Métodos de optimización convexa en aprendizaje automatico

Álvaro Vielba

20 de marzo de 2019

Resumen

Muchos métodos clásicos de aprendizaje supervisado, en contextos de regresión o de clasificación, presentan comportamientos problemáticos en el análisis de datos de alta dimensión, cada vez más frecuentes en las aplicaciones. Desde la introducción de los métodos SVM (Cortés y Vapnik, 1995) y lasso (Tibshirani, 1996) muchos métodos de aprendizaje se han adaptado al contexto de alta dimensión mediante la penalización de las funciones criterio clásicas con términos adicionales orientados a conseguir mejores propiedades estadísticas de los estimadores resultantes (mayor estabilidad, dispersión, adaptación, etc). Desde un punto de vista computacional, el cálculo de las reglas de aprendizaje anteriores se reduce a un problema de optimización convexa. Los métodos clásicos de solución numérica (de descenso por gradiente o de tipo Newton) pueden ser adaptados para aprovechar la estructura especial de algunos de estos problemas, lo que posibilita el tratamiento eficiente de grandes conjuntos de datos. En este trabajo se explorarán algunas de las técnicas recientes desarrolladas en este campo para la resolución numérica de problemas de aprendizaje automático, con atención especial a las capaces de tratar con funciones criterio no suaves (tal como la correspondiente al lasso), capaces de aprovechar representaciones estocásticas de la función objetivo.

Índice general

Introducción **7**

Notación básica **9**

1. Optimización convexa **11**
1.1. Convexidad: Nociones básicas y su importancia en optimización 11
1.2. Algunas consideraciones sobre la convergencia 14

2. Métodos de descenso por gradiente y otras variantes **17**
2.1. Descenso por gradiente y descenso por gradiente proyectado 17
2.1.1. Funciones β-suaves 27
2.1.2. Funciones fuertemente convexas 31
2.1.3. Funciones β-suaves y α-fuertemente convexas 33
2.2. El método de Frank-Wolfe 41
2.3. Cotas inferiores 44
2.4. Descenso geométrico 45
2.4.1. Interpretación geométrica 45
2.4.2. El método 48
2.4.3. Convergencia e implementación 49
2.5. Descenso por gradiente acelerado de Nesterov 50

3. Métodos alternativos **55**
3.1. ISTA (Iterative Shrinkage-Thresholding Algorithm) 55
3.1.1. Aplicación proximal y envolvente de Moreau 56
3.1.2. Descripción del método 58
3.1.3. Convergencia 59
3.2. FISTA (Fast ISTA) 63

4. Implementación práctica del problema LASSO **67**
4.1. Regresión lineal regularizada 67
4.1.1. La regresión ridge 69
4.1.2. LASSO 71
4.2. Descenso por coordenadas 76
4.3. Resolución del problema LASSO en R 77
4.3.1. Definición del problema estandarizado con condiciones centradas 77
4.3.2. Resolución mediante FISTA 79
4.3.3. Resolución mediante descenso múltiple por coordenadas 81
4.3.4. Comparación de los métodos 81

Conclusión **85**

Bibliografía **87**

Introducción

El aprendizaje automático o *machine learning*, como es originalmente conocido, ha evolucionado con el paso de los años desde un juego experimental allá por la década de los 50 a una herramienta fundamental empleada por entidades bancarias, gobiernos, empresas de marketing y ventas o servicios sanitarios para mejorar sus prestaciones. Esta disciplina engloba todo tipo de métodos capaces de predecir el comportamiento más probable a partir de un histórico de datos de tamaño considerable. Y como siempre, cuando se trata de realizar cualquier acción de la mejor manera posible, se llega a un problema de optimización. Es por ello que los problemas de optimización juegan un papel fundamental dentro del aprendizaje automático. Además dada la ingente cantidad de datos que se manejan en la sociedad actual se hace necesario que los algoritmos destinados a resolver estos problemas sean suficientemente rápidos y escalables, a parte de gozar de una serie de características de interés (estabilidad, interpretabilidad...). Esta es la razón principal por la que en este trabajo se considerarán problemas de optimización convexa; y es que, cuando se trata de optimizar funciones convexas, la velocidad de los algoritmos a la hora de encontrar soluciones cercanas al verdadero valor aumenta notablemente. No solo eso, sino que es frecuente que de una manera directa o indirecta la convexidad aparezca en los problemas que se tratan, ya que aunque un modelo dado por una función convexa puede no ser tan fiel a la realidad como otro modelo representado por una función más compleja, este último será en la mayoría de los casos irresoluble. Por su parte, los problemas de optimización convexa, entre los que están incluidos el problema de mínimos cuadrados y los problemas de programación lineal, se van a poder manejar con mayor o menor facilidad en un gran número de ocasiones.

