PARALLELIZED LOGISTIC REGRESSION USING GRADIENT DESCENT

Anuja Wani

CSE 633- Parallel Algorithms

Guided By- Dr. Russ Miller

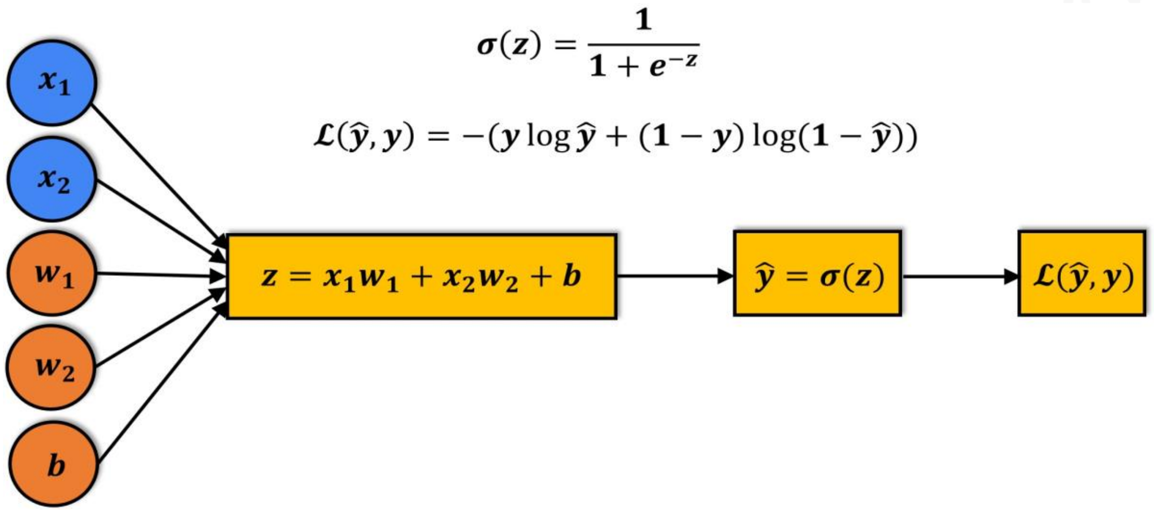
University at Buffalo The State University of New York

Contents

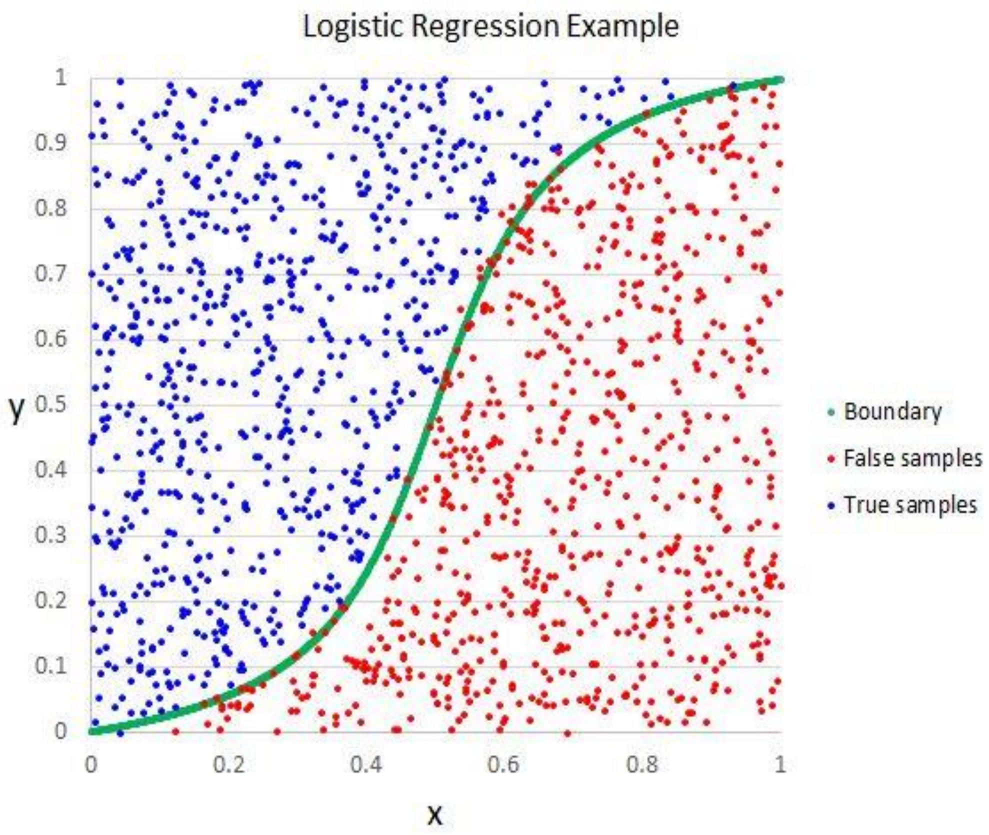
- Brief overview of Logistic Regression
- Gradient descent
- Sequential algorithm
- Applications and why should we use parallelization?
- Parallel algorithm
- Example
- Results
- Conclusion
- Future Work
- References

