Regularized Logistic Regression using Gradient Descent

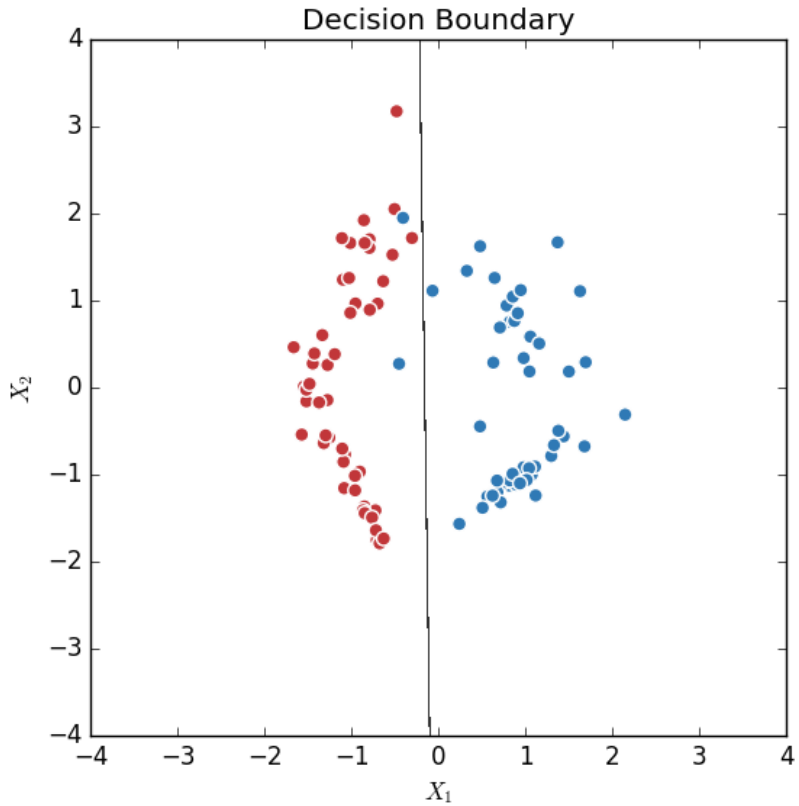
Abhishek, Mirza Galib A H Baig

Indian Institute of Technology Guwahati

November 19, 2015

Regularized Logistic Regression: Intuition

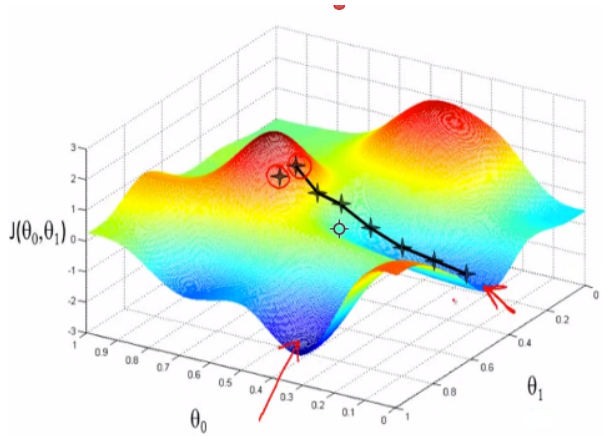
- We want to find the $Pr(Y = 1|x)$ as a function of x i.e $\sigma(\theta^T x)$.
- Unknown parameter (θ) in the function is learned by minimizing $J(\theta) = -1/m \sum_{n=1}^{m} (y^i log(\sigma(\theta^T, X^i)) + (1 - y^i) log(1 - \sigma(\theta^T, X^i)))$ which is derived by maximum likelihood estimation.



Serial Gradient Descent and Bottlenecks in parallelization

Algorithm 1 Pseudo-code for Gradient Decent

```
data_set ← FileRead
for i = 1 to I do
    for j = 1 to θ do
        sum = 0, ERROR = []
        for rows to data_set do
            sum += (hypothesis(row, θ) − label) ∗ row_j
        end for
        ERROR_j = α ∗ sum/number_of_rows
    end for
    θ = θ − ERROR
end for
```