El objetivo de este trabajo es presentar y analizar diversos métodos de optimización convexa y aplicar alguno de ellos en el ámbito del aprendizaje automático. El punto de partida serán los métodos de descenso por gradiente. Se trata de algoritmos aplicables bajo hipótesis de regularidad bastante débiles para las funciones objetivo, incluso se pueden extender para funciones no diferenciables haciendo uso de la noción de subgradiente, que exploran el espacio ambiente en busca de la dirección de máximo descenso. Esta cualidad es ampliamente aprovechable cuando se trata con funciones convexas, pues de presentar un único mínimo relativo, este resulta ser además es el mínimo global en este tipo de funciones. Es conveniente hacer mención de los métodos de segundo orden, que requieren de la existencia de matriz hessiana para la función objetivo. Es cierto que ofrecen una convergencia mucho más rápida bajo condiciones de regularidad, eso sí, más exigentes; sin embargo, no se considerarán, ya que suponiendo que sea factible obtener una expresión para la matriz hessiana, algo ya poco probable, si se tiene en cuenta que el espacio en el que se va a trabajar será de dimensión elevada, su cálculo será indudablemente costoso (de orden superior al cálculo del gradiente) pudiendo acarrear incluso problemas de almacenamiento. El inconveniente que pueden plantear los métodos de descenso por gradiente por su parte consiste en una mala elección de la longitud

de paso que conduzca a soluciones no deseadas o directamente a la divergencia del método. Por ello el foco de interés se centrará en elegir longitudes de paso de tal manera que sea posible obtener cotas sobre la complejidad de los algoritmos, entendiendo por *complejidad* el número de iteraciones necesarias para aproximarse con una precisión suficiente al óptimo u optimizador real de la función objetivo.

Así se discutirá en el capítulo inicial la importancia de la optimizacion convexa, exponiendo algunos resultados que explican las ventajas de este tipo de optimización. En el segundo capítulo se desarrollará el método clásico de descenso por gradiente observando cómo a través de ciertos requerimientos sobre la función objetivo, siendo siempre uno de ellos la convexidad, una elección adecuada de la longitud de paso acelera la convergencia del método. Se analizará la complejidad de estas nuevas versiones del descenso por gradiente y, por último, se completará el capítulo con algoritmos que para idénticas exigencias sobre la función objetivo mejoran la velocidad de convergencia de los métodos de descenso por gradiente. En el tercer capítulo se tratará con situaciones menos favorables, en las que se goza siempre de convexidad pero no de las otras propiedades ventajosas. Reemplazando estas propiedades por un conociemiento sobre la estructura de la función a optimizar se verá que es posible recuperar las cotas óptimas sobre la complejidad que se tenían en el capítulo anterior y se llegará al método ISTA y a su versión aún más rápida, FISTA. Finalmente, en el capitulo 4, se implementarán este último algoritmo y el descenso múltiple por coordenadas (ampliamente utilizado pero aparentemente más lento al realizar actualizaciones para cada coordenada) para la resolución de un problema habitual en el aprendizaje automático y que deriva de una penalización sobre el modelo de regresión lineal a partir de la norma l_1, conocido como LASSO. La meta es llegar a unos resultados que avalen la velocidad de convergencia de los algoritmos de optimización convexa.

Notación básica

Se ha creído conveniente incluir esta breve sección que explica el significado de parte de la notación que se utilizará durante el resto del trabajo con la intención de evitar confusiones y clarificar el contenido del mismo antes de empezar a analizar la materia de interés.

Se trabajará, a no ser que se diga lo contrario, en el espacio ambiente $\mathbb{R}^n$. Un elemento cualquiera $x \in \mathbb{R}^n$ vendrá caracterizado por lo tanto por n componentes que se denotarán, cada una de ellas, por $x(i), \quad i = 1, \ldots, n \quad$ (a no ser que se especifique claramente que $x = (x_1, \ldots, x_n)$). En el caso de trabajar con matrices $X \in \mathbb{R}^n \times \mathbb{R}^n$ se denotará por $X(i,j)$ el elemento que ocupa la fila i-ésima y la columna j-ésima . $X(i,:)$ hará referencia a la fila i-ésima, mientras que $X(:,j)$, a la columna j-ésima.

