Karen Peña Albarrán
175952

Control de lectura:

"Convex optimization for big data"

Volkan Cevher, Stephen Becker, and Mark Schmidt

El artículo condensa los avances que se tienen para desarrollar algoritmos de optimización convexos para *big data* pues ellos buscan hacer más eficiente las áreas donde convergen las herramientas computacionales, de almacenamiento y de comunicaciones.

La organización de este documento se encuentra desarrollada en torno a los algoritmos de optimización compuesta los cuales se basan en los tres pilares: métodos de primer orden, aleatorización, cómputo en paralelo y distribuido.

Métodos de primer orden

Los métodos de primer orden utilizan las soluciones numéricas a partir de los métodos de optimización pues funcionan incluso para funciones que no son suaves mediante el análisis de vecindad. En este tipo de métodos es posible incluir regularizaciones, incluso a funciones no suaves pues pueden desempeñar un papel indispensable en la calidad de la solución.

El método de gradiente consiste en calcular el gradiente en un punto (local) y avanzar una determinada unidad (tamaño de paso) en la dirección de mayor crecimiento/decrecimiento y posteriormente realizar iteraciones.

$$x^{k+1} = x^k - \alpha_k \nabla f(x^k),$$

El método de gradiente acelerado logra la mejor tasa de error posible del peor de los casos, y por lo tanto, se lo conoce como un método óptimo de primer orden.

El problema compuesto canónico, donde la función objetivo consiste en una función convexa diferenciable f y una función convexa no suave g. Se basa en una aproximación cuadrática local simple de f:

$$x^{k+1} = \underset{y \in \mathbb{R}^p}{\operatorname{argmin}} \left\{ f(x^k) + \nabla f(x^k)^T (y - x^k) + \frac{1}{2\alpha_k} \left\| y - x^k \right\|^2 \right\}$$

Los objetivos compuestos están lejos de ser problemas genéricos de optimización convexa no suave. Los métodos de descenso por gradiente aprovechan la estructura compuesta para mantener las mismas tasas de convergencia del método de gradiente para las clases de problemas suaves.

El método acelerado de gradiente proximal se define de la siguiente forma:

Algorithm 2 Accelerated proximal gradient method to solve (1) [11, 15]. Set $v^0 = x^0$.

1: $x^{k+1} = \text{prox}_{\alpha_k g}\left(v^k - \alpha_k \nabla f(v^k)\right)$
2: $v^{k+1} = x^{k+1} + \beta_k(x^{k+1} - x^k)$

Los métodos de primer orden no son directamente aplicables por tanto, encontraremos útil considerar la siguiente forma funcional Estableciendo que los operadores de proximidad de h y g son ambos eficientes.

$$\min_{x,z\in\mathbb{R}^p} \left\{F(x,z) \stackrel{\text{def}}{=} h(x) + g(z) : \Phi z = x\right\}$$

Es posible aplicar un algoritmo simple, denominado método de multiplicación de multiplicadores (ADMM) para sus soluciones, que aprovecha poderosas técnicas aumentadas de Lagrange y descomposición dual.

No debemos olvidar que en ocasiones la matriz no es diagonalizable y puede ocurrir que no exista convergencia. Para solucionar ese problema, el autor describe un algoritmo que ayuda a resolver el problema.

Escalamiento BigData vía aleatorización

Las técnicas de aleatorización mejoran la escalabilidad de los métodos de primer orden ya que controlan su comportamiento además, incluyen actualizaciones parciales aleatorias de variables de optimización pues reemplazan el gradiente determinista y los cálculos proximales con estimadores estadísticos y acelerando las rutinas básicas de álgebra lineal mediante la aleatorización.

A pesar de ello, comentan que en la práctica, los cálculos numéricos y las iteraciones de los métodos de primer orden pueden hacer que incluso estos métodos simples sean inviables a medida que crecen las dimensiones del problema.

Para solucionar lo anterior, existen aproximaciones aleatorias emergentes que aumentan el alcance de los métodos de primer orden.

a) Coordinar los métodos de descenso: El cálculo del gradiente completo para la formulación del problema del PageRank requiere una operación matriz-vector en cada iteración. Una operación de vector más económica sería elegir una coordenada i de x y solo modificar la variable correspondiente xi para mejorar la función objetivo.

b) Métodos de gradiente estocástico: estos métodos actualizan todas las coordenadas simultáneamente pero usan gradientes aproximados.

c) Álgebra lineal aleatorizada: se busca construir una representación de bajo rango por selección de subconjunto de columna o fila para acelerar el cálculo es decir, aproximar M= Q (QTM) con Q en R^{p*r}.

El rol del cómputo en paralelo y distribuido

En este apartado, los autores plantean los dos grandes problemas que actualmente están dominando

a) Problema de comunicación por no diseñar de forma adecuada los canales dentro de los algoritmos.
b) Problema de sincronización en la solución de tareas en forma distribuida

En el caso del primer problema, sugieren que pueden ser solucionados a partir de minimizar la cantidad de comunicación y la creación de copias locales. Para el segundo problema, se sugiere el uso de algoritmos asíncronos que utilizan una versión no actualizada del vector.

Dentro de este tema, se considera indispensable el uso de tecnologías que tienden a la paralelización como hadoop, spark, etc.

Finalmente, el autor sugiere que es necesario usar modelos compuestos por los tres grandes rubros que generen soluciones que conjuguen la facilidad de interpretación de las condiciones de primer orden optimizados con aleatorización al momento de elegir las coordenadas y pensando e implementando por medio de una filosofía intrínseca de paralelización.