Bottlenecks

- Initial File read is sequential
- Wait for combining result

Design of Parallel Gradient Descent

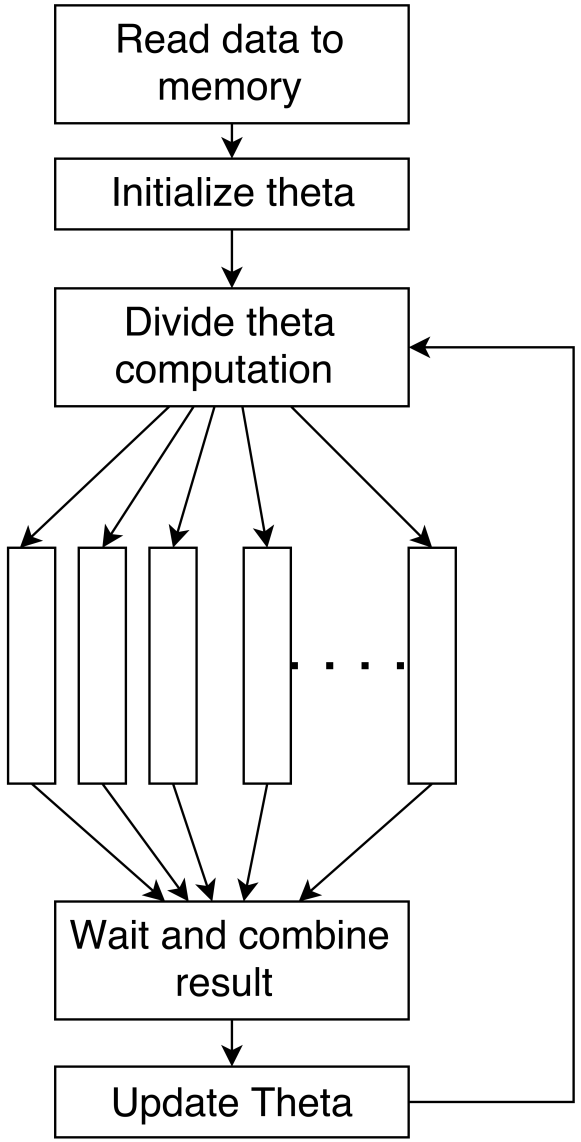


Figure: Parallel Gradient Descent.

Bottlenecks in Distributed implementation

- Data should be divided among different nodes.
- Communication between different nodes.
- Wait till all nodes have computed before executing next iteration.

Design of Distributed Gradient Descent

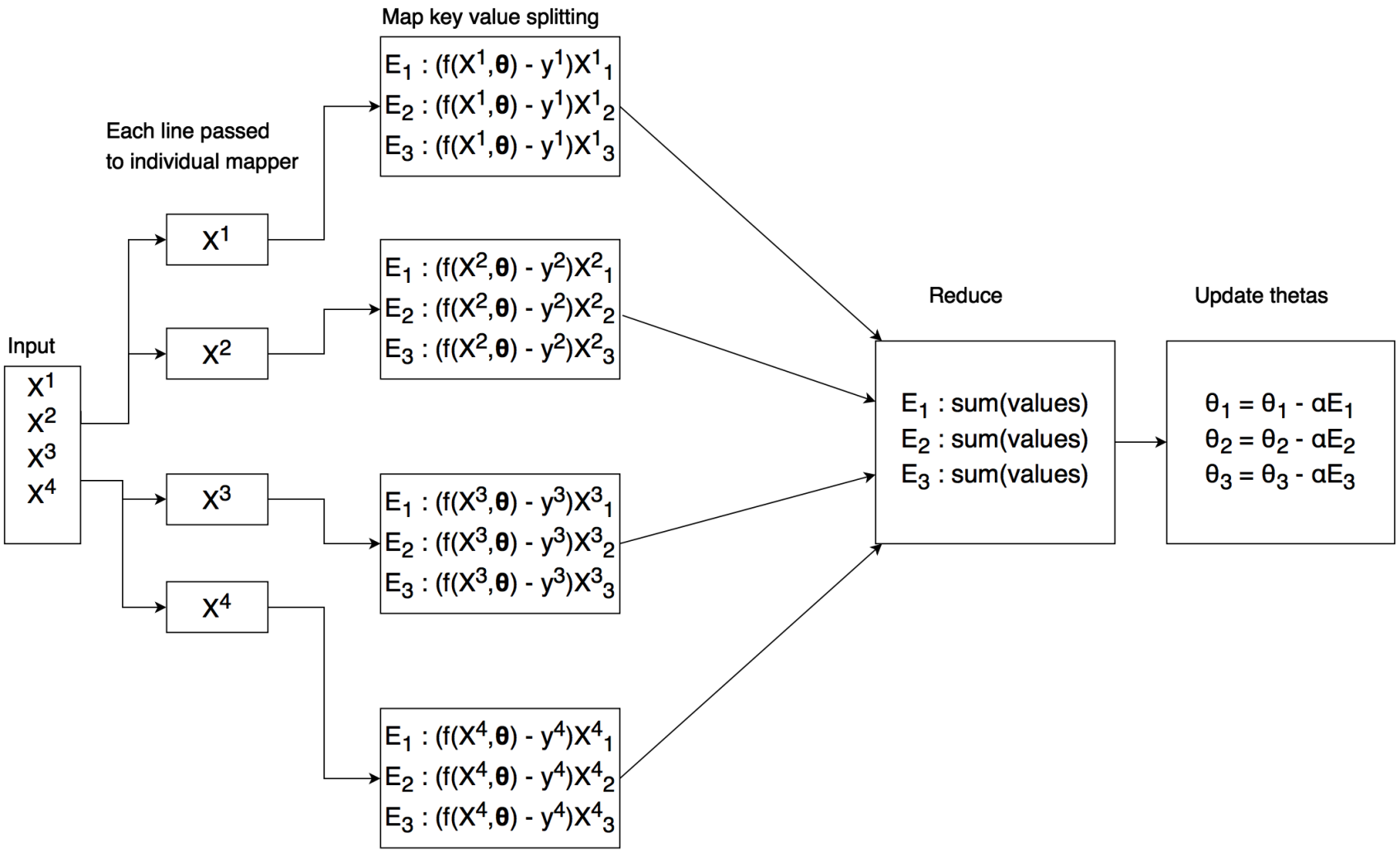


Figure: Map Reduce Regularized Logistic Regression.