Sean $x, y \in \mathbb{R}^n$. El producto escalar usual entre ambos se denotará o bien por el producto matricial $x^T y$ o bien por la notación $< x, y >$.

$$x^T y =< x, y >= x(1)y(1) + \ldots + x(n)y(n)$$

$||\cdot||$ denotará por lo general la norma dos, procedente de este producto escalar, a no ser que explícitamente se diga que $||\cdot||$ hace referencia a una norma cualquiera. En este caso se escribiría $||\cdot||_2$ para especificar que se está usando la norma dos.

Dado un punto $x \in \mathbb{R}^n$, $B(x,r)$ y $\bar{B}(x,r)$ denotarán respectivamente la bola abierta y la bola cerrada de radio $r > 0$ (de nuevo respecto a la distancia inducida por el producto escalar usual).

$$B(x,r) = \{y \in \mathbb{R}^n : ||x - y|| < r\}$$
$$\bar{B}(x,r) = \{y \in \mathbb{R}^n : ||x - y|| \leq r\}$$

$X \subset \mathbb{R}^n$ admitirá la posibilidad $X = \mathbb{R}^n$ y por lo tanto para hablar de una contención estricta se escribirá $X \subsetneq \mathbb{R}^n$.

X^C denotará, como es usual, el complementario de X.

$Int(X)$, $\overline{X}$ y δX denotarán respectivamente el interior, la adherencia y la frontera de X.

$$\begin{aligned} \text{Int}(X) &= \{x \in X : \exists r > 0 \ \text{ con } \ B(x,r) \subset X\} \\ \overline{X} &= \{x \in X : \forall r > 0, \quad B(x,r) \cap X \neq \emptyset\} \\ \delta X &= \bar{X} \cap \overline{\mathbb{R}^n \setminus X} \end{aligned}$$

El resto de la notación se introducirá a lo largo del texto cuando sea necesario.

Capítulo 1

Optimización convexa

En este capítulo inicial se darán las definiciones de convexidad relativas a conjuntos y funciones y se discutirán las razones que priorizan la optimización convexa frenta a cualquier otra. Para terminar se tratarán una serie de cuestiones sobre qué se va a entender por "complejidad" de un algoritmo de optimización convexa y por qué.

1.1. Convexidad: Nociones básicas y su importancia en optimización

Sean $x, y \in \mathbb{R}^n$, el conjunto de puntos de la forma

$$(1-\phi)x + \phi y \text{ con } \phi \in \mathbb{R}$$

constituye la línea entre x e y. Además si se considera exclusivamente $\phi \in [0,1]$ se habla del segmento de extremos x e y. Así, un conjunto convexo será aquel que contiene todos los segmentos de extremos dos de sus puntos.

Más formalmente, un conjunto $X \subset \mathbb{R}^n$ es convexo si para dos puntos cualesquiera $x, y \in X$ y para todo $\phi \in [0,1]$ se tiene que

$$(1-\phi)x + \phi y \in X$$

El concepto de convexidad no se ciñe solo a los conjuntos, se puede extender a otros objetos matemáticos, las funciones.

Una función con llegada en $\mathbb{R}$ es convexa si la imagen por f de un punto de un segmento es menor que el punto correspondiente (el de idéntico valor del parámetro ϕ) del segmento de extremos las imágenes de los extremos del segmento original, es decir, si el grafo de f en un segmento queda por debajo del segmento cuyos extremos son las imágenes de los extremos del primero.

Más formalmente, una función $f : X \subset \mathbb{R}^n \to \mathbb{R}$ es convexa si

$$f((1-\phi)x + \phi y) \leq (1-\phi)f(x) + \phi f(y)$$

para todo $x, y \in X$ y para todo $\phi \in [0,1]$.

De aquí en adelante el objetivo será el de minimizar una función real definida en $\mathbb{R}^n$ o en un subconjunto de $\mathbb{R}^n$.

$$\min_{x \in X \subset \mathbb{R}^n} f(x)$$

Aunque para aclarar ciertas ideas se darán ejemplos de funciones de las que se conoce una expresión para $\nabla f(x)$ y se puede resolver la ecuación $\nabla f(x) = 0$ o llegar a una ecuación resoluble mediante el teorema de los multiplicadores de Lagrange, el interés no se centrará en estos casos. Por el contrario se centrará en la minimización de funciones para las que no resulta factible obtener una solución analítica de $\nabla f(x) = 0$ y se recurre a un proceso iterativo a través del cual se obtienen puntos del dominio de f que se van aproximando al mínimizador o cuya imagen se va aproximando al mínimo de f. En algunos casos f no será diferenciable con lo cual no cabe la posibilidad siquiera de plantearse resolver la ecuación $\nabla f(x) = 0$ y también se intentará llegar a un aproximante lo más cercano posible al mínimo o minimizador real de la función mediante un proceso iterativo.

