INTRODUCTION

- Big data: not just many data points, also many features

A popular approach is feature selection:

$$\min_{x \in \mathbb{R}^d} \sum_{i=1}^{n} f(x^T a_i, b_i) + \lambda \|x\|_1$$

- Selects most important features, others set to zero
- Existing approaches limited:
  - Distributed algorithms require intensive communication
  - Computation per feature disproportionate to importance

OUR CONTRIBUTIONS

- Algorithms that aggressively prioritize features
  - Significantly reduces communication
  - Prioritizes resources in theoretically sound manner
  - Runs fast in distributed, multicore, and memory-limited settings
- Effective use of subproblems on feature subsets
  - Eliminates features guaranteed to be irrelevant
  - Discovers important features with high probability

EMPIRICAL RESULTS

- Blitz algorithm, sequential setting (1 CPU)

- QuickStep, Distributed Setting (16 nodes)

Distributed feature engineering

- Problem: predict stock volatility from financial report data
- Consider candidate features in parallel on worker nodes
- Solve subproblems on master, reducing runtimes considerably

CONCLUSIONS & FUTURE WORK

- Feature subsets effective for large-scale feature selection
- Can be used in distributed, sequential, or approximate settings
- In future, continue push to understand feature engineering
- Extend ideas to other important optimization problems