Logistic Regression

- Statistical Model that predicts the possibility of an event occurring
- Often used for classification and prediction
- Used when dependent variable is categorical



- Output can be:
  - Binary – only two possible outcomes
  - Multinomial - three or more possible outcomes without ordering
  - Ordinal - three or more possible outcomes with ordering
- A threshold called decision boundary is set to predict which class the given data lies



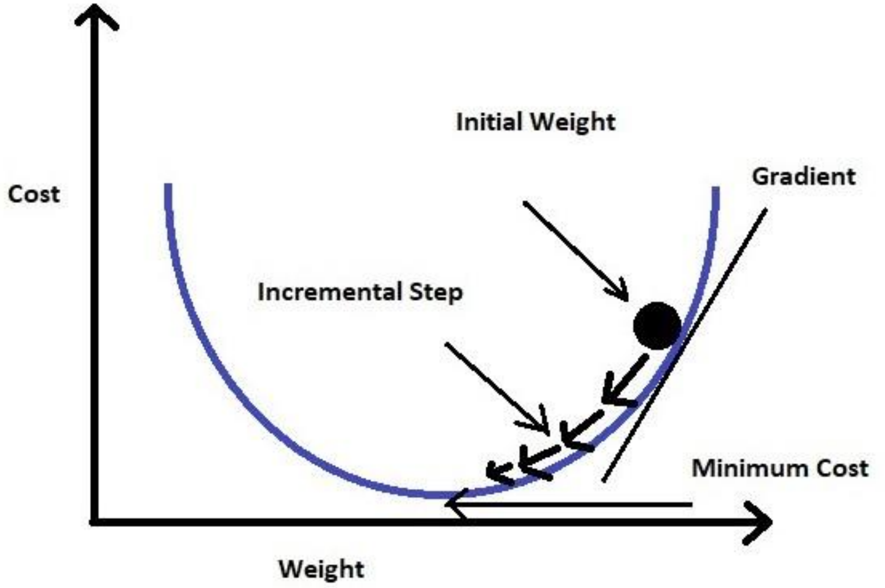
Gradient Descent

- Iterative optimization algorithm to find the minimum of a differentiable function
- Used to minimize loss function to improvise logistic regression prediction
- Applied on the cost function in logistic regression to find optimal solution
- The gradient descent algorithm can be summarized as:

$$\theta_j = \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta)$$

Sequential Algorithm

1. Select any random point on the graph
2. Compute derivate which will point to the minima
3. Multiply resultant to learning rate
4. Subtract result from the old value to get new value
5. Do this for every iteration
6. We perform these action till we reach the global minimum - lowest loss possible in prediction
7. Perform validation and predict the values