Aún más, el foco de interés se centrará en minimizar funciones convexas definidas generalmente en conjuntos convexos de $\mathbb{R}^n$. El comportamiento de este tipo de funciones va a dar lugar a algoritmos más rápidos. En el capítulo 2 se hablará de uno de ellos en particular. Ahora se presentarán un par de resultados que justifican el buen comportamiento de las funciones convexas.

Proposición 1.1 (Los mínimos locales son mínimos globales) Sea $f : X \subset \mathbb{R}^n \to \mathbb{R}$ convexa. Si x es un mínimo local de f entonces x es también un mínimo global de f.

Dem: Sea x un mínimo local de f, sea $y \in X$ y sea $\phi > 0$ suficientemente pequeño para que $(1-\phi)x + \phi y$ pertenezca al entorno de x donde este minimiza la función. Entonces

$$f(x) \leq f((1-\phi)x + \phi y) \leq (1-\phi)f(x) + \phi f(y)$$

y se deduce que $f(x) \leq f(y)$.

□

Proposición 1.2 (Caracterización del mínimo de una función convexa) Dados $X \subset \mathbb{R}^n$ cerrado y convexo, U abierto de $\mathbb{R}^n$ y $f : X \to \mathbb{R}$ convexa y diferenciable en $U \supset X$ entonces se tiene que

$$x^* \in argmin_{x \in X} f(x) \Leftrightarrow \nabla f(x^*)^T (x^* - y) \leq 0, \quad \forall y \in X$$

Nótese que trivialmente, si $x^* \in int(X)$, estas dos condiciones son equivalentes a $\nabla f(x^*) = 0$.

Dem:

$\Leftarrow$ Para la condición suficiente se probará primero que

$$f(x) - f(y) \leq \nabla f(x)^T (x - y) \, \forall \, x, y \in X$$

Sea $\phi \in [0, 1]$ y sea $h = \phi(y - x)$. Para todo $x, y \in X$ basta ver que se verifica, gracias a la convexidad de f

$$\begin{aligned} f(y) \quad &\geq \quad f(x) + \frac{f((1-\phi)x + \phi y) - f(x)}{\phi} = f(x) + \frac{f(x+h) - f(x)}{h}(y - x) \\ &\longrightarrow \quad f(x) + \nabla f(x)^T (y - x) \text{ cuando } \phi \longrightarrow 0 \; (h \longrightarrow 0) \end{aligned}$$

Si existe $x^* \in X$ tal que para todo $y \in \mathbb{R}^n$, $\nabla f(x^*)^T(x^* - y) \leq 0$ aplicando lo anterior se llega a que $f(x^*) \leq f(y)$, por lo tanto $x^* \in argmin_{x \in X} f(x)$.

$\Rightarrow$ Para la condición necesaria se considera para cada $y \in X$ la función real de variable real $h_y(t) := f(x^* + t(y - x^*))$ cuya derivada en virtud de la regla de la cadena es $h'_y(t) = \nabla f(x^* + t(y - x^*))^T(y - x^*)$. Dado que $h_y(0) = f(x^*)$, 0 es un punto de mínimo de la función, por lo que existe $\delta > 0$ para el cual $h'_y(t) \geq 0$ si $|t| < \delta$. En particular $h'_y(0) = \nabla f(x^*)^T(y - x^*) \geq 0$, de donde se obtiene el resultado.

□

Ejemplo 1.3 La proyección de un punto $z \in \mathbb{R}^n$ sobre la bola unidad centrada en el origen es

$$\begin{cases} z & si \quad ||z|| \leq 1 \\ \frac{z}{||z||} & si \quad ||z|| > 1 \end{cases}$$

