PROJECT 02

WHEAT SEED K-MEANS CLUSTERING
WITH C AND CUDA

DAVID NGUYEN

D_NGUYEN@CSU.FULLERTON.EDU

CALIFORNIA STATE UNIVERSITY, FULLERTON

CPSC 479

HIGH PERFORMANCE COMPUTING

DOINA BEIN

Table of Contents

Abstract

Project 2 of California State University, Fullerton’s, CPSC-479 High Performance Computing, where we are tasked to apply high performance computing to a data science problem utilizing a parallel computing library such like MPI, openMP or CUDA.

This project showcases a single pass of k-means clustering, implemented with C, CUDA and Python.

Pseudocode

```
Read in dataset and store into respective arrays.

Choose initial centroids.

Allocate memory to GPU device.

Copy arrays to be used into GPU memory using CUDA function - cudaMemcpy(…)

Call GPU device function to run a single pass of k-means clustering.
   1. Compute Euclidean distances to all centroids.
   2. Set predicted value based on smallest Euclidean distance.

Copy predicted values array from GPU memory back to host memory using CUDA function - cudaMemcpy(…)

Call function to compute accuracy of current centroids and output.

Call function to update centroid coordinates.

Output data of features used, predicted values, and updated centroid values to csv.

Make scatterplots using matplotlib.
```

How to run program

To run program:

1. In terminal, change to root directory with file wheat_cluster.cu
2. Run `**nvcc wheat_cluster.cu**` to build the program.
3. Run `**./a.out**` to run program
4. Run `**python make_plots.py**` to make a scatterplot.
5. Success! You can view your plot in the *plots* directory.

Refer to follow pages for screenshots of example outputs.

Screenshots

Example Outputs

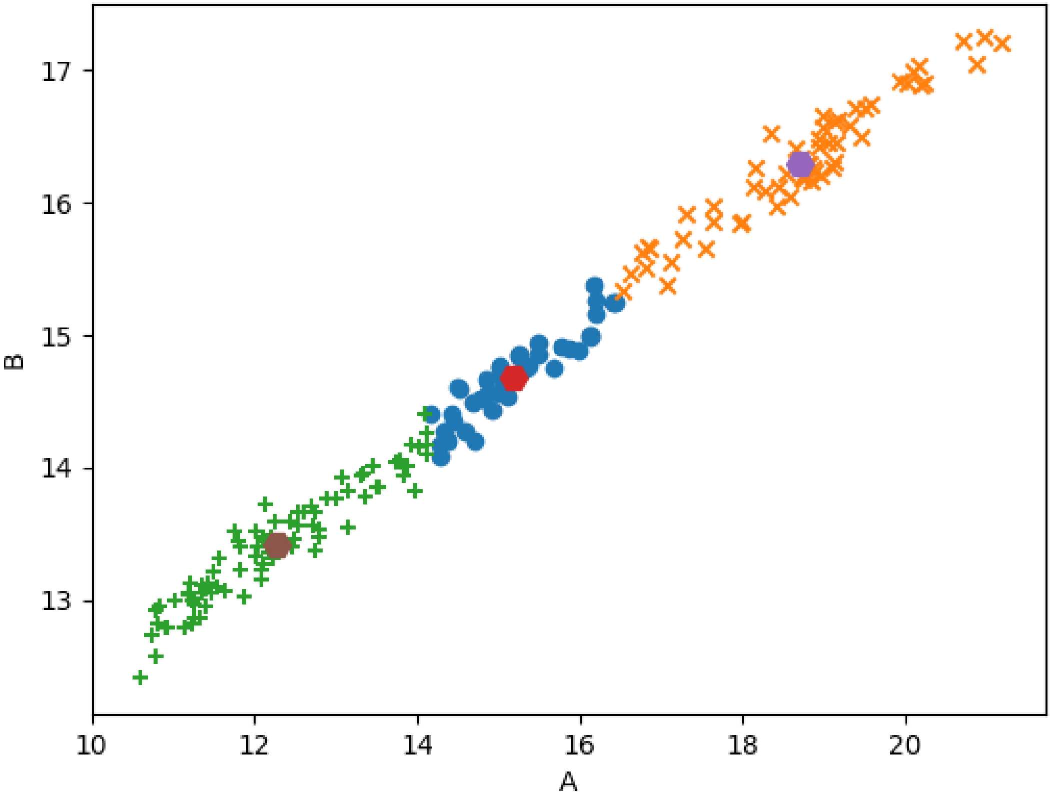

Figure 1. 1 pass of k-means clustering of Area vs. Perimeter