Projecte TGA

Implementació KNN en CUDA

Documentació

*2n Quadrimestre - curs 2019/2020*

Laia Batlle i Pau Ballber



# 

[**Introducció**](#_u3qdoaf9czkw) **3**

[**Definició del problema a resoldre**](#_fn3so1kmref1) **3**

[**Implementacions CUDA realitzades**](#_lv4r1sfxl15s) **4**

[**Anàlisi de rendiment**](#_30ku0xf1hv4e) **4**

[**Bibliografia**](#_6fesdorgzn9y) **4**

# 

# Introducció

En aquest projecte el que es vol fer és implementar una versió de l’algorisme k-nearest neighbors utilitzant CUDA. Per fer-ho aprofitarem les eines que tenim disponibles per aconseguir a base d’optimitzacions, millores en el rendiment del nostre programa.

Com que ja hi ha moltes implementacions seqüencials d’aquest algorisme publicades i nosaltres el que realment volem és explotar l’utilització de CUDA el que hem fet ha estat partir d’una de les implementacions més bàsiques que hem trobat en llenguatge C per adaptar-la a la nostra manera. El codi seqüencial en el que ens hem basat es pot consultar a l’apartat bibliogràfic d’aquesta documentació.

El nostre objectiu principal per adaptar aquest algorisme a CUDA és aprofitar l’utilització de kernels per paral·lelitzar part del nostre codi i obtenir millores en el temps d’execució significatives. Així doncs, podrem analitzar el rendiment del programa per cada millora afegida o si és el cas investigar possibles inconsistències.

# Definició del problema a resoldre

El k-nearest neighbor, o simplement kNN, és un algorisme d’aprenentatge automàtic molt utilitzat tant per a la classificació estadística com per a l’anàlisi de la regressió. En ambdós casos l’entrada consisteix en les dades d’entrenament més properes a l’espai de funcions. La sortida, en canvi, depèn de si s’utilitza per a la classificació o la regressió:

En la regressió kNN, la sortida és el valor de la propietat per a l’objecte. Aquest valor és la mitjana dels valors dels k veïns més propers.

D’altra banda, en la classificació kNN, la sortida és una pertinença a una classe. Es tracta d’una aproximació a la classificació de dades que estima la probabilitat de que un punt de dades sigui membre d’un grup o d’un altre depenent de a quin grup es troben els punts de dades que l’envolten. Més concretament, l’algorisme calcula la distància de l’element nou a cada un dels existents i ordena aquestes distàncies de menor a major per anar seleccionant el grup al qual ha de pertànyer. Aquest grup serà, per tant, el de major freqüència amb menors distàncies.

Normalment, k és un nombre enter positiu, petit i imparell. Si k és igual a 1, l’objecte s’assigna a la classe d’aquest veí més proper. Com més gran sigui k, més precisa serà la classificació, però més temps es triga a realitzar-la.

# Implementacions CUDA realitzades

# Anàlisi de rendiment

# Bibliografia

**KNN**.

* Pàgina d’on hem extret el codi de l’algorisme seqüencial:

<https://www.geeksforgeeks.org/k-nearest-neighbours/>

* Documentació utilitzada per entendre el problema:
* <https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm>
* <https://www.techopedia.com/definition/32066/k-nearest-neighbor-k-nn>
* <https://www.analiticaweb.es/algoritmo-knn-modelado-datos/>