MS&E 317/CS 263: Algorithms for Modern Data Models, Spring 2014
`http://msande317.stanford.edu`.
Instructors: Ashish Goel and Reza Zadeh, Stanford University.

Lecture 13, 5/14/2014. Scribed by James Gippetti.

13.1 Overview of Optimization

We saw 2 ML algorithms, SGD and ADMM, which are both good if you have lots of data as long as seperable. By seperable we mean:

$$f(x) = \sum_{i=1}^{N} f_i(x)$$

On a single machine:

- Gradient Descent (GD) Update: $x^{k+1} = x^k - \alpha \sum_{i=1}^{N} \Delta f_i(x^k)$
- Stochastic Gradient Descent (SGD) Update: $x^{k+1} = x^k - \alpha \Delta f_j(x^k)$, which assumes uniform sampling of f_i's
- GD for convergence to ε accuracy: $N \log(1/\varepsilon))$
- SGD for convergence to ε accuracy: $1/\varepsilon$ - in expectation, need to run through a few times

13.2 Optimization Algorithms in Parallel

Parallell SGD:

1. Shuffle data
2. Split between machines
3. Run SGD Locally
4. Average the x^k's with Allreduce

ADMM:

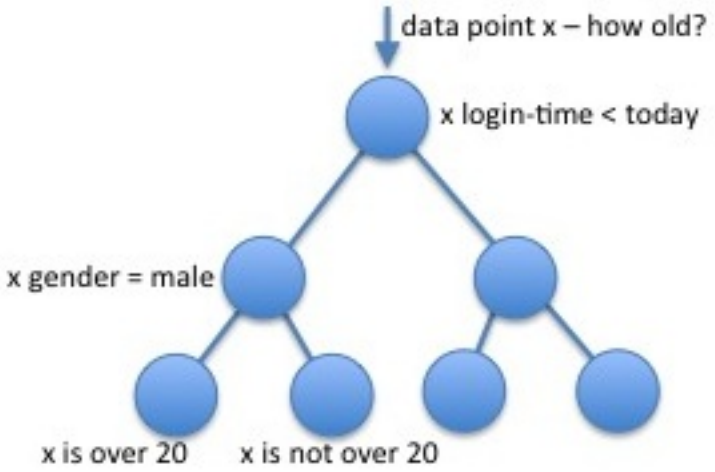
1. Local ADMM interation (abitrary optimization primitive - potentially SGD)
2. Allreduce Average
3. Repeat as we need to converge (slightly more iterations but not too much talking - only in the Allreduce)

13.3 Consensus SVM

Please refer to the slides for the example we went through in class.

13.4 Decision Trees

Use training dataset to build a tree of this form:



$w_{model}^T x$ gives estimate of age and the **purity** of a node is how close it is to having all of 1 classifier

Use local optimation to figure out:

1. What feature to split on?
2. What threshold to use?

Stopping Criterion (for building out tree):

1. Totally pure nodes
2. Number of points below threshold

Once you run and build your decision tree model on training dataset, you have validation set so if you overfit you can go and prune the tree

PLANET (see paper posted for additional details):

1. Run distributed job per node
2. Until nodes become small enough to fit in memory, then run local decision tree learning

How to find best feature and threshold? - consider a sorted feature i and then binary search for the best threshold

13.5 Equidepth Histograms

Given single dimensional points (i.e. focus on one feature) not sorted, compute an equidepth histogram, which looks like a typical histogram with count on the y-axis and the feature values on the x-axis, bucketed out by split points. Only consider split points on the histogram.

On distributed system:

1. Sample each feature
2. Compute equidepth histogram
3. Use histogram spit points for that feature

13.6 Summary and Comparison Table

SVM/LR/LS	**DT(Decision Trees)**
Two Classes	Many Classes
Real Features	Real and Categorical
Simple Decision Boundaries	Highly Complex Decision Boundaries
Less Overfitting	Potential for Overfitting
Not Interpretable	Interpretable
Prove Optimality	Not Optimal Problem