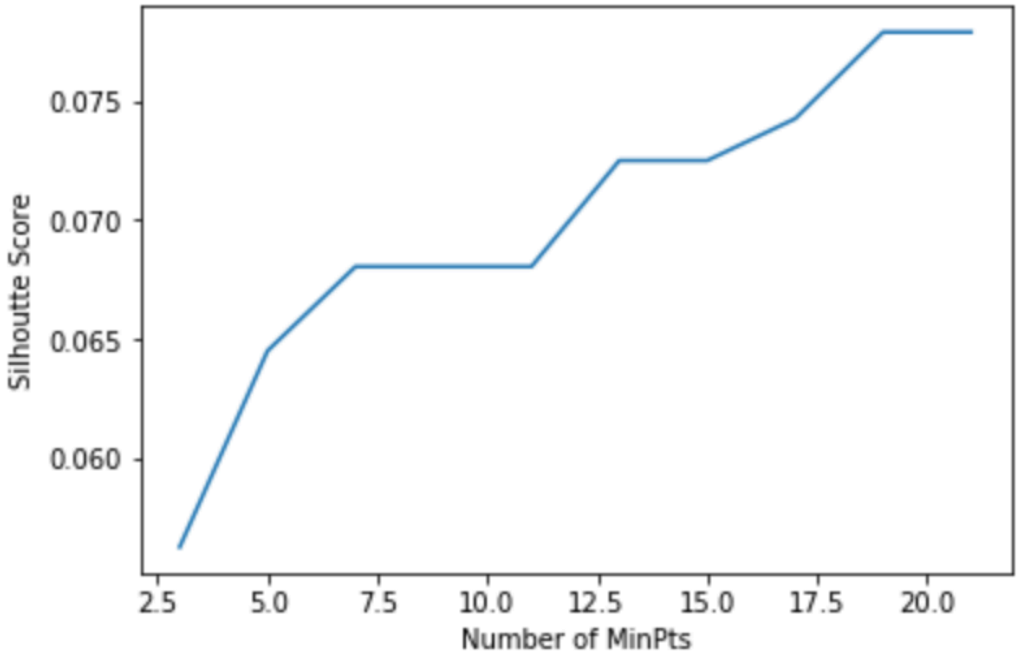


Fig: Number of MinPts vs Silhoette Score

Feature Selection:

Truncated SVD was used for dimensionality reduction. The input data has term numbers and not actual terms. It was not possible to identify what words were there in the input. Truncated SVD was used to reduce components to 8.

Conclusion

From MinPts vs Silhoette Score graph, we can see that as the number of MinPts increased silhouette score increased and then becomes a constant. When eps is too low, noise points increase. When eps is too high, there will be more points in few clusters. As eps increases, number of clusters decrease and then becomes a constant