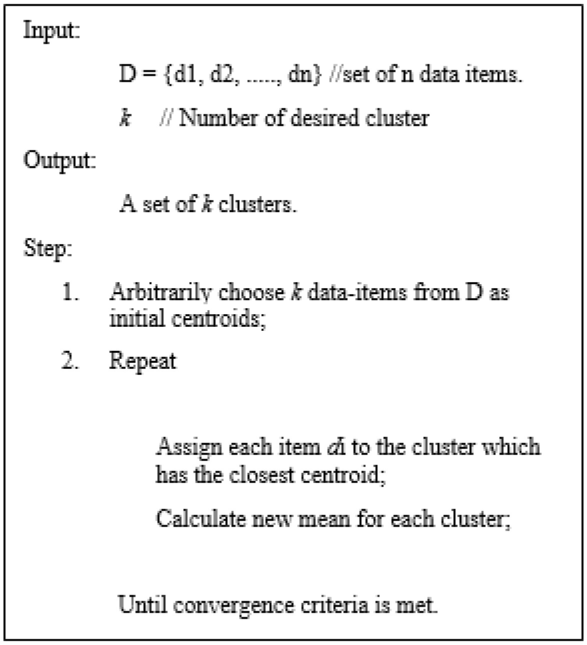
Graph Theory
Assignment-3

1. **Write the algorithm and time complexity for your project:**
   K- means clustering:
   - Select the number K to decide the number of clusters.
   - Select random K points or centroids. (It can be different from the input dataset).
   - Assign each data point to their closest centroid, which will form the predefined K clusters.
   - Calculate the variance and place a new centroid of each cluster.
   - Repeat the third steps, which means assign each datapoint to the new closest centroid of each cluster.
   - If any reassignment occurs, then repeat step-4 else end.
   - As we are trying to classify our tweets into 3 sentiments, we are considering 3 clusters (Positive feedback,Negative feedback, Neutral feedback)

Pseudocode:



```
Input:
        D = {d1, d2, ....., dn} //set of n data items.
        k    // Number of desired cluster
Output:
        A set of k clusters.
Step:
    1.  Arbitrarily choose k data-items from D as
        initial centroids;
    2.  Repeat

            Assign each item di to the cluster which
            has the closest centroid;
            Calculate new mean for each cluster;

        Until convergence criteria is met.
```

Time complexity (K-Means) :O(n^2)

MST based clustering :

- Compute the complete graph from dataset
- Construct the minimum spanning tree of the complete graph
- Find the inconsistent edges in the MST
- Inconsistent edges partition the graph into clusters.
    - Here we created an MST, where we are using TF-IDF vectors for similarity metric, on the basis of the TF-IDF matrix, MST is created.
    - Once MST is created we need to find 2 longest edges, to break the graph into 3 clusters (Positive feedback,Negative feedback, Neutral feedback)

Pseudocode:

```
Input: a set of n data points (D) and a distance function (dist)
Output : k clusters.
Algorithm:
    1. Create an n x n distance matrix - dist_matrix that stores the
       distance between each pair of data points in D.
    2. Compute the minimum spanning tree (MST) of the graph with
       dist_matrix as its adjacency matrix.
    3. Remove the k-1 largest edges from the MST to obtain k
       connected components (clusters)
```

Time complexity (MST based clustering) :O(n^(3/2))

Construction of MST requires O(E+Vlog(E+V))

For finding the consistent edges -

Sorting requires O(VlogV)

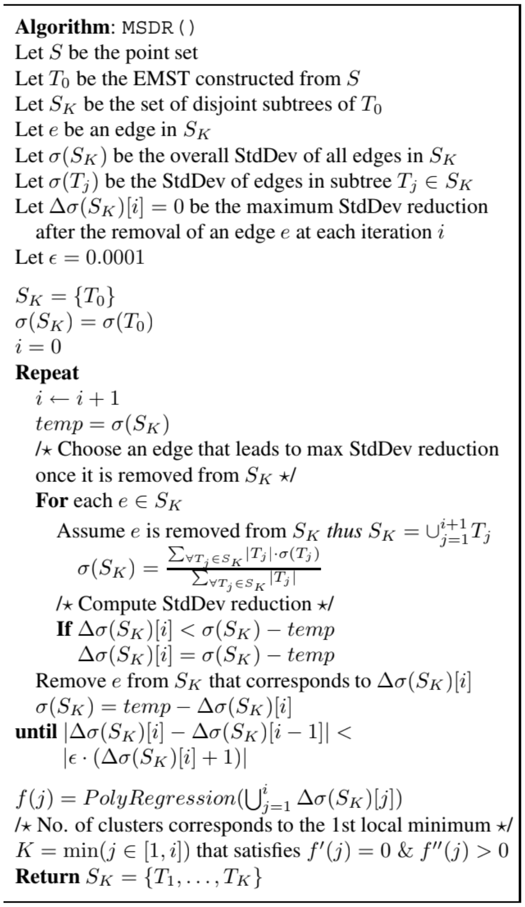
Finding the max requires O((K-1)(V-1))

Finding the connected components takes O((K-1)(E+V))

Overall time complexity - O((E+V)logV)

MSDR

- ➔ Compute the complete graph from dataset
- ➔ Construct the minimum spanning tree of the complete graph
- ➔ Find the inconsistent edges in the MST with the help of standard deviation.
- ➔ Inconsistent edges partition the graph into clusters.
    - ◆ Here we created an MST, where we are using TF-IDF vectors for similarity metric, on the basis of the TF-IDF matrix, MST is created.
    - ◆ In this algorithm we are using standard deviation to remove 2 edges to break the graph into 3 clusters (Positive feedback,Negative feedback, Neutral feedback)



```
Algorithm: MSDR()
Let $S$ be the point set
Let $T_0$ be the EMST constructed from $S$
Let $S_K$ be the set of disjoint subtrees of $T_0$
Let $e$ be an edge in $S_K$
Let $\sigma(S_K)$ be the overall StdDev of all edges in $S_K$
Let $\sigma(T_j)$ be the StdDev of edges in subtree $T_j \in S_K$
Let $\Delta\sigma(S_K)[i] = 0$ be the maximum StdDev reduction
   after the removal of an edge $e$ at each iteration $i$
Let $\epsilon = 0.0001$

$S_K = \{T_0\}$
$\sigma(S_K) = \sigma(T_0)$
$i = 0$
Repeat
   $i \leftarrow i + 1$
   $temp = \sigma(S_K)$
   /⋆ Choose an edge that leads to max StdDev reduction
   once it is removed from $S_K$ ⋆/
   For each $e \in S_K$
      Assume $e$ is removed from $S_K$ thus $S_K = \cup_{j=1}^{i+1} T_j$
         $\sigma(S_K) = \frac{\sum_{\forall T_j \in S_K} |T_j| \cdot \sigma(T_j)}{\sum_{\forall T_j \in S_K} |T_j|}$
      /⋆ Compute StdDev reduction ⋆/
      If $\Delta\sigma(S_K)[i] < \sigma(S_K) - temp$
         $\Delta\sigma(S_K)[i] = \sigma(S_K) - temp$
   Remove $e$ from $S_K$ that corresponds to $\Delta\sigma(S_K)[i]$
   $\sigma(S_K) = temp - \Delta\sigma(S_K)[i]$
until $|\Delta\sigma(S_K)[i] - \Delta\sigma(S_K)[i-1]| <$
      $|\epsilon \cdot (\Delta\sigma(S_K)[i] + 1)|$

$f(j) = PolyRegression(\bigcup_{j=1}^{i} \Delta\sigma(S_K)[j])$
/⋆ No. of clusters corresponds to the 1st local minimum ⋆/
$K = \min(j \in [1, i])$ that satisfies $f'(j) = 0$ & $f''(j) > 0$
Return $S_K = \{T_1, \ldots, T_K\}$
```

Time Complexity (MSDR) :

MSDR does not have a well-defined time complexity in terms of big O notation since it is a heuristic method that depends on the number of features and the number of iterations required for convergence. Therefore, it is not possible to express the time complexity of MSDR in terms of big O notation.

2. **Description of the dataset:**

Our dataset consists of 14640 tweets. It is a record of tweets about airlines in the US. Along with other information, it contains ID of Tweet, sentiment of tweet (neutral, negative and positive), reason for negative tweet, name of airline and text of tweet.

From the given dataset we need to perform sentiment analysis, i.e. we need to classify all the tweets in 3 different clusters and also if a new tweet comes we should be able to add it to its respective cluster.

3. **Experimental result:**

Silhouette coefficient: It is a measure of how well each data point in a cluster is separated from other clusters in a clustering solution. It measures the compactness and separation of the clusters simultaneously.

So for comparing the cluster qualities for all three methods we are using the Silhouette coefficient.

Cluster Quality	**K-means**	**MST based Clustering**	**MST clustering using maximum standard reduction**
Dataset	0.011890576824406708	-0.9583333333333334	-0.9613821138211383

Execution Time (in seconds)	**K-means**	**MST based Clustering**	**MST clustering using maximum standard reduction**
Time	3.2	46.8	826.5

All the above scores are not calculated on the complete dataset. Here we are considering a small subset of our dataset due to system restrictions. We are not able to run our complete dataset for MST based clustering and MSDR because of its large size