Fundamentos de Computação de Alto Desempenho

UNIVERSIDADE DE ÉVORA
ESCOLA DE CIÊNCIAS E TECNOLOGIA
DEPARTAMENTO DE INFORMÁTICA
Nuno Lourenço
23/24

Projeto: K-means distribuído

O K-means é um algoritmo de clustering utilizado em análise de dados para agrupar pontos de dados num número pré-determinado de grupos (ou clusters) com base nas suas características. A ideia principal é minimizar a soma das distâncias quadráticas entre os pontos nos dados e a média (centroide) dos respetivos grupos."

O K-means é um método iterativo que divide um conjunto de M observações em K grupos (clusters), onde cada observação pertence ao grupo cuja média (centroide) é a mais próxima. O processo segue os seguintes passos:

- **Inicialização:** Selecionam-se inicialmente K pontos aleatórios do conjunto de dados como os centroides iniciais dos grupos. Estes pontos podem ser escolhidos aleatoriamente ou através de uma heurística.
- **Atribuição ao grupo**: Cada ponto no conjunto de dados é atribuído ao grupo cujo centroide tem a menor distância euclidiana em relação ao ponto. Esta distância é geralmente calculada usando a fórmula da distância euclidiana entre dois pontos.
- **Atualização dos centroides**: Após todos os pontos terem sido atribuídos a um grupo, recalcula-se o centroide de cada grupo. Isto é feito tomando a média de todos os pontos que foram atribuídos ao grupo na etapa anterior.
- **Iteração:** Repetem-se os passos 2 e 3 até que os centroides não mudem significativamente entre as iterações ou até que se atinja um número máximo de iterações. Isso indica que o algoritmo atingiu a convergência.
- **Finalização:** O processo termina quando não há alterações significativas nos centroides ou após um número fixo de iterações, resultando numa partição do conjunto de dados em k grupos.

O pseudocódigo é o seguinte:

K-means

```
Entradas: o número de clusters a considerar K, e os dados N pontos
Saídas: o cluster a que pertence cada ponto, e os respetivos centroides

Início do K-means

  Escolhe k pontos aleatórios como centroides iniciais

  Repete até à convergência:
    Para cada ponto no conjunto de dados:
      Atribui o ponto ao grupo cujo centroide é o mais próximo
    Para cada grupo:
      Atualiza o centroide para a média de todos os pontos atribuídos ao grupo

  Retorna os grupos e os seus centroides
Fim
```

Fundamentos de Computação de Alto Desempenho

UNIVERSIDADE DE ÉVORA
ESCOLA DE CIÊNCIAS E TECNOLOGIA
DEPARTAMENTO DE INFORMÁTICA
Nuno Lourenço
23/24

Objetivo

O sistema a desenvolver deve permitir a execução em paralelo com N = 1,2,4, 8 e 16 trabalhadores paralelos e pode ser implementado em C ou Python com recurso á plataforma open-mpi (que implementa operações distribuídas sobre arrays – *pode encontrar exemplos ilustrativos no moodle do projeto*) ou em Bash (gestão/sincronização dos processos) + C/Python (implementação das partes do algoritmo).

O programa recebe como argumento: o número de processos paralelos N; o número de clusters K; e o nome do ficheiro dos dados. Os formatos do ficheiro de entrada a suportar, são JSON, CSV e H5DF, contendo uma lista de pontos em $\mathbb{Z}^D$, i.e. as componentes dos pontos são inteiros e D é a dimensão do espaço. (*no separador do projeto no moodle pode encontrar alguns ficheiros de dados, bem como programas ilustrativos da sua leitura*). As saídas (atribuição dos clusters e centroides) devem ser gravadas no formato CSV.

É fornecida no moodle uma implementação de referência série que lê e escreve em CSV, sem a utilização de qualquer biblioteca adicional necessitando apenas do **gcc** para ser compilada ou python3.8+ para ser executada. Contudo a utilização de bibliotecas como libhdf5-dev e libjansson-dev for C or json and h5py packages for Python é obrigatória.

O comando para executar o k-means é:

```
#k-means input_data K out_clusters out_centroids N
```

Além do K-means distribuído, deve ainda ser criado um script que prepara a execução do programa. Neste projeto vamos considerar um sistema ubuntu acabado de instalar como a base de onde parte o script, que deve garantir que todas a dependências necessárias são instaladas e compila o programa e prepara os dados, entre outras operações que podem ser necessárias.

(Será distribuído um container nos próximos dias onde poderá verificar este script.)

Grupos e entrega

O trabalho é para realizar em grupos de 2 a 4 alunos e a entrega será no dia 4 de Julho no moodle.

Deve ser entregue num único zip:

- Código fonte C/Python
- Scripts (Bash)
- Relatório com identificação dos membros do grupo e das opções apresentadas. (Max. 5 páginas tamanho de letra > 10)
    - O relatório deve reportar e explicar a evolução do *speedup* com N (1…16)

Podem ser usados todos e quaisquer recursos para a resolução do problema (incluindo ferramentas de IA), contudo a incapacidade de justificar e explicar os conteúdos entregues (qualquer linha do código fonte, relatório ou apresentação) implica a sua não avaliação.

Fundamentos de Computação de Alto Desempenho

UNIVERSIDADE DE ÉVORA
ESCOLA DE CIÊNCIAS E TECNOLOGIA
DEPARTAMENTO DE INFORMÁTICA
Nuno Lourenço
23/24

Avaliação

A avaliação do trabalho é feita tendo em conta a qualidade do programa desenvolvido bem como o relatório entregue, bem com uma apresentação (pitch) de 3 minutos com participação obrigatória de todos os elementos do grupo, que será seguida de uma discussão. A apresentação e discussão ocorrerá remotamente em horário a combinar com os grupos.

Bónus e Penalizações

- +1 para melhor speedup com N = 4/8 *
- +1 para melhor speedup com N = 4/8 **

- Executa sem erros mas produz resultados errados: nota máxima 15
- Não executa ou executa com erros: nota máxima 12
- Sem execução paralela: nota máxima **0**

* Speedup médio medido pelo docente em relação á implementação fornecida no início do trabalho com um subconjunto dos ficheiros de dados fornecidos.

** Speedup médio medido pelo docente em relação á execução com N = 1 com um subconjunto dos ficheiros de dados fornecidos.