La proposición 1.2 da una sencilla justificación para este hecho. Sea $f(x) = \frac{||x-z||}{2}$. Proyectar z sobre la bola unidad centrada en el origen no es otra cosa que calcular

$$\underset{x \in \mathbb{R}^n : ||x|| \leq 1}{\arg\min} ||x - z|| = \underset{x \in \mathbb{R}^n : ||x|| \leq 1}{\arg\min} f(x)$$

y dado que f es claramente una función convexa y $\nabla f(x) = \frac{x-z}{||x-z||}$, la proposición 1.2 afirma que x^* será un mínimizador de f, si y sólo si, $(x^* - z)^T(x^* - y) \leq 0 \ \forall \, y \in \mathbb{R}^n$ tal que $||y|| \leq 1$. Si $||z|| \leq 1$ es obvio que el propio z es un minimizador para f, en caso contrario, si $||z|| > 1$, se comprueba que $x^* = \frac{z}{||z||}$.

$$\begin{aligned} (x^* - z)^T(x^* - y) \leq 0 \quad &\Leftrightarrow \quad ||x^*||^2 - (x^*)^T(y + z) + z^T y \\ &\Leftrightarrow \quad 1 - \frac{z^T y}{||z||} - ||z|| + z^T y \leq 0 \\ &\Leftrightarrow \quad 1 - ||z|| + z^T y \left(\frac{||z|| - 1}{||z||} \right) \\ &\Leftrightarrow \quad \frac{z^T y}{||z||} - 1 \leq 0 \end{aligned}$$

Y la última desigualdad es cierta por el hecho de que $||z|| = sup_{||y|| \leq 1} z^T y$.

x^* es el único minimizador posible ya que la proyección sobre un conjunto convexo, como es la bola unidad, es única. Se justificará en el teorema 2.1.

Dos factores resultan clave para entender el interés por optimizar funciones convexas definidas en conjuntos convexos. El primero es la simplicidad que estas ofrecen gracias a la aparición de fenómenos globlales que sustituyen a fenómenos locales que se dan en funciones arbitrarias. Esto queda patente en la proposición 1.1. Los extremos relativos de funciones convexas son además extremos absolutos y de hecho son mínimos, lo cual se traduce en un ahorro computacional considerable. Una característica local ampliamente conocida de cualquier función es el gradiente, que en el caso de existir nos aporta información local sobre el comportamiento de la función.

Este concepto viene habitualmente sustituido en optimización convexa por el de subgradiente, ampliamente relacionado, y que nos ofrece una cota inferior para la función en todo su dominio, no solo localmente, como se verá en la sección 2.2.

Por otro lado este bajo coste computacional que aporta el buen comportamiento de las funciones convexas no serviría de nada si en los problemas de optimización estas apenas aparecieran. No es que a la hora de modelar un fenómeno aparezcan de forma natural habitualmente, pero la mayoría de los problemas de optimización son irresolubles, mientras que los problemas de optimización convexa admiten una solución. Este es el motivo, por ejemplo, de que la programación lineal sea tan empleada cuando el comportamiento de cualquier proceso en el mundo real es no lineal. En otro contexto, como es el del problema de clasificación, es conocido que el problema de minimizar el error aparente de clasificación, asociado a la pérdida 0-1, conduce a problemas de optimización irresolubles en la práctica (véase [10], por ejemplo,). Esto se puede superar con convexificaciones del problema, como la que lleva al SVM. La reformulación convexa constituye por lo tanto esta segunda clave, aunque no se vaya a tratar más allá de esta breve mención en este texto.

1.2. Algunas consideraciones sobre la convergencia

En el capítulo 2 se empezará a hablar de un método iterativo para la optimización de funciones conocido como descenso por gradiente (DG). Tanto para este como para el resto de métodos que aparecen a lo largo del texto resulta de interés estudiar la convergencia de los aproximantes a mínimo o minimizador de la función objetivo que estos generan. Usualmente se calcula el número de operaciones (costo computacional) del que precisan los algoritmos para acercarse lo suficiente al objeto aproximado. Sin embargo, los métodos de descenso por gradiente utilizan en su proceso iterativo el valor del gradiente (o en su defecto el subgradiente) de la función en el iterante inmediatamente anterior. Otros, como se irá viendo, calculan la imagen de la función en un determinado punto. En cualquier caso el costo computacional de estos métodos quedaría supeditado al número de operaciones que conlleva el cálculo de estas evaluaciones, y por lo tanto, a la propia función.

