Documentation for the kernel k-medoids algorithm

General Description

The Matlab script kmedoid.m uses a distance matrix (representing dissimilarity between models) to define locations of Earth models using Multi-Dimensional Scaling (stored in the vector Y) and then performs the kernel k-medoids algorithm. A medoid can be defined as the object of a cluster, whose average dissimilarity to all the objects in the cluster is minimal i.e. it is a most centrally located point in the given data set.

Description of the parameters

Input Parameters

- **d**: dissimilarity distance matrix (not necessarily Euclidean). The distance matrix can be a full dissimilarity matrix, or an upper triangle matrix such as is output by the Matlab function 'pdist'.
- **nbclusters**: number of clusters to split the data into
- **maxIteration**: the maximum number of iterations to run the algorithm for

Output Parameters

- **Clustering.idxtosimulate**: indices of the medoids selected by the clustering algorithm. Length of vector is nbclusters.
- **Clustering.T**: vector of size the number of Earth models, which indicates to which cluster the realization belongs to.
- **Y:** location of Earth models in MDS

Matlab function required

- cmdscale.m (in Matlab)
- rbf_kernel.m
- plotcmdmap_KKM.m
    - Only to plot the clusters – not used to perform the clustering

Description of the code

1. Multi-Dimensional Scaling (MDS)

The function Matlab cmdscale performs MDS. It takes an n-by-n distance matrix D, and returns an n-by-p configuration matrix Y. Rows of Y are the coordinates of n points in p-dimensional space for some p < n. It also returns the eigenvalues (e) of Y*Y'. If the first k elements of e are much larger than the remaining (n-k), then you can use the first k columns of Y as k-dimensional points whose inter-point distances approximate D.

- `[Y_, e_] = cmdscale(d)`

The dimension of the MDS space is chosen to keep 99% of the total energy.

2. Kernel Matrix

In case of kernel medoids, the definition of the distance between any two points requires the definition of a kernel matrix. The function Matlab rbf_kernel defines a Gaussian radial basis kernel function, with a bandwidth sigma.

sigma can be defined as 20% of the maximum distance in the distance matrix

3. Kernel K-Medoids

- Choose the initial centers by selecting the points at random
- Assign each point to the closest medoid
    - The distance is computed in the Feature space defined by the kernel matrix (where points behave more linearly). The distance in the Feature space is defined as a function of the kernel (see below)
    - Compute the distance between each point and each medoid:
        - `dist_points_medoids(i,j)= K(i,i)-2*K(i,k(j))+ K(k(j),k(j));`
    - Assign each point to the cluster defined by the closest medoid. T is a vector is size npoint, which contains the cluster number the Earth models belongs to
        - `[B,T] = min(dist_points_medoids,[],2)`
- While the clustering configuration varies do:
    - Compute the distance between points in a same cluster
        - `dist_within_cluster`
    - The point with the average distance to other points minimal is the new medoid
        - `[dclust idx_min] = min(mean(dist_within_cluster));`
    - Update the vector containing the new medoids
        - `k_new(i) = idx_in_clusters(idx_min);`
    - Compute the distance between each point and each medoid:

- Assign each point to the cluster defined by the closest medoid. T is a vector is size npoint, which contains the cluster number the Earth model belongs to