El método de región de confianza para funciones con ruido

José Miguel Saavedra Aguilar
Carlos Manuel Bosch Machado

Optimización
Centro de Investigación en Matemáticas, A.C.

28 de mayo de 2023

CIMAT

Esquema

Esquema

Métodos de región de confianza

En los métodos clásicos de región de confianza, se busca obtener la dirección de descenso minimizando el modelo

$$\begin{aligned} d_k = \ & \underset{d \in \mathbb{R}^n}{\arg\ \min} && m_k(d) := \frac{1}{2} d^\top B_k d + g_k^\top d + f(x_k) \\ & \text{sujeto a} && \|d\| \leq \Delta_k \end{aligned} \tag{1}$$

Para $g_k = \nabla f(x_k)$ y B_k una matriz SPD que aproxima a la Hessiana $\nabla^2 f(x_k)$.

Métodos de región de confianza

Una vez elegida la dirección, se calcula la *medida de confianza* ρ:

$$\rho_k = \frac{f(x_k + d_k) - f(x_k)}{m_k(d_k) - m_k(0)}$$

Métodos de región de confianza

Con base en el valor de ρ_k y dadas $\eta_1 < \eta_2$, y $\lambda_1 < 1 < \lambda_2$ actualizamos el radio Δ_k de la forma:

$$\Delta_{k+1} = \begin{cases} \lambda_1 \Delta_k & \rho_k < \eta_1 \\ \Delta_k & \rho_k \in [\eta_1, \eta_2] \\ \lambda_2 \Delta_k & \rho_k > \eta_2 \end{cases}$$

Finalmente, se actualiza x_k de la forma:

$$x_{k+1} = \begin{cases} x_k & \rho_k < \eta_1 \\ x_k + d_k & \rho_k \geq \eta_1 \end{cases}$$

Esquema

Funciones con ruido

En la actualidad, los fenómenos complejos son representados por medio de una función ruidosa:

$$f(x) = f_s(x) + \varepsilon_f(x) \tag{2}$$

donde $\varepsilon(x)$ es un término de ruido, que es supuesto aleatorio.

Funciones con ruido

Algunos ejemplos de funciones con ruido

- Regresión logística
- Experimentos en laboratorio (Por ejemplo, reacciones químicas o física cuántica)

Método de región de confianza para funciones con ruido

En el caso que $f(x)$ sea una función con ruido, si aplicamos el método de región de confianza usual, es posible que para $\Delta_k \ll 1$, ρ_k sea dominado por el ruido[Moré and Wild, 2012].
Por este motivo, [Sun and Nocedal, 2022] proponen la medida de confianza tolerante al ruido:

$$\rho_k = \frac{f(x_k + d_k) - f(x_k) - r\epsilon_f}{m_k(d_k) - m_k(0) - r\epsilon_f} \tag{3}$$

Donde ϵ_f es una cota superior del ruido de f, es decir, $|\varepsilon_f| \leq \epsilon_f$ y $r = \frac{2}{1-\eta_2}$.

Método de región de confianza para funciones con ruido

Tomando los siguientes supuestos:

1. El gradiente de la función objetivo $\nabla f_s(x)$ es Lipschitz continua.
2. Los términos de error ε_f y ε_g están acotados:

$$|\varepsilon_f| \leq \epsilon_f \qquad\qquad |\varepsilon_g| \leq \epsilon_g$$

3. Existe una constante $L_B > 0$ tal que las matrices B_k satisfacen:

$$\|B_k\| < L_B, \forall k$$

4. El paso p_k es cuando menos tan bueno como el del punto de Cauchy:

$$m_k(0) - m_k(p_k) \geq m_k(0) - m_k(p_k^C) \geq \frac{1}{2}\|g_k\| \operatorname{mín}\left\{\Delta_k, \frac{\|g_k\|}{\|B_k\|}\right\}$$

5. f_s está acotada inferiormente.

Método de región de confianza para funciones con ruido

Bajo estos supuestos, [Sun and Nocedal, 2022] muestra que el algoritmo de región de confianza para funciones con ruido genera una sucesión que visita infinitas veces una región crítica C_1 dada por:

$$C_1 := \left\{ x : \left\| g(x) \leq (r+1)\epsilon_g + \frac{\beta}{2} \right\| \right\}$$

y que una vez que visitamos la región crítica C_1, no podemos alejarnos mucho de esta región.

