Exercise 1
In the sub-folder ‘dim-reduction’ you can find a serial script performing PCA in Python and another subfolder ‘data’ containing the file of the dataset for radar returns from the ionosphere, downloaded from:
http://archive.ics.uci.edu/ml/datasets/Ionosphere

Encode the dataset in a cudf dataset and call the cuML function to perform the PCA.
Therefore run the new code on Leonardo single GPU and measure runtime.
Plot runtime comparison of serial PCA (written from scratch), scikit-learn PCA, and cuML PCA.

Exercise 2
Following the steps description code a kernelPCA algorithm, from scratch, first using only numpy and scipy. Compare it with the scikit-learn version.
Then, try to parallelize the code you wrote using only dask delayed and distributed.
Run this code on leonardo and measure the time.
Then write a Python code calling the dask-ml respective function and compare the runtime of this parallel version with the one obtained by your code.
Then write a Python code calling the cuml function and run it on Leonardo GPUs.

Exercise 3
In the folder clustering you can find a serial version of the most common algorithm for clustering: K-MEANS.
Parallelize this code using only dask delayed and distributed, and compare its runtime with the version of dask-ml and cuml.