PC-2016/17 Course Project
Implementation in CUDA of KMeans algorithm

Tommaso Ceccarini
E-mail address
tommaso.ceccarini1@stud.unifi.it

Federico Schipani
E-mail address
federico.schipani@stud.unifi.it

Abstract

KMeans algorithm is one of the most popular method for clustering analysis. In our work we provide a CUDA implementation that use an Nvidia GPU to solve the clustering problem. We also provide a performance analysis with the purpose of compare the performance of our parallel CUDA implementation with a sequential implementation written in C language.

1. Introduction

KMeans algorithm is one of the most popular method for clustering analysis. The purpose of the cluster analysis is that of divide data into meaningfull group, called cluster. The resultant cluster should then capture the structure of the data.

KMeans methods attempt to do this by evaluating a similarity measure according to the mean value of the data that are cointained in the clusters. So, given a set of observation $(\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots, \boldsymbol{x_N})$ where each observation is a $P - dimensional$ real vector, k-means clustering aims to partition the N observation into $K(\leq N)$ sets $\boldsymbol{S} = \{S_1, S_2, \ldots, S_K\}$ so as to minimize the within-cluster sum of squares. In other words, its objective is to find:

$$\arg\min_{\boldsymbol{S}} \sum_{i=1}^{k} \sum_{x \in S_i} ||\boldsymbol{x} - \boldsymbol{\mu}_i||^2 \tag{1}$$

where μ_i is the mean of points in S_i.[1]

Algorithm 1 is the pseudocode of an iterative implementation of KMeans algorithm. Before the *while* cycle there are the initial assignment step. In this step the first k data are assigned to the first k cluster. A pseudocode of this initial assignment is showed in Algorithm 1

Algorithm 1 KMeans

```
procedure KMEANSSEQUENTIAL(data[n][p])
    mean[k][p], oldMean[k][p]
    assignment[n]
    (mean, assignment) ← $initAss(data)$
    while !$stop$ do
        assignment ← $calcMin$(mean, data)
        oldMean ← mean
        mean ← $calcMean$(assignment, data)
        stop ← $stopCriterion$(mean, oldMean)
    end while
    return mean
end procedure
```

Algorithm 2 Initial Assignment

```
procedure INITASS(data[n][p])
    mean[k][p]
    assignment[n]
    for i = 0; i < k; i + + do
        assignment[i] = i
        for j = 0; j < p : j + + do
            mean[i][j] = data[i][j]
        end for
    end for
    return (mean, assignment)
end procedure
```

References

[1] Wikipedia. K-means clustering — wikipedia, the free encyclopedia, 2017. [Online; accessed 23-January-2017]. 1