Estimando la cota del ruido

Una vez que para calcular la medida de confianza ρ_k requerimos una cota del ruido de f, estimamos ϵ_f utilizando el algoritmo `ECNoise` de [Moré and Wild, 2011], evaluando $f(x)$ en m puntos equiespaciados y tomando las diferencias:

$$\Delta^{k+1} f(t) = \Delta^k f(t+h) - \Delta^k f(t)$$

Estimando la cota del ruido

La varianza de la k-ésima diferencia es

$$\mathrm{Var}\left(\Delta^k \epsilon_f(x)\right) = \frac{\mathrm{Var}\left(\epsilon_f\right)}{\gamma_k}$$
$$\gamma_k = \frac{(k!)^2}{(2k)!}$$

De forma que

$$\left(\gamma_k \mathbb{E}\left[(\Delta^k f(x))^2\right]\right)^{1/2} \to \mathrm{Var}\left(\epsilon_f\right) \tag{4}$$

cuando $h = x_{i+1} - x_i \to 0$.

Estimando la cota del ruido

Una vez que aproximamos la varianza de ε_f, podemos aproximar ϵ_f por la desigualdad de Chebyshev[Wasserman, 2004]:

$$\mathbb{P}\left(\left|\varepsilon_f - \mathbb{E}\left[\varepsilon_f\right]\right| \geq k\sqrt{\mathrm{Var}\left(\varepsilon_f\right)}\right) \leq \frac{1}{k^2}$$

Actualización por BFGS

La ecuación de actualización BFGS para la aproximación de la Hessiana está dado por:

$$B_{k+1} = B_k + \frac{\mathbf{y}_k \mathbf{y}_k^{\mathrm{T}}}{\mathbf{y}_k^{\mathrm{T}} \mathbf{s}_k} - \frac{B_k \mathbf{s}_k \mathbf{s}_k^{\mathrm{T}} B_k^{\mathrm{T}}}{\mathbf{s}_k^{\mathrm{T}} B_k \mathbf{s}_k} \tag{5}$$

donde B_k es la aproximación actual de la matriz Hessiana, $y_k = \nabla f(x_{k+1}) - \nabla f(x_k)$ es la diferencia de gradientes entre dos iteraciones sucesivas, $s_k = x_{k+1} - x_k$ es la diferencia de posiciones entre dos iteraciones sucesivas. [Nocedal and Wright, 2006]

Actualización por BFGS

En el caso de la implementación de BFGS para el método de región de confianza, la actualización de s_k va a ser la dirección d_k obtenida que minimiza el modelo (1), en este caso se toma la aproximación de Dogleg:

$$d_k = \begin{cases} -\frac{\Delta_k}{\|g_k\|} g_k, & \Delta_k \leq \|g_k\| \\ -(t g_k + (1-t) B_k^{-1} g_k), & \|g_k\| < \Delta_k < \left\| B_k^{-1} g_k \right\| \\ -B_k^{-1} g_k, & \left\| B_k^{-1} g_k \right\| \leq \Delta_k \end{cases}$$

Esquema

ECNoise

Algoritmo 1: Algoritmo ECNoise.

Entrada: f, x_0, m, h
Salida: $\hat{\epsilon}_f$

para $i = 0$ ***hasta*** m **hacer**
  $T_i^0 \leftarrow f(x_0 + ih)$;
fin
para $k = 0$ ***hasta*** $m - 1$ **hacer**
  $\gamma_k \leftarrow \frac{(k!)^2}{(2k)!}$;
  $\sigma_k \leftarrow \sqrt{\frac{\gamma_k}{m+1-k} \sum_{i=0}^{m-k} (T_i^k)^2}$;
  para $i = 0$ ***hasta*** $m - k$ **hacer**
    $T_i^{k+1} \leftarrow T_{i+1}^k - T_i^k$;
  fin
fin

Trust Region

Algoritmo 2: Algoritmo de Región de confianza para funciones con ruido con BFGS.

