Distributed computation of linear algebra operations

Distributed systems and networks laboratory

Gabriele Aloisio [503264] Samuel Giacomo Raffa [matricola]

2023

Università degli studi di Messina

Table of Contents

Matrix operations

Matrix operations are fundamental in numerous scientific and computational domains, serving as the building blocks for various applications. However, traditional sequential computation on a single machine often becomes a bottleneck, limiting the speed and scalability of matrix calculations.

Matrix operations

To address this problem, we are going to use an open-source distributed computing framework called Ray.

Overview of distributed systems and networks

A distributed system consists of multiple interconnected computers that collaborate and coordinate their activities to achieve a common goal. These systems are designed to tackle tasks that cannot be efficiently computed by a single machine.

Networks serve as the backbone of distributed systems, enabling communication among the connected nodes. Network infrastructures enable coordination, data sharing, and synchronization across distributed systems, regardless of their physical locations.

Issues with distributed systems

Distributed systems also present unique challenges:

- Data consistency
- Fault tolerance
- Load balancing
- Network latency

Solving challenges with Ray

When it comes to matrix operation computation, using Ray for distributed computation offers several advantages over performing the operations on a single machine:

- Faster execution
- Scalability
- Fault tolerance
- Resource utilization

Faster execution

With Ray you can use multiple machines and CPUs to compute operations. This significantly reduces the computation time compared to a single machine. Each machine can work on a subset of the matrix data, completing the calculations in parallel. This distributed approach can lead to substantial speedups.

Scalability

As the size of the matrices grows, a single machine may struggle to handle the computational demands due to memory limitations or processing power constraints. Ray allows you to scale horizontally by adding more machines to the distributed setup.

Fault tolerance

Ray offers fault tolerance mechanisms that ensure the continuity of computation even in the presence of failures. If a machine participating in the distributed computation fails, Ray can automatically redistribute the workload to other available machines.

Resource utilization

With Ray, each machine contributes its processing power and memory capacity to the overall computation. This efficient utilization of resources allows you to make the most of the available hardware infrastructure, compared to a single machine that may be underutilized.

Table of Contents

Explanation of common linear algebra operations

Challenges in performing these operations on large datasets


STUDIORUM UNIVERSITAS
MESSANAE

Table of Contents

Overview of the Ray library and its capabilities

Key features and advantages of using Ray for distributed computation

Table of Contents

List of operation

We will showcase the following operations:

- Product
- Determinant
- Inverse
- Rank

The Matrix and RayMatrix classes

The Matrix class

The RayMatrix class