Mirtha Ayala Macías

56596

Convex Optimization for Big Data

El artículo trata de los avances recientes en algoritmos de optimización convexa para resolver problemas de grandes volúmenes de datos, enfocados en resolver los problemas computacionales, de espacio y de comunicaciones. Se describen técnicas de aproximación tales como métodos de primer orden y aleatorización así como el papel del cómputo en paralelo y distribuido.

El artículo menciona que la importancia de las formulaciones convexas y optimización ha aumentado en la última década debido al surgimiento de modelos exitosos de aprendizaje estadístico (ej. máquinas de soporte vectorial).

Menciona tres pilares para entender los algoritmos de optimización de Big data:

1. **Métodos de primer orden**: con estos métodos se obtiene una precisión numérica baja-media, utilizando información de primer orden del objetivo. Típicamente se realizan en cómputo distribuido y en paralelo y son robustos para realizar optimizaciones.

2. **Aleatorización**: Estas técnicas destacas sobre otras técnicas de aproximación para mejorar la escalabilidad de métodos de primer orden debido a que es posible controlar su comportamiento esperado. Ideas clave de este concepto son las actualizaciones parciales aleatorias de variables de optimización, reemplazo de gradiente determinístico entre otros.

3. **Cómputo distribuido y en paralelo**: Los métodos de primer orden proveen un marco de referencia para las tareas de optimización cómputo en paralelo. Estos pueden ser complementados incrementando niveles de escalabilidad, desde algoritmos paralelos síncronos idealizados con comunicación centralizada a algoritmos asíncronos.

A motivation for first-order methods

1. First-Order Methods for Smooth and Non-Smooth Convex Optimization

Esta sección describe métodos de primer orden dentro de este contexto, enfatizando algoritmos específicos con de convergencia global.

- **Smooth objectives**

Considera los casos en donde F consiste de una función diferenciable convexa ‘f’. Al realizar suposiciones simples acerca de ‘f’, podemos analizar el número de iteraciones que requiere realizar el método gradiente para alcanzar una solución precisa.

- **Composite objectives**

Se refiere al problema canónico compuesto, F se refiere a funciones ‘f’ diferenciables y una función convexa ‘g’ no suavizada. Los métodos de gradientes aprovechan la estructura compuesta manteniendo las razones del método gradiente para las clases de problema suave.

2. **Big Data scaling via randomization**

Se menciona que los métodos de primer orden son recomendables para abordar problemas a gran escala. Sin embargo, en la práctica los cálculos numéricos exactos que exigen sus iteraciones pueden hacer estos métodos simples no factibles a medida que las dimensiones del problema crecen. Afortunadamente, resulta que primer orden los métodos son bastante robustos para usar aproximaciones de sus primitivas de optimización, como el gradiente y cálculos proximales.

Esta sección describe las aproximaciones aleatorias emergentes que aumentan el alcance de los métodos de primer orden a escalas extraordinarias.

2.1 Coordinate descent methods

Para calcular el gradiente completo de PageRank se requiere de operaciones matriz vector en cada iteración. Un vector menos costoso consiste en seleccionar una coordenada i de x y modificar únicamente la variable correspondiente x_i para mejorar la función objetivo. Esta idea es la esencia de los métodos de descenso coordinado.

2.2 Stochastic gradient methods

Este método a diferencia de los métodos aleatorizados de descenso gradiente actualiza todas las coordenadas de manera simultánea con gradientes aproximados.

2.3 Randomized linear algebra

Cuando se requiere para la solución de problemas de Big Data, el uso de operaciones de álgebra lineal, estas pueden representar cuellos de botella dada la dependencia lineal en dimensiones. Sin embargo, cuando las matrices tienen una representación low-rank

3. **The role of parallel and distributed computation**

Mientras que los métodos de primer orden parecen idealmente adecuados para mejorar de rendimiento, dos problemas nos bloquean cuando usamos hardware distribuido y heterogéneo:

• **Comunicación**: enlaces de comunicación desiguales o defectuosos entre las computadoras y dentro de la jerarquía de memoria local puede reducir significativamente la eficiencia numérica general de métodos de primer orden. Dos enfoques abordan ampliamente tales inconvenientes. Primero, podemos diseñar específicamente algoritmos que minimizan la comunicación. En segundo lugar, podemos eliminar un vector maestro x^k y en su lugar, trabajar con una copia local en cada máquina que conduzca a un consenso x^* en la convergencia.

• **Sincronización**: para realizar exactamente los cálculos de forma distribuida de tipo primer orden los métodos deben coordinar las actividades de diferentes computadoras cuyas primitivas numéricas dependen del mismo vector x^k en cada iteración. Sin embargo, este procedimiento se ralentiza incluso cuando una sola máquina lleva mucho más tiempo que las otras. Contra este problema los algoritmos asíncronos permiten actualizaciones utilizando versiones desactualizadas de sus parámetros

4 Outlook for convex optimization

Para resolver problemas de Big Data se requiere enfocarnos en la forma en que diseñamos algoritmos de optimización convexos, y sugerir elecciones computacionales no convencionales. Para resolver problemas cada vez más grandes de optimización con un crecimiento relativamente modesto en recursos computacionales, este artículo señala que debemos identificar algoritmos de estructuras dependientes clave.

Dado que las limitaciones de sincronización y comunicación del hardware disponible naturalmente dicta la elección de los algoritmos, esperamos que las nuevas herramientas de aproximación sigan adaptando algoritmos convexos a las plataformas computacionales heterogéneas. También se prevé una mayor utilización de modelos compuestos para hacer frente al ruido y otras limitaciones.