Lisan Al Ghaib

# 1 In search of Spice

**References**

1. Array creation routines
2. Logic Functions
3. Broadcasting
4. Linear Algebra
5. Random Generation
6. K Means CLustering

## The House of Harkonnens' Dilemma

The House of Harkonnens, the great spice miners of Arrakis, faced a grave dilemma. The extraction of the precious spice, Melange, was fraught with uncertainty. The spice blooms in remote regions, hidden beneath the endless dunes, and its distribution was irregular and unpredictable.

Legend has it that an old wise woman, inspired by the prophecy of the Lisan-al-Gaib, devised a method to unveil the spice's secret locations. She believed that the spice's distribution held patterns, invisible to the naked eye but decipherable through the whispers of data.

Her method, known as the Lisan-al-Gaib Algorithm, became a beacon of hope for the spice miners. This algorithm takes as input the available locations where traces of spice were found (called spice points), and finds out potential locations of spice nodes (or spice centers). Every spice center has a cluster of spice points, or conversely, every spice point belongs to a cluster of a spice center.

The algorithm is explained below.

## Algorithm

**Input:**

1. A dataset containing 2D coordinates of locations where traces of spice were found.

2. $K$ (guess of number of spice nodes).

**Output:** Potential locations of $K$ spice centers/nodes.
**Steps:**

1. For each of the spice points, maintain a label (0 to $K-1$, corresponding to the cluster of the spice center it belongs to).
2. Initialize with a random guess for the coordinates of $K$ spice centers. Choose them randomly from the $K$ spice locations themselves.
3. **Loop:**
   (a) **E Step:** (Keeping the coordinates of the $K$ spice centers fixed, update the labels)
      i. For each spice point, find the nearest spice center, and assign the spice point to the cluster of that spice center.
      ii. To do this, for each spice point, compute the Euclidean distance to all spice centers and assign the label corresponding to the spice node with the shortest distance.
   (b) **M Step:** (Keeping the labels of spice points fixed, update the coordinates of the spice centers)
      i. The new coordinate of the $i^{\text{th}}$ spice center is the mean of the coordinates of all spice points belonging to the cluster with label $i$.
   (c) Repeat the loop until the labels do not change.

## Notes:

- Do not change anything other than where asked for filling the TODOs in `spice.py`.
- You need to achieve the required output without using loops. Each additional `for`/`while` incurs a penalty of $-2$. Do not use them even within comments!
- To visualize the final results of clustering of spice nodes, run the `run.py` script. You can change the `data_path` and value of $K$ (number of clusters) and observe the output clustering. You are given with `kmeans_1.png` (`data_path = 'spice_locations.txt', K = 2`) and `kmeans_2.png` (`data_path = 'spice_locations2.txt', K = 4`).

## Your Task

Complete the TODO in spice.py.