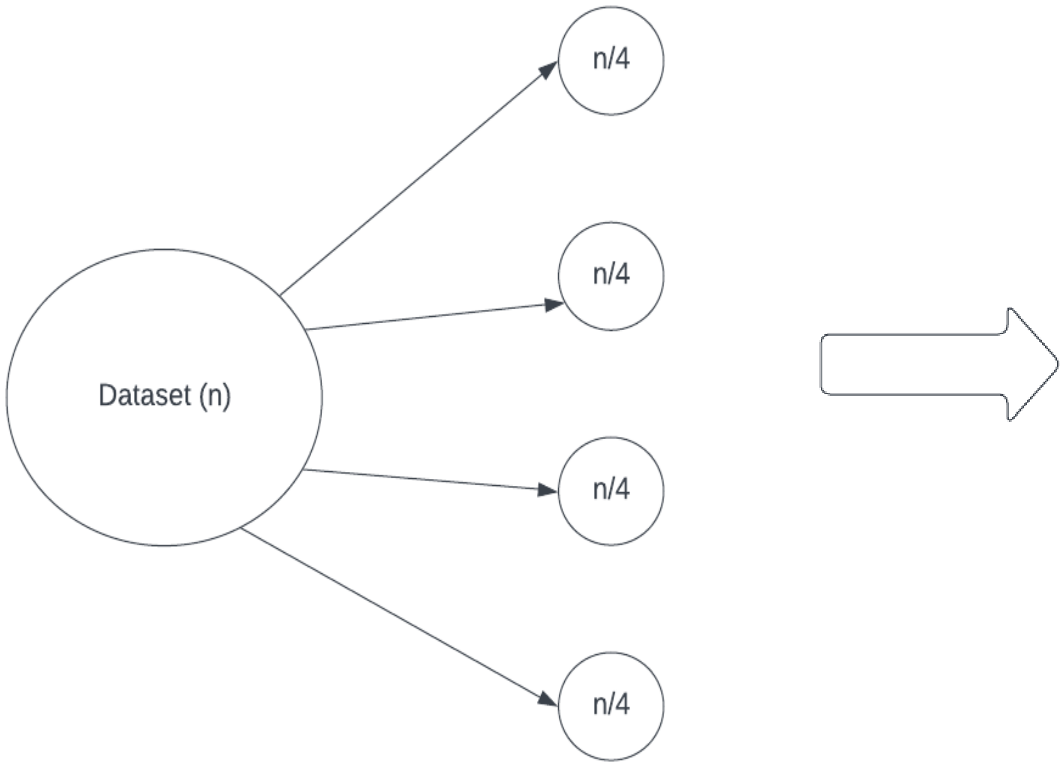
Applications and why should we use parallelization?

- Logistic regression has a wide variety of applications
- Used in credit card fraud detection, medical research, insurance policy approval, etc.
- Many such applications involve huge amount of data which needs to be processed quickly and accurately
- Calculating gradient descent is a very heavy computation task
- Parallelizing the task improves the processing speed significantly

