CSE 633
Parallel Solution of
Marginal Fisher Analysis

Suxin Guo

Nov 22, 2011

Fisher Discriminant Analysis

- **Fisher Discriminant Analysis aims to maximize between-class separability and minimize within-class variability.**

$$J(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{S}_B \mathbf{w}}{\mathbf{w}^T \mathbf{S}_W \mathbf{w}}$$

where

$$\mathbf{S}_B = (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T$$

$$\mathbf{S}_W = \sum_{i=1,2} \sum_{j=1}^{n_i} (\mathbf{x}_j^i - \mathbf{m}_i)(\mathbf{x}_j^i - \mathbf{m}_i)^T$$

Fisher Discriminant Analysis

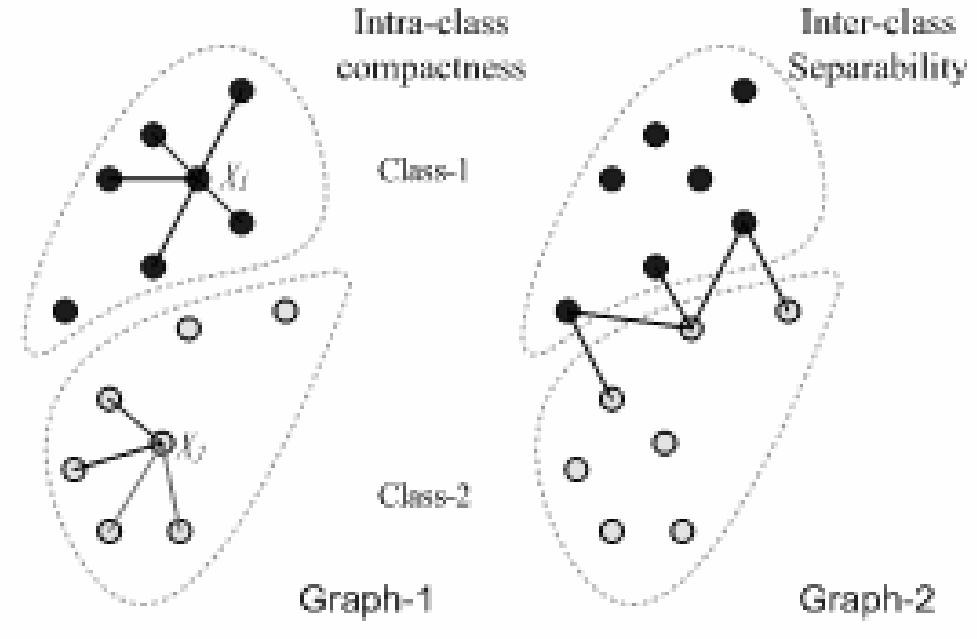
- The objective of FDA is to find a projection vector w such that J(w) is a maximum. The solution for such w can be obtained by differentiating J(w) with respect to w yielding:

$$\mathbf{w} = \mathbf{S}_W^{-1}(\mathbf{m}_1 - \mathbf{m}_2)$$

Marginal Fisher Analysis

- FLD assumes that the data of each class is of a Gaussian distribution, which is not always satisfied in real-world problems.
- Develop new criteria that characterize intra-class compactness and inter-class separability.
  - Intra-class compactness is represented as the sum of distances between each point and its k1 -nearest neighbors of the same class
  - The separability of different classes is characterized as the sum of distances between the margin points and their neighboring points of different classes.

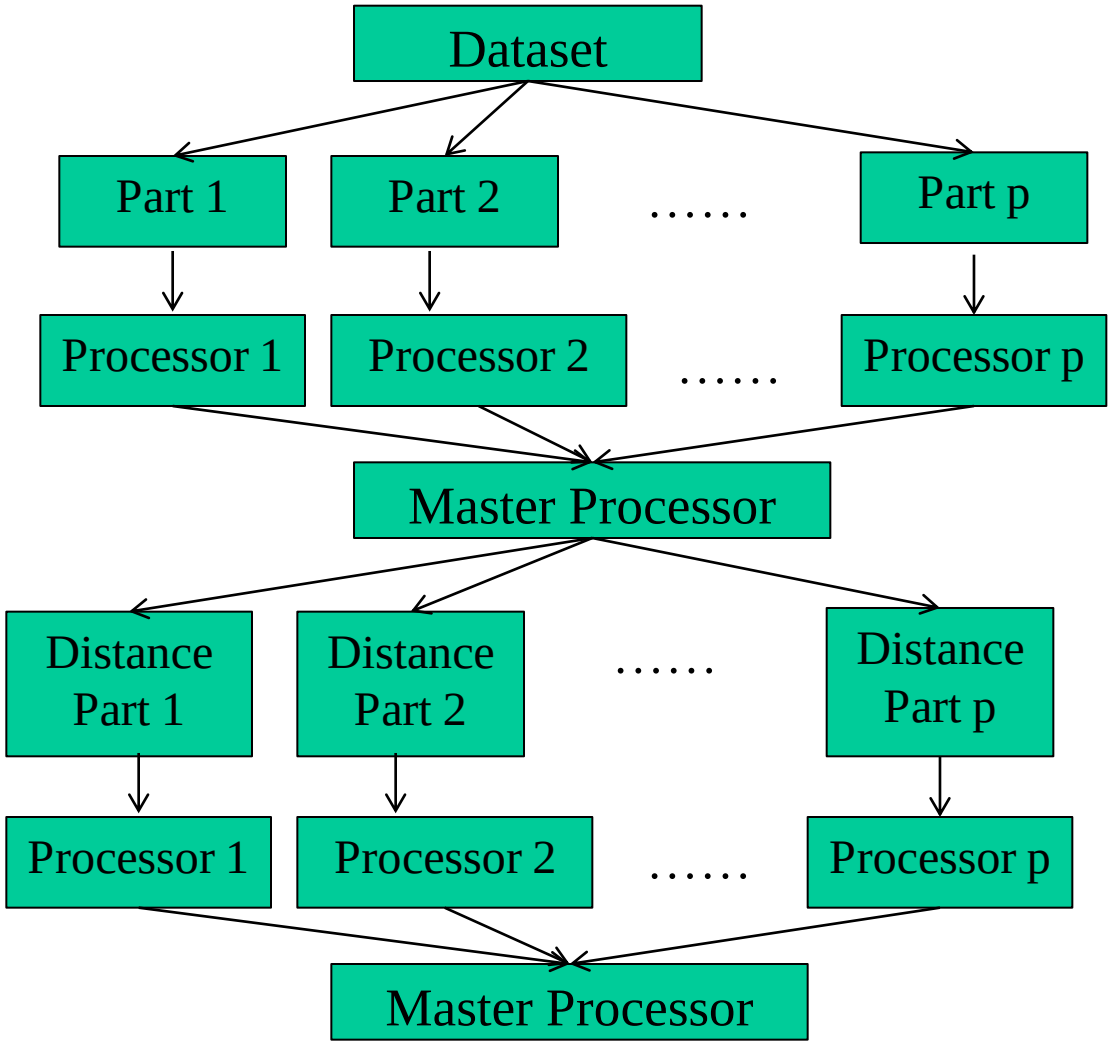
Marginal Fisher Analysis



To Implement..

- We need to compute the distances between every pair of nodes -------$O(n^2)$
- Then we have to sort the distances: For typical sorting algorithms, good behavior is O(NlogN) and bad behavior is $O(N^2)$
- n is the number of total points and N is the number of distances between the points
- $O(n^2 \log n) – O(n^4)$
- When n is large, it can be very time consuming!
- We need the parallel solution

Parallel Solution



Split the dataset into p parts, each of which has n/p points. p is the number of processors

Each processor computes the distances between the data points in its own part

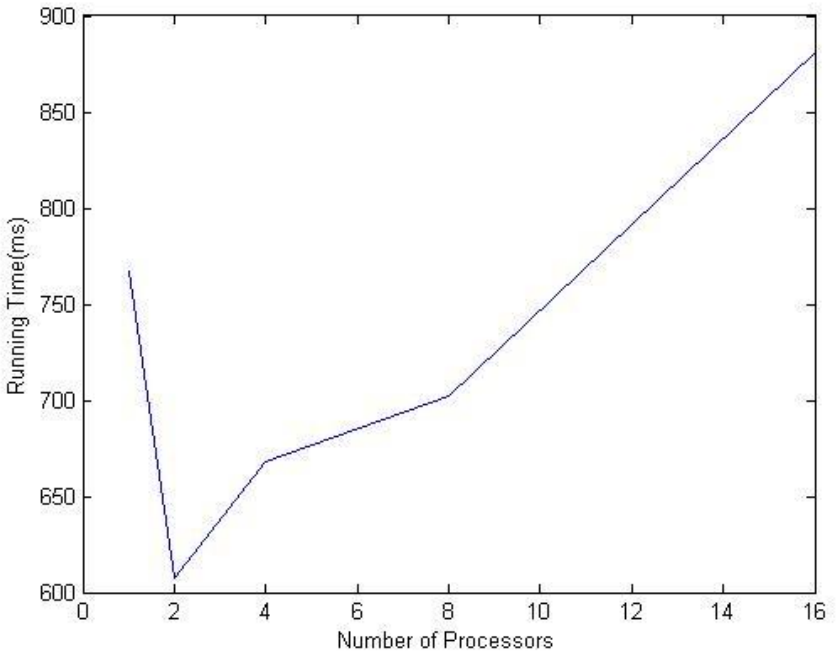
The master processor collects all the distances and do the remaining computations to get all of the distances

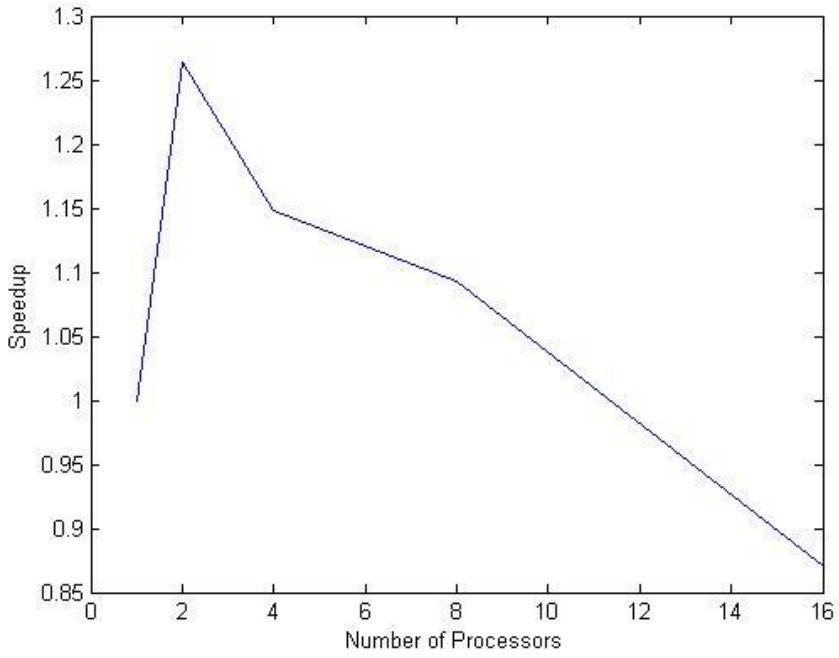
Split the distances into p parts and give them to the p processors respectively to sort

Each processor sorts its own part and give the results to the master. The master then merge the results and perform following computations.

Results

Total running time and speedup of the master node:



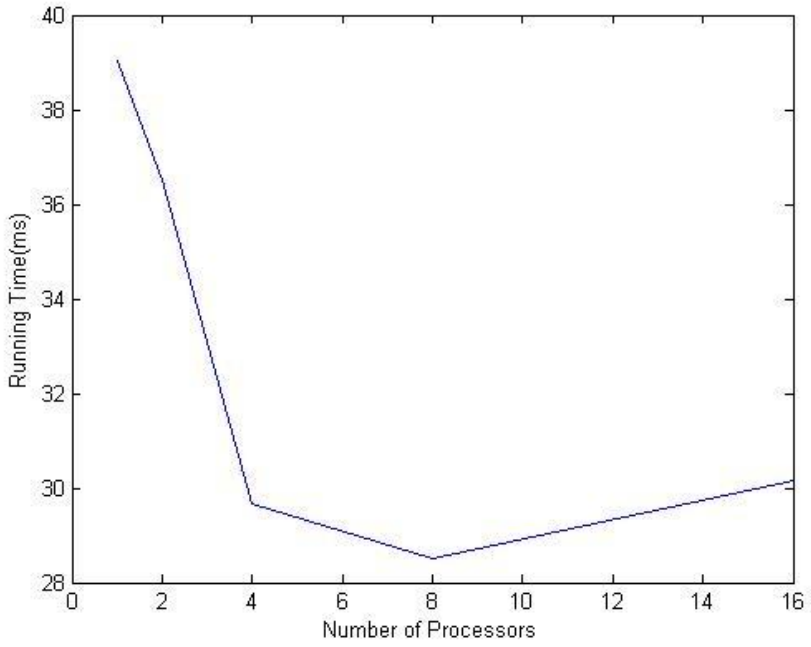


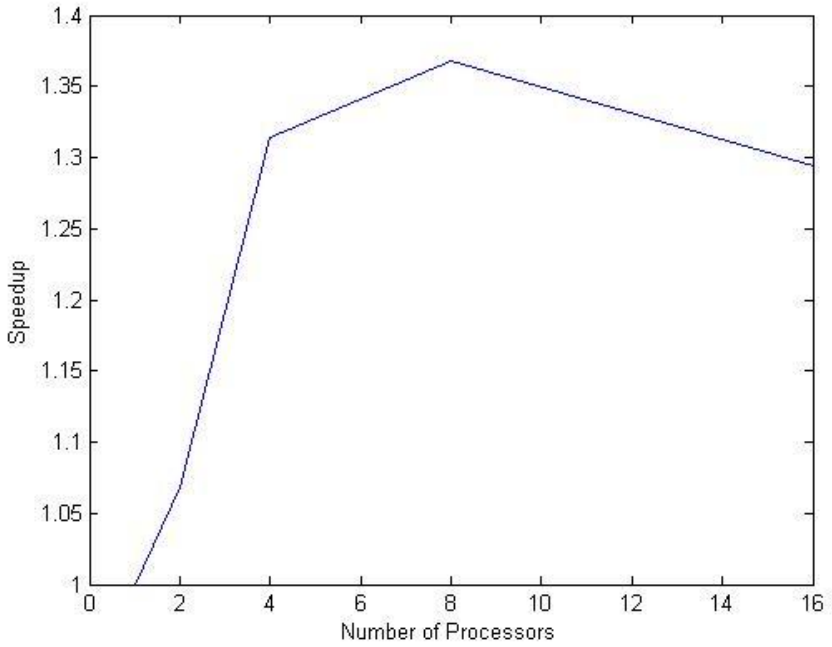
Number of data samples: 800

Number of features: 4

Results

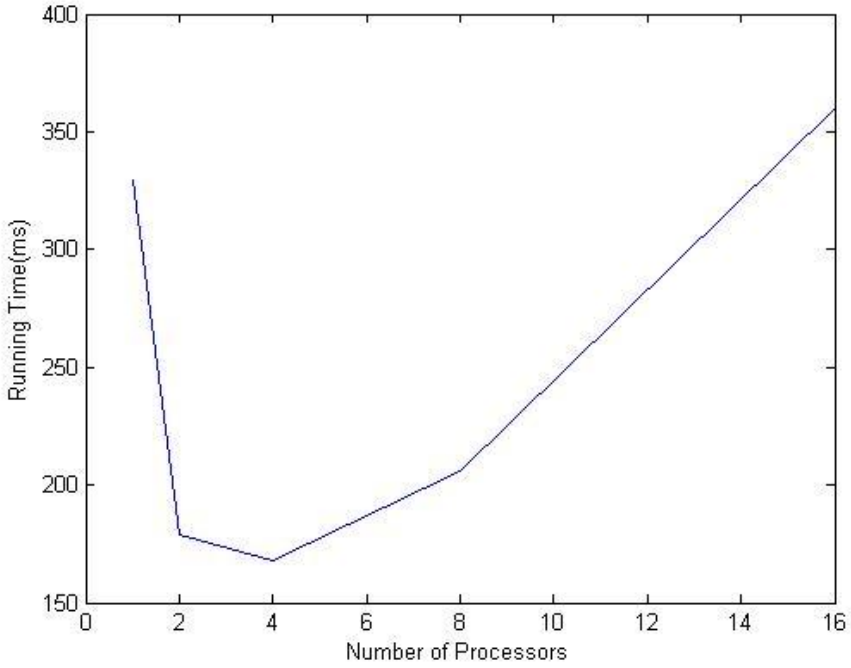
Running time and speedup for distance computing of the master node:

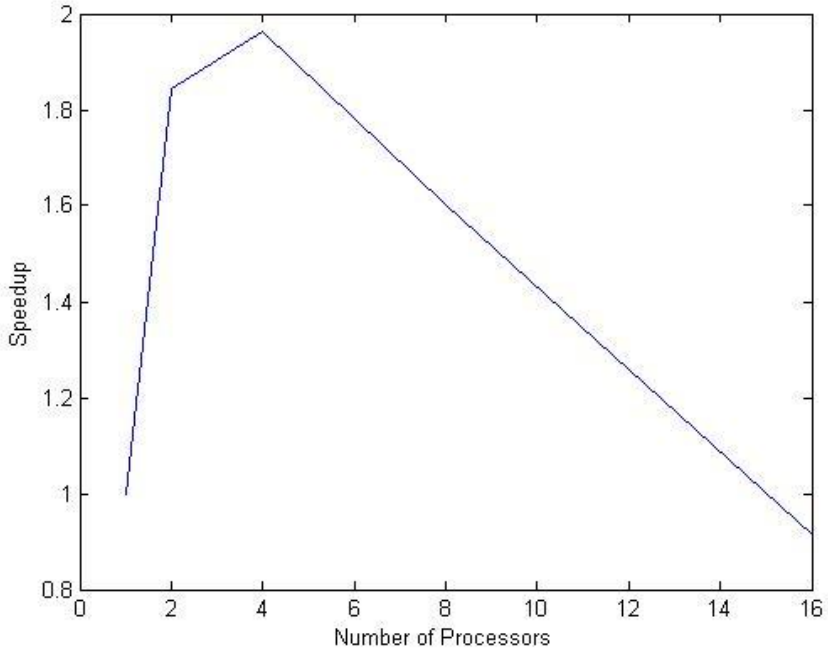




Results

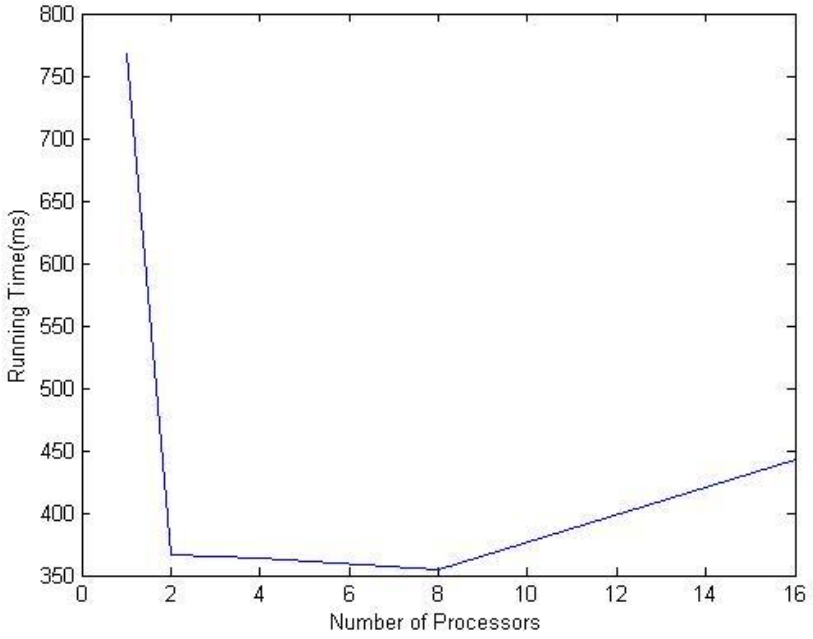
Running time and speedup for sorting of the master node:

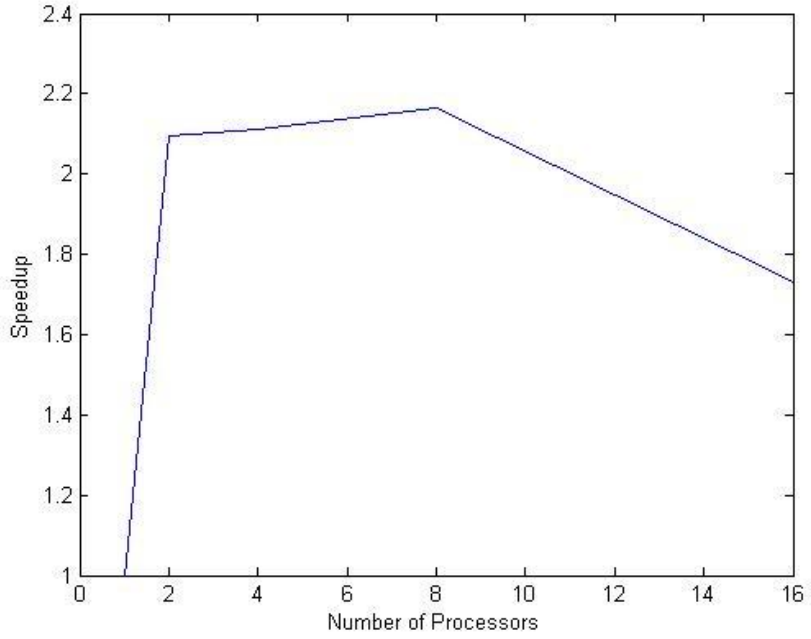




Results

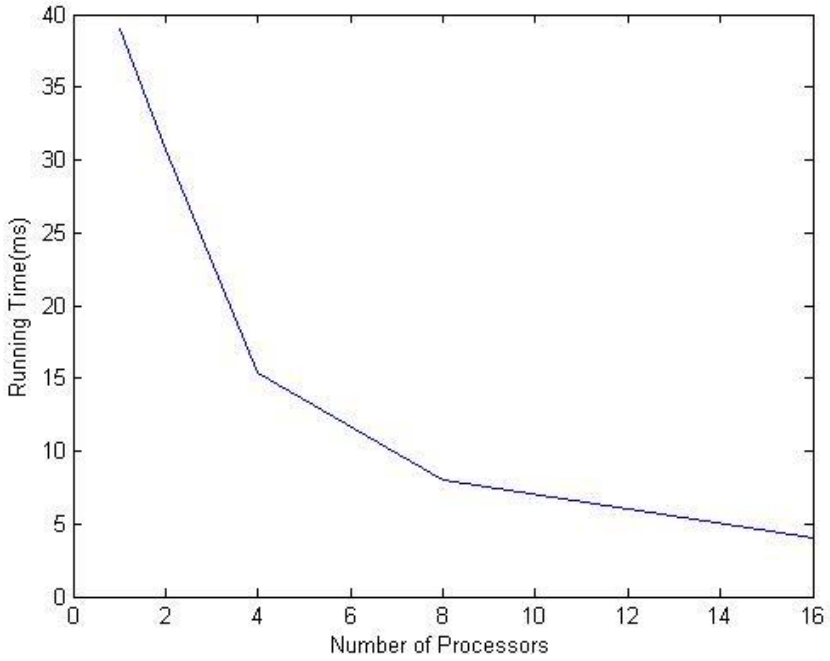
Total running time and speedup of the slave nodes:

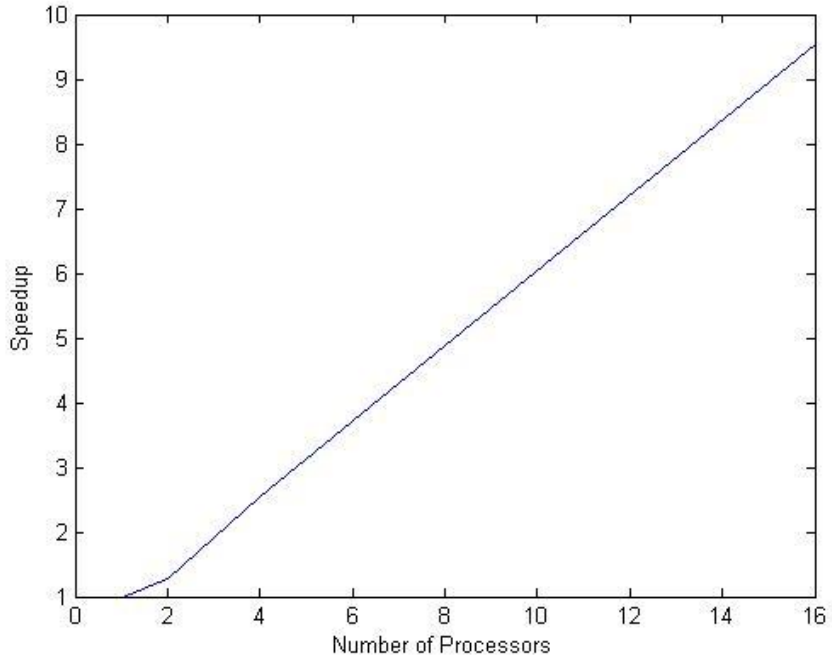




Results

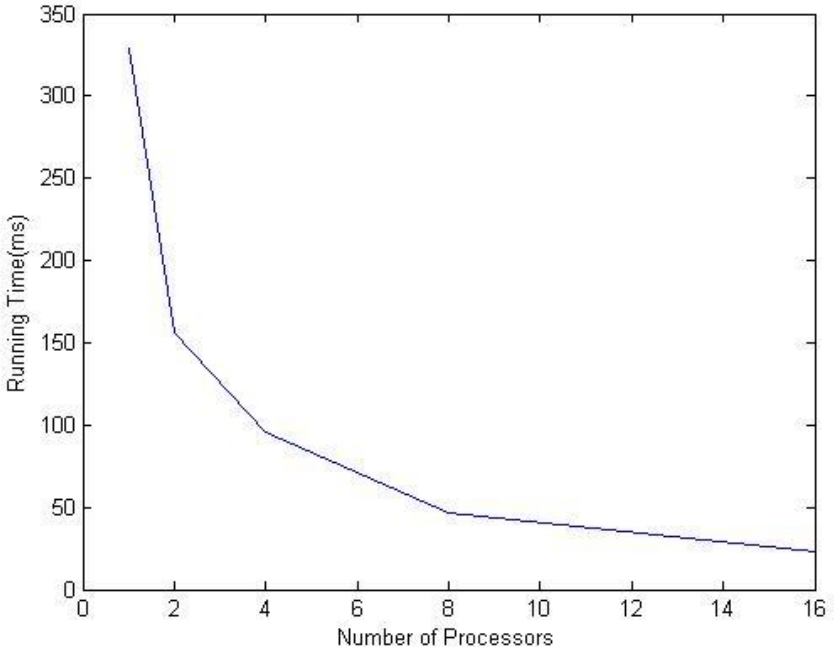
Running time and speedup for distance computing of the slave nodes:

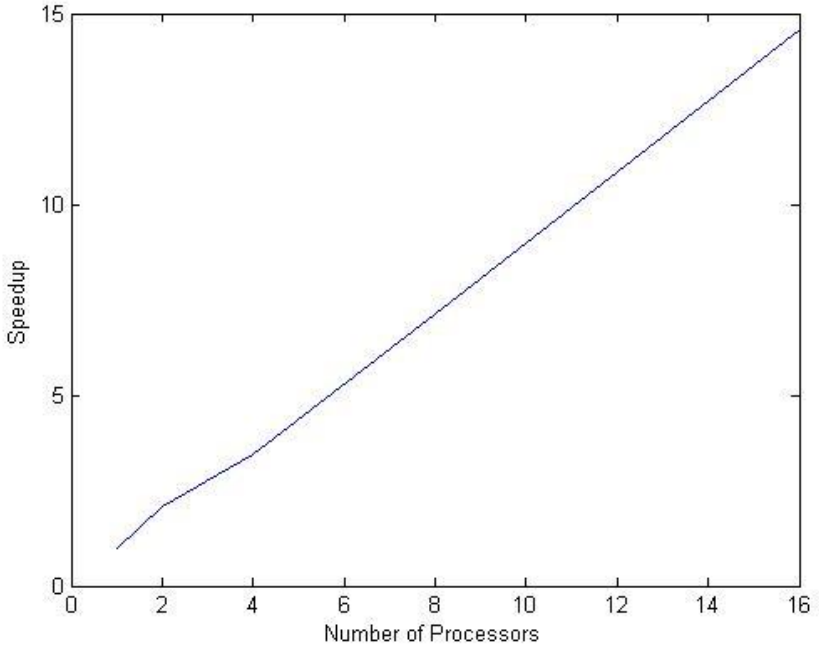




Results

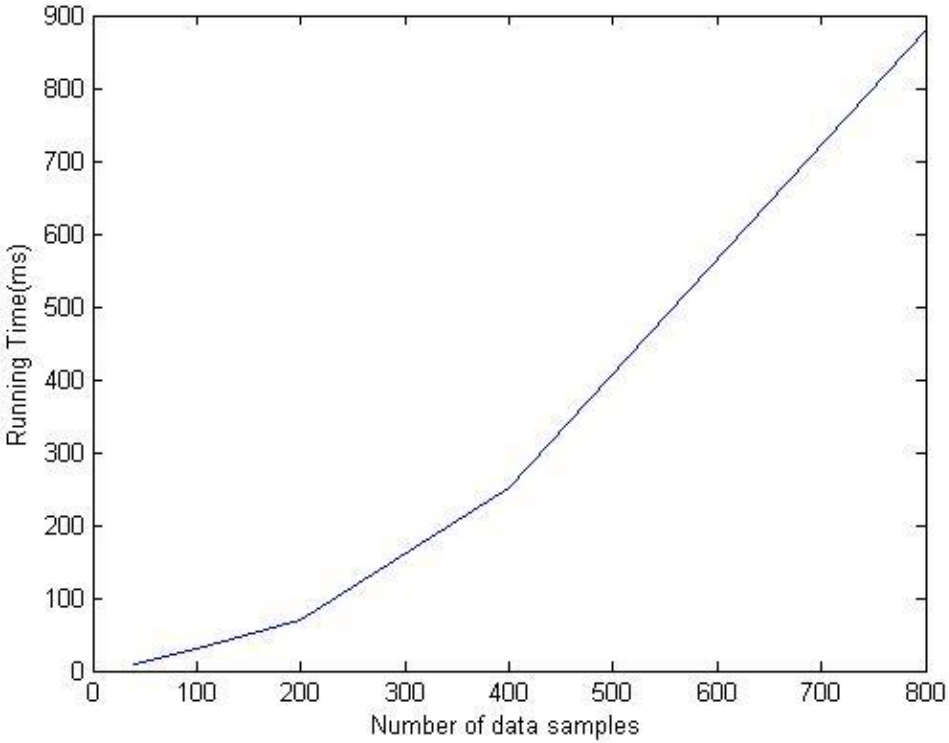
Running time and speedup for sorting of the slave nodes:





Results

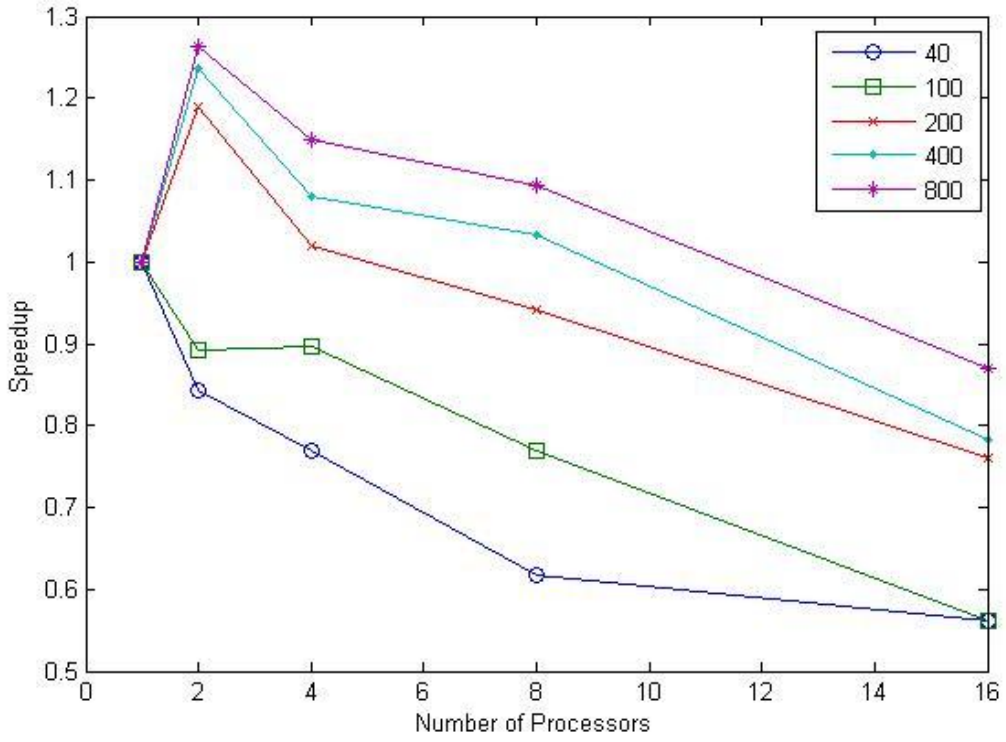
Running time increases when the size of dataset increases:



Number of processors: 16

Results

Speedup changes when the size of dataset increases:



Questions?

Thank you!