Heart Disease Clustering

Phase 1

- The input data is 22 features and 319,795 samples
- The Output of the covariance matrix is 22x22 features (We used two different methods to compute the covariance matrix and obtained the same results.)
- eigenvalues are 22 eigenvalues, and eigenvectors are 22x22 features. (We used the built-in and scratch methods to compute the eigenvalues and eigenvectors and obtained the same results.)
- We created the matrix Q by sorting the normalized eigenvectors in descending order of eigenvalues.
- We tried different numbers of components (k), and the best result was 21 because the reconstruction error was the lowest.
- Best reconstruction error 1625.209 (From 5171.140 to 1625.209)
- We transformed the original matrix [F' = Q (F - m)] to the transformed matrix and reconstructed the transformed matrix (inverse) [F = (Q - 1 * F') + m].
- We truncated matrix Q (take some values) from Q, and then the new F length is smaller than the old (reduce dimensionality).
- The code iterates over different numbers of retained components (k_values) and reconstructs the data using the selected principal components. The reconstruction error is computed for each iteration.
- Best Results The best results are reported based on the minimum reconstruction error. The optimal number of retained components (best_k), the corresponding Q matrix (best_Q_matrix), and the minimum reconstruction error are printed.
- This report provides insights into the analysis, including the sorted eigenvalues and eigenvectors, the normalization of eigenvectors, and determining the best reconstruction using PCA. The output will show the best Q matrix, the optimal number of retained components, and the associated reconstruction error.

Phase 2

When employing PCA in conjunction with fuzzy C-means clustering, it appears that the algorithm converges more efficiently, displaying a notable improvement in convergence quality compared to using the entire dataset. This observation is particularly evident when the dataset is smaller and simpler than our own.

Fuzzy C-means

- It is dividing the data points into a set of clusters using the membership function between each point in the dataset and all clusters of centroids.
- A particular member of the set may be a member of several clusters with different values of membership.
- Input for FCM: Data, C (number of clusters)
- Output for FCM: Mij (membership matrix), Cj (cluster centroid) [1 <=j<= C], [1<=i<=n]

steps for Fuzzy C-mean (FCM)

1. Assume the number of clusters to be made C. Such that: 2<=C<=N (N: number of samples)
2. Choose an appropriate level of cluster fuzziness. Such that: $g > 1$
3. Initialize the NxC-sized membership matrix [M] at random such that: (a) Mij∈ [0.0,1.0] and their sum should be 1
4. Compute centroids
5. Calculate the Euclidean distance between each data point i-th and j6. Update fuzzy membership matrix [M] according to dij
6. Repeat until the changes in [M] come out to be less than some pre-specified values

Team Member

- Mohamed Hisham (20200483)
- Abdelrahman Amin (20200311)
- Abdelrahman Mostafa (20200827)
- Yossef Mohamed (20200669)
- Esraa Abdelmoneam (20201015)