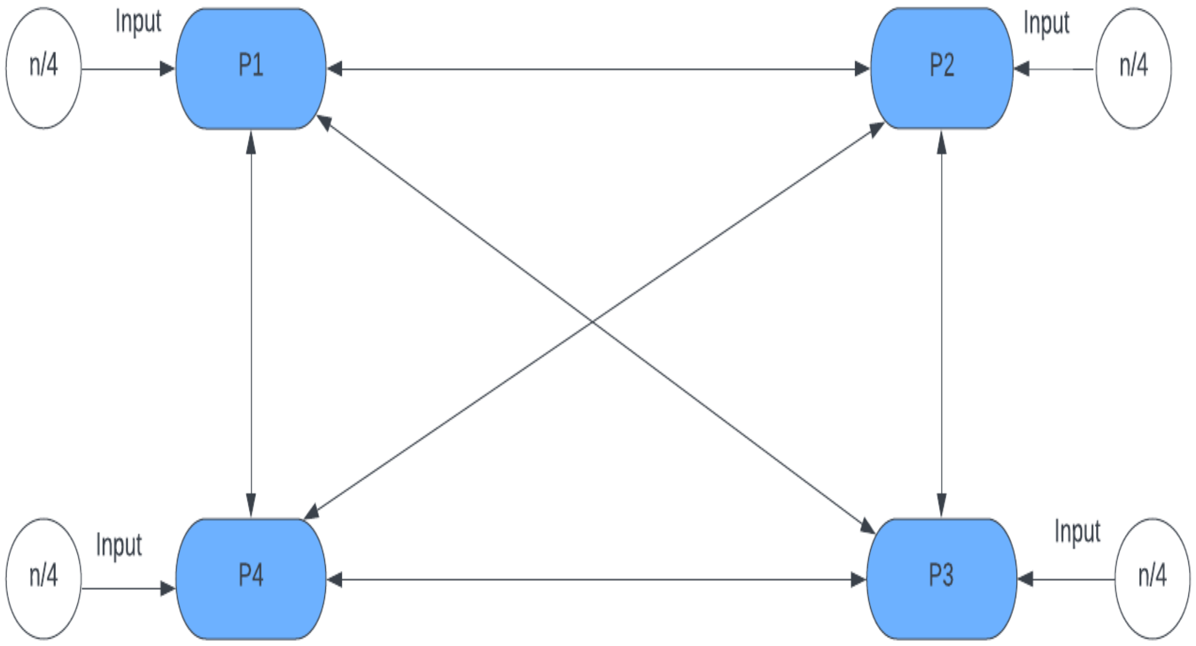
Parallel Algorithm

- We focus on improvising the gradient descent calculation in our parallel implementation
- The data is distributed uniformly between all processors
- Each processor is initialized with random weights
- Each processor calculates the gradient descent on its data
- Communicates it to the other processors
- Each processor then updates its local weights based on its received values
- Repeat this until the number of epochs are exhausted or gradient values converge
- Perform validation and predict the values