Con ánimo de evitar esta dificultad se estudiará la convergencia a partir del número de iteraciones necesarias para obtener un aproximante a distancia menor que un cierto $\epsilon > 0$ del verdadero mínimo o minimizador; lo que se llamará ϵ-*aproximante*. Se hará uso de la notación O de Landau indicando que un número de iteraciones $g(\epsilon)$ es $\mathrm{O}(h(\epsilon))$ refiriéndose siempre a ϵ en un entorno de 0 si no se indica lo contrario, y se llamará indistintamente *orden de convergencia* o *complejidad* de un método al número de iteraciones $h(\epsilon)$ del que este precisa para llegar a un ϵ-aproximante. Los algoritmos de descenso por gradiente que se abordarán parten de los siguientes supuestos. El dominio de la función $X \subset \mathbb{R}^n$ es conocido. No lo es así la función que se desea optimizar, pero es posible conocer cierta información sobre ella (bien la imagen de f en un punto, bien el subgradiente de f en un punto).

En [4] se habla del modelo black-box. El término *black-box* hace referencia a un objeto, sistema o algoritmo cuyo funcionamiento interno es opaco y que funciona en términos de estímulos y respuestas. Así, en el contexto matemático, se están enmarcando los algoritmos de descenso

por gradiente dentro de este modelo, ya que se supone desconocida la expresión explícita de la función y solo se puede acceder a ella a través del del subgradiente (respuesta) en determinados puntos (estímulos). Se habla así de oráculo cada vez que se obtiene una información acerca de la función a partir del subgradiente en un punto y se cuenta el número de ocasiones en las que se recurre a un oráculo para llegar a un ϵ-aproximante (*complejidad oráculo*). En la mayoría de ocasiones esta coincide con el número de iteraciones que realiza el algoritmo, luego con la complejidad tal y como se ha definido.

Tales consideraciones sobre la convergencia ofrecen dos ventajas fundamentales. La primera se apreciará en los métodos de descenso por gradiente, la independiencia respecto a la dimensión. Muchos algoritmos basados en el modelo black-box logran que el número de iteraciones necesario para alcanzar un ϵ-aproximante se desapegue por completo de la dimensión, lo cual resulta conveniente cuando se trabaja en espacios de dimensión elevada. La otra gran ventaja consiste en que muchos de estos algoritmos son poco sensibles a errores en la información que nos ofrece la función.

Capítulo 2

Métodos de descenso por gradiente y otras variantes

Como su propio nombre indica, el descenso por gradiente es un método de descenso (además lineal). Esto quiere decir que cada iterante se obtiene como combinación (lineal) del iterante anterior y de una dirección de búsqueda graduada por una longitud de paso.

$$x_{t+1} = x_t + \eta_t \Delta x_t$$

La dirección de descenso para el descenso por gradiente será la opuesta al propio gradiente, y de ahí su nombre. A partir de este método se desarrollarán otros para solventar ciertos problemas, como el descenso por gradiente proyectado (DGP), que hace posible considerar el hecho de que la dirección de búsqueda conduzca a un punto exterior al dominio de la función; el descenso por gradiente condicional, que enriquece el método con propiedades apreciables como la dispersión de los iterantes; o los métodos de descenso geométrico, que mejoran la convergencia hacia el óptimo. Todos ellos se tratarán como algoritmos de tipo black-box y ofrecerán, en consecuencia, cotas de convergencia independientes de la dimensión que se analizarán en cada caso.

En particular, para DG y DGP se irán imponiendo condiciones de reguralidad cada vez más fuertes sobre la función objetivo con la idea de acelerar la convergencia lo máximo posible. Estas cotas serán de hecho óptimas bajo las hipótesis dadas en 1.2 acerca del desconocimiento de la expresión explícita de la función (es decir, dentro del modelo black-box), salvo para un par de casos que dan pie a hablar del método de descenso geométrico y el método acelerado de Nesterov como colofón al capítulo. Estos alcanzan la cota óptima bajo las mismas condiciones exigidas en los dos casos mencionados.

2.1. Descenso por gradiente y descenso por gradiente proyectado

El método de descenso por gradiente (*gradient descent*) es uno de los métodos más conocidos para encontrar el mínimo de una función $f : \mathbb{R}^n \to \mathbb{R}$ diferenciable. Este método aprovecha la información ofrecida por el gradiente de una función, es decir, hacia dónde crece la función, para moverse precisamente en sentido contrario pues el objetivo es ir descendiendo hasta encontrar el

mínimo. Partiendo de un punto $x_1 \in \mathbb{R}^n$ se realiza la siguiente iteración para cada $t \geq 1$

