

Projeto do algoritmo

ALUNOS:

Daniel Sá Barretto Prado Garcia 103734344

Felipe Guilermmo Santuche Moleiro 10724010

Laura Alves de Jesus 10801180

Tiago Marino Silva 10734748

PCAM

Particionamento

A matriz de ordem $N \times M$ é particionada em M colunas, em que para cada coluna existirão K métricas a serem calculadas, ou seja, teremos $M \times K$ tarefas. Cada tarefa possui uma coluna e uma função de métrica da mesma para calcular. O cálculo de cada métrica em cada coluna pode ser feito em paralelo.

Comunicação

As colunas são independentes entre si. Ocorrem algumas dependências sobre métricas de uma mesma coluna. São elas:

- Variância depende da Média Aritmética
- Desvio Padrão depende da Variância
- Coeficiente de variação depende da Média Aritmética e do Desvio Padrão
- Mediana e Moda precisam da coluna ordenada

Aglomerção

Será feita uma thread para cada núcleo. Nela serão criadas tasks para cada métrica e suas dependências, cada thread executa uma próxima tarefa conforme termina de executar a prévia. Na nossa aglomeração cada task se responsabiliza por calcular uma métrica por coluna, mas as tasks podem ser feitas em paralelo. As tasks a serem feitas por colunas são:

- Task para cálculo da média aritmética
- Task para cálculo da média harmônica
- Task para ordenação da coluna
- Task para cálculo da mediana
- Task para cálculo da moda
- Task para cálculo da variância
- Task para cálculo do desvio padrão
- Task para cálculo do coeficiente de variação

Essas tasks são colocadas em um task pool e para serem atribuídas às threads conforme estas ficam livres.

Mapeamento

Há um número de threads T , assim, cada coluna enfileira suas tasks em um task pool. As threads vão executando essas tasks conforme ficam ociosas. O número de thread depende da máquina, portanto pode acontecer de cada núcleo processar apenas uma thread ou precisar processar mais de uma thread (se o número de threads for superior ao de núcleos), esse escalonamento será feito pelo SO que fará um escalonamento dinâmico para a execução das threads nos núcleos.