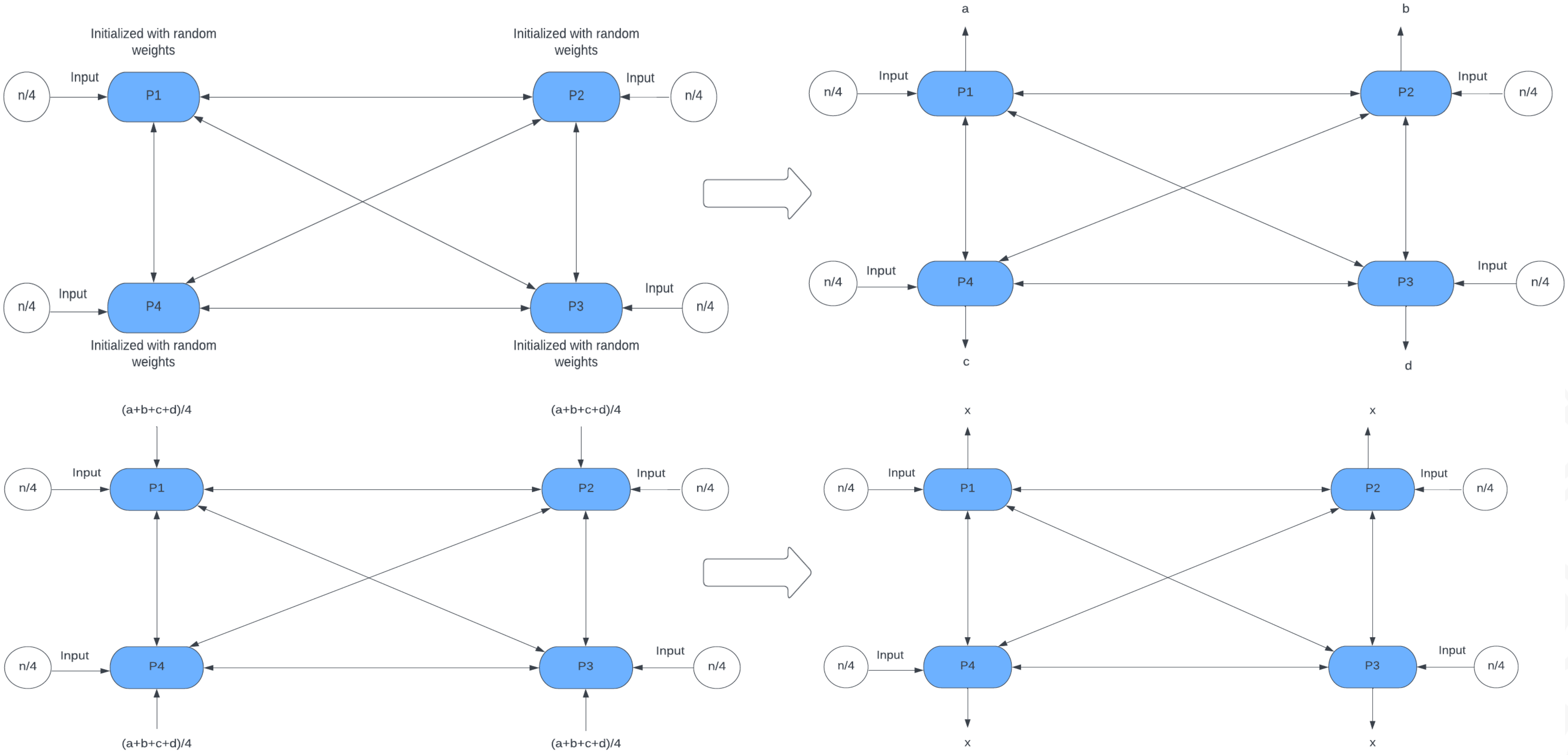
Example



Dataset (n)
n/4
n/4
n/4
n/4