$$x_{t+1} = x_t - \eta_t \nabla f(x_t), \;\; \eta_t \in \mathbb{R} \tag{2.1.1}$$

La razón que hay detrás de (2.1.1) es como ya se ha dicho la de moverse en la dirección de máximo descenso y se puede comprobar fácilmente que $-\nabla f(x_t)$ es efectivamente esta dirección deseada puesto que minimiza el polinomio de Taylor de orden 1 de la función objetivo. Si en cada paso el desplazamiento se realiza en la dirección $u \in \mathbb{R}^n$ unitario, es decir, de x_t a $x_{t+1} = x_t + \eta_t u$, se desea que $f(x_{t+1}) - f(x_t)$ sea lo menor posible pues esto indica que el descenso en ese paso es máximo. Si se define $g : \mathbb{R} \to \mathbb{R}$ por $g(\eta_t) = f(x_t + \eta_t u)$ se puede ver utilizando el polinomio de Taylor de orden 1 de g que

$$f(x_{t+1}) - f(x_t) = g(\eta_t) - g(0) = g'(0)\eta_t + O(\eta_t^2)$$

Como por la regla de la cadena $g'(\eta_t) = \nabla f(x_t + \eta_t u)^T u$ se obtiene

$$f(x_{t+1}) - f(x_t) = \eta_t \nabla f(x_t)^T u + O(\eta_t^2)$$

y claramente el mínimizador de esta última expresión en u unitario es $\frac{-\nabla f(x_t)}{||\nabla f(x_t)||}$.

Ahora bien, no solo es importante la dirección de búsqueda. Una longitud de paso imprecisa puede hacer que se produzca una demora considerable del proceso o incluso que nunca se llegue a alcanzar un minimizador con tolerancia menor que un cierto $\epsilon > 0$.

Se puede observar (Figura 2.1) cómo en el caso de funciones que presentan varios extremos

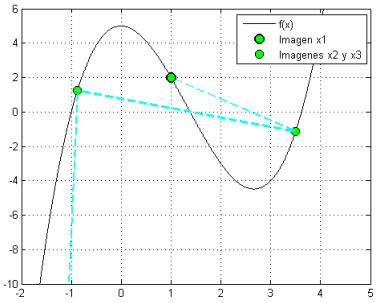


Figura 2.1: Ejemplo de descenso por gradiente con paso $\eta = \frac{1}{2}$

relativos, veáse $x^3 - 4x^2 + 5$, una elección inconveniente de la longitud de paso puede conducir a un aproximante de un mínimo distinto al que se estaba buscando. Sin embargo, si la función es convexa, y no lineal, el mínimo de la función se va a alcanzar y va a ser único, con lo cual el descenso por gradiente converge irremediablemente hacia él. Si además se elige η_t convenientemente, se puede conseguir acelerar al máximo el método como se tratará de hacer de aquí en adelante. A partir de la subsección 2.1.1 se pedirán ciertos requerimientos de regularidad a la función objetivo (suvidad, convexidad fuerte o ambas simultáneamente) con el objetivo de acelerar la convergencia de DG y DGP. Para ello es fundamental acertar con la longitud de paso como se verá a través de distintos teoremas que, tomando como hipótesis que la función objetivo cumple con los requerimientos exigidos, garantizan cotas sobre la complejidad de los métodos a partir de las cuales estos proporcionan con total certeza un ϵ-aproximante (ya sea a mínimo o

minimizador). El aproximante que toman los distintos métodos puede ser el último iterante que se obtiene, la imagen de este o bien alguna clase de combinación convexa de todos los iterantes obtenidos.

Por otra parte; en la mayoría de los problemas de optimización que se tratan f está definida en $X \subsetneq \mathbb{R}^n$ por lo cual no se tiene la certeza de que $x_{t+1} \in X$ en (2.1.1). La forma más obvia de solventar esta situación parece ser la de, en el caso de que el punto obtenido no pertenezca a X, tomar el punto más próximo que sí cumpla esta propiedad. Se tropieza sin embargo con la falta de garantías de que este punto sea único, a no ser que X tenga unas propiedades favorables, y aquí vuelve a entrar en juego la convexidad.

Teorema 2.1 (Teorema de la proyección convexa) Sea $X \subset \mathbb{R}^n$ convexo, no vacío y cerrado y sea $y \in \mathbb{R}^n$. $\exists! \, \bar{x} \in X$ tal que $||y - \bar{x}|| \leq ||y - x||, \quad \forall x \in X$.

Dem: Sea $r > 0$ tal que $A := \overline{B}(y, r) \cap X$ es no vacío. Nótese que A es compacto por ser intersección de un compacto y un cerrado. Entonces la función d_y definida por $d_y(x) = ||x - y||$ alcanza su mínimo en A, al tratarse de una función continua, es decir, existe $\bar{x} \in A$ tal que $d_y(\bar{x}) \leq d_y(x) \, \forall \, x \in A$.