Entrada: f, x_0, ϵ_f
Salida: x^*

$k \leftarrow 0$;
$r \leftarrow 2/(1-\eta_2)$;
aproximar por diferencias finitas $g_0 \approx \nabla f(x_0)$;
Inicializar $B_0 = \mathbb{I}$;
mientras $\|g_k\| > 0$ **hacer**
    aproximar d_k, la solución de (1);
    calcular $\rho_k = \frac{f(x_k) - f(x_k + d_k) + r\epsilon_f}{m_k(0) - m_k(d_k) + r\epsilon_f}$;
    si $\rho_k < \eta_1$ **entonces**
        $\Delta_{k+1} \leftarrow \lambda_1 \Delta_k$
    en otro caso
        $x_{k+1} \leftarrow x_k + d_k$;
        Aproximar por diferencias finitas $g_{k+1} \approx \nabla f(x_{k+1})$;
        $y_k \leftarrow g_{k+1} - g_k$;
        $B_{k+1} \leftarrow B_k + \frac{y_k y_k^\top}{y_k^\top d_k} - \frac{B_k d_k d_k^\top B_k}{d_k^\top B_k d_k}$;
        si $\rho_k > \eta_2$ **entonces** $\Delta_{k+1} \leftarrow \lambda_2 \Delta_k$;
    fin
    $k \leftarrow k + 1$;
fin

Esquema

Resultados

Se modificó el algorimo `ECNoise` disponible en el sitio web de Stefan Wild para trabajar en Julia y poder tomar como argumentos de entrada la función f, el punto x_0, el número de evaluaciones m y el tamaño de paso h, de forma que coincida con el pseudocódigo 1. De esta forma, se puede incorporar directamente en el algoritmo 2.

Resultados

Como cota del gradiente, una vez que:

$$\frac{\partial f}{\partial x_i} f(x) = \frac{f_s(x + he_i) - f_s(x - he_i)}{2h} + \frac{\varepsilon_f(x + he_i) - \varepsilon_f(x - he_i)}{2h}$$

tenemos que $(\epsilon_g(x))_i \approx \frac{k\epsilon_f}{\sqrt{2}h}$, de forma que la desviación estándar de $\|\epsilon_g(x)\|$ está acotada por $\omega = \frac{\sqrt{n}\epsilon_f}{\sqrt{2}h}$, y tomamos

$$tol = 10^{-6} + \frac{\sqrt{n}\hat{\epsilon}_f}{2h} = 10^{-6} + \frac{\omega}{\sqrt{2}}$$

para la convergencia, con $h = 10^{-3}$, tomando en cuenta [Moré and Wild, 2012].

Resultados

A modo de prueba, tomamos la función de Rosenbrock para $n = 2$ y $n = 100$ más un ruido normal con media 0 y varianza 10^{-4} y 10^{-6} respectivamente, así como la función de Wood más un ruido uniforme $(0, 10^{-5})$.

Los resultados se muestran en la tabla:

Función	Convergencia	Iteraciones	$\hat{\epsilon}_f$	ϵ_f
Rosenbrock $n = 2$	Si	28	8.50747×10^{-5}	10^{-4}
Rosenbrock $n = 100$	Si	1627	1.32527×10^{-6}	10^{-6}
Wood	Si	155	2.14961×10^{-6}	$\frac{1}{2\sqrt{3}} \times 10^{-5}$

Tabla 1: Resultados del algortimo 2 para funciones con ruido aleatorio.

Moré, J. J. and Wild, S. M. (2012).
Estimating derivatives of noisy simulations.
ACM Transactions on Mathematical Software, 38(3):1–21.

Moré, J. J. and Wild, S. M. (2011).
Estimating computational noise.
SIAM Journal on Scientific Computing, 33(3):1292–1314.

Nocedal, J. and Wright, S. (2006).
***Numerical Optimization*.**
Springer Series in Operations Research and Financial Engineering. Springer, New York, NY, 2 edition.

Sun, S. and Nocedal, J. (2022).
A trust region method for the optimization of noisy functions.

Referencias ii

Wasserman, L. (2004).
***All of Statistics*.**
Springer New York.