Pseudo-code for the clustering algorithm based on a least-squares distance measure

The idea of the algorithm

The aim of the algorithm is to segment the dynamic PET data into k clusters based on a weighted least-squares distance measure. The clustering is done in such a way that the weighted least-squares distance measure is minimised.

The algorithm has four main steps that are shortly described in the following.

1. The first step is to define the initial cluster centroids. It is suggested that these centroids are initialised randomly.
2. The second step is to compute distances from all cluster centroids for all pixels and assign each pixel to the nearest cluster. The distance measure used is the Euclidian distance measure $D(x,y) = ||x-y||$, where $x, y \in R^n$. Weighting factors can also be used.
3. The third step is to calculate new cluster centroids. A cluster centroid is defined to be the average of TACs in the cluster.
4. The final step is to calculate the value of a cost function that is based on a least squares distance measure. If the cost function value has decreased since the previous iteration, the process is repeated from step 2.

Pseudo-code

/* Input is a matrix A constituting of m rows (pixels) and n columns (time frames). An element of A in place (i,j) is denoted a_{ij}. k is the number of clusters */

/* Define weighting factors for each time frame. They can for example be chosen to be proportional to the scanning intervals of the experiment. */

```
for j = 1 to n
    w_j = weighting factor
endfor
```

/* Set the initial cluster centres to be random numbers. m_{tj} denotes the value of the mean TAC in cluster t in time frame j. */

```
for t = 1 to k
    for j = 1 to n
        m_tj = random number
    endfor
endfor
```

/* Function **Dist(i,t)** returns the Euclidian distance measure of TAC i and cluster t. This function is needed later in the clustering algorithm .*/

```
Dist (i,t)
  dist  = 0
  while  j ≤ n
         dist  = dist + (a_ij – m_tj)^2 * w_j
         j = j + 1
  endwhile
  return dist


oldf  = ∞     // initial value for the cost function must be a large number
changed  = true

while  changed = = true

        /* Assign each TAC to the nearest cluster according to the Euclidian distance measure */

      for i = 1 to m
          mindist  = ∞
          c_i  = 0
          t  = 1
          while t ≤ k
                d  = Dist(i,t)
                if  d ≤ mindist
                   then mindist  = d
                        c_i  = t       // c_i is the cluster where TAC i belongs
                   endif
                   t  = t + 1
          endwhile
      endfor

        /* Compute the cluster centroids  */

       for t  = 1 to k
         for j  = 1 to n
             num = 0   // number of TACs in cluster t
             sum  = 0  // sum of TAC values in cluster t in time frame j
             i  = 1
             while  i  ≤ m
                   if c_i = t
                     then num  = num + 1
                          sum  = sum + a_ij
                   endif
                   i = i + 1
             endwhile
             m_tj = sum / num
         endfor
       endfor
```

```
        /* Compute the value of the cost function using the new cluster centroids. */

            f = 0
            while  t ≤ k
                   s  = 0
                   while  i ≤ m
                          s  = s + Dist(i,t)
                          i  = i + 1
                   endwhile
                   f  = f + s
                   t  = t + 1
            endwhile

        /* Make the comparisons to decide whether to continue iterating or not.  */
            if   oldf  – f > 0.001
               then changed  = true
                    oldf  = f
               else  changed  = false
            endif
endwhile
```

Reference

K. P. Wong et al.: Segmentation of dynamic PET Images Using Cluster Analysis. *IEEE Transactions on Nuclear Science* 49:200–207(2002)