Además $\bar{x}$ es el mínimo de d_y en X pues si $x \in X \setminus \overline{B}(y, r)$ se tiene que $||x - y|| > r \geq ||y - \bar{x}||$.

Para probar la unicidad se supone que existen dos puntos x_1, x_2 de proyección de y. Por la convexidad de X el punto medio $x = \frac{x_1 + x_2}{2}$ del segmento entre ambos también pertenece a X. Ahora, ya que $d_y(x_1) = d_y(x_2)$

$$\begin{aligned} \frac{1}{2} \langle x_1 - x_2, y - x \rangle &= \frac{1}{2} \left\langle x_1 - x_2, y - \frac{1}{2}(x_1 + x_2) \right\rangle \\ &= \frac{1}{4} \langle (y - x_2) - (y - x_1), (y - x_1) + (y - x_2) \rangle \\ &= \frac{1}{4}(||y - x_1||^2 - ||y - x_2||^2) = 0 \end{aligned}$$

Por el teorema de Pitágoras se verifica $||y - x||^2 + ||\frac{x_1 - x_2}{2}||^2 = ||y - x_2||^2$ y teniendo en cuenta que $x - x_2 = \frac{x_1 - x_2}{2}$ se tiene la siguiente igualdad

$$||y - x||^2 + ||x - x_2||^2 = ||y - x_2||^2$$

de modo que $||y - x||^2 \leq ||y - x_2||^2$. La igualdad se da, si y solo si, $x = x_2$, por lo que $x_1 = x_2$.

□

Este $\bar{x}$ se llama proyección convexa de y sobre X y se denotará por $\Pi_X(y)$.

Se cumple además la siguiente desigualdad

$$(\Pi_X(y) - y)^T (\Pi_X(y) - x) \leq 0, \quad \forall x \in X$$

que en particular implica

$$||\Pi_X(y) - y||^2 + ||\Pi_X(y) - x||^2 \leq ||y - x||^2 \tag{2.1.2}$$

La primera desigualdad es consecuencia directa de la proposición 1.2. dado que por el teorema previo $\Pi_X(y) = \arg\min_{x\in X} ||x-y||$. La segunda se deduce de la anterior y de la siguiente cadena de desigualdades

$$\begin{aligned} ||y-x||^2 &= ||y-\Pi_X(y)+\Pi_X(y)-x||^2 \\ &= ||y-\Pi_X(y)||^2+||\Pi_X(y)-x||^2+2(y-\Pi_X(y))(\Pi_X(y)-x) \\ &\geq ||y-\Pi_X(y)||^2+||\Pi_X(y)-x||^2 \end{aligned}$$

El teorema 2.1 adquiere una importancia fundamental, pues es el parche que se va a utilizar para adaptar el descenso por gradiente a problemas de optimización con restricciones dando lugar a la primera de sus variantes, el descenso por gradiente proyectado. Antes de describir el método se hace necesario introducir la noción de subgradiente, que ya se adelantaba en el capítulo 1, así como una serie de resultados que se pueden encontrar en cualquier libro de análisis convexo (veáse [9], que constituye una referencia clásica en este campo) y que son esenciales para entender la geometría de los conjuntos convexos y dar una interpretación del concepto de subgradiente.

Definición 2.2 Sea $X \subset \mathbb{R}^n$, sea $f : X \to \mathbb{R}$. $\xi \in \mathbb{R}^n$ se denomina subgradiente de f en x si para todo $y \in \mathbb{R}^n$

$$f(x) - f(y) \leq \xi^T(x-y) \tag{2.1.3}$$

El conjunto de subgradientes de f en x se denomina subdiferencial de f en x y se denota por $\partial f(x)$.

$$\partial f(x) = \left\{\xi \in \mathbb{R}^n : f(x) - f(y) \leq \xi^T(x-y)\right\}$$

Definición 2.3 Sea $f : X \subset \mathbb{R}^n \to \mathbb{R}$. Se denomina epígrafo de f y se denota epi(f) al conjunto

$$\text{epi}(f) = \{(x,t) \in X \times \mathbb{R} : t \geq f(x)\}$$

Esta noción va a caracterizar además a las funciones convexas.

Lema 2.4 Una función $f : X \to \mathbb{R}$ es convexa si y solo si su epígrafo es convexo.

Dem: Para la condición necesaria, sