MapReduce distributed computing

Write a program simulating the MapReduce distributed computing model for constructing a sample mean and sample covariance matrix from homework #1.

Specifically, let $(x_1, x_2, \dots, x_n)$ be a sequence of column vectors:

$$x_i = \begin{bmatrix} x_i^1 \\ \vdots \\ x_i^m \end{bmatrix}, \qquad i = 1, \dots, n.$$

Write a MapReduce-style program which computes the **sample mean** vector

$$X = \frac{1}{n} \sum_{i=1}^{n} x_i$$

and the **sample covariance** (or **variance-covariance**) matrix

$$V = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - X)(x_i - X)^T,$$

where x^T is the transpose of the column vector x.

- Use the canonical information representation designed in the homework #1.
- As a guiding example, you can use the code “Mean_MR.py”, which illustrates the calculation of the arithmetic mean for a set of numbers.
- To generate a set of random columns with a given average and covariance matrix, you can use the code from “MV_ArGen.py”.