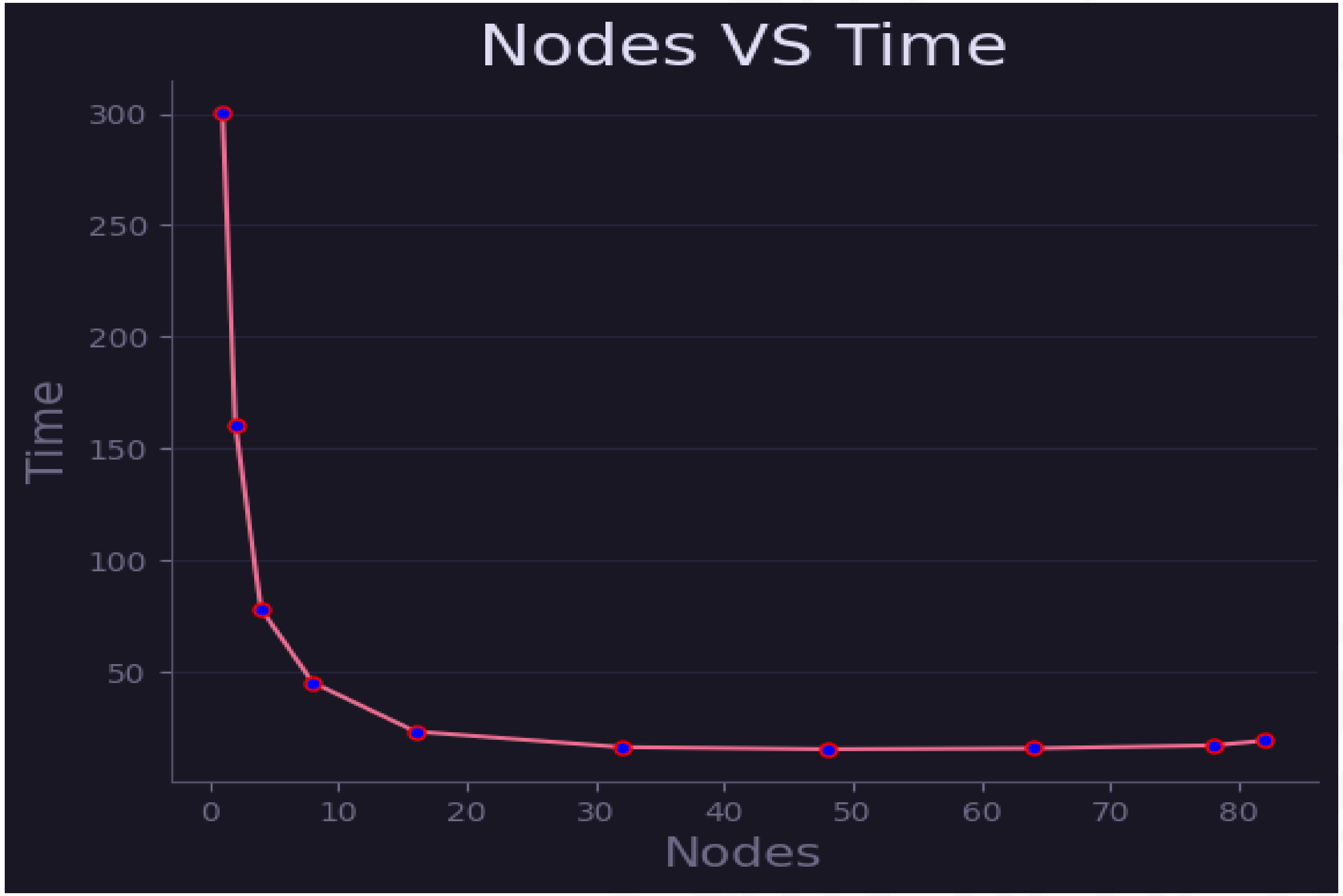
n/4
Input
P1
P2
Input
n/4
n/4
Input
P4
P3
Input
n/4

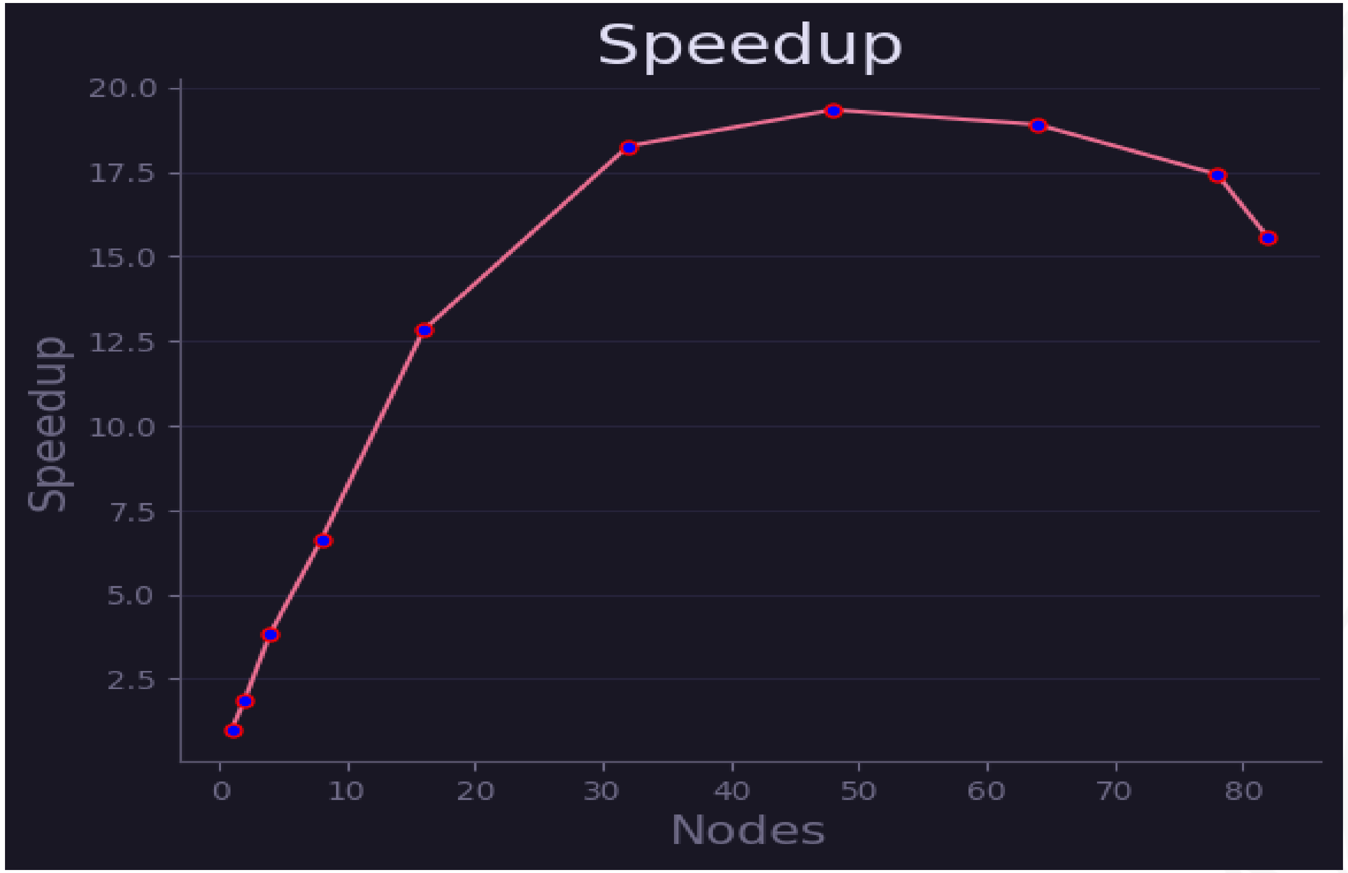


Initialized with random weights
Input
n/4
P1
P2
P3
P4
a
b
c
d
(a+b+c+d)/4
x

Results

Nodes	Time
1	300.5
2	160.9
4	78.4
8	45.5
16	23.4
32	16.4
48	15.5
64	15.9
78	17.2
82	19.3




Speedup
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
Speedup
0
10
20
30
40
50
60
70
80
Nodes

Next Steps…(Midterm)

- Once all processors receive the gradient value, they should update and calculate the gradient of the cost function again. ✓
- Scale it to more processors ✓
- Improve the efficiency of the model ✓

Conclusion

- As the number of processors increase, the time for gradient descent calculation also decreases
- However, after a certain number of nodes, the time required for gradient descent calculation increases with increase in number of processors
- This is due to the communication overhead overshadowing the increase in efficiency

Future Work

- Try to implement this for more nodes
- Try multiple cores per node
- Current efficiency ~84% for parallelized approach, and for serial implementation ~92%, try to reduce this gap

References

- https://towardsdatascience.com/logistic-regression-detailed-overview-46c4da4303bc
- https://www.baeldung.com/cs/gradient-descent-logistic-regression
- http://courses.cms.caltech.edu/cs179/Old/2015_lectures/cs179_2015_lec16.pdf
- https://ieeexplore.ieee.org/document/6691743
- https://www.cs.cmu.edu/~daria/papers/fslr.pdf
- https://penghaoruo.com/res/Evaluating_Parallel_Logistic_Regression_Models.pdf
- https://freakonometrics.hypotheses.org/53283

THANK YOU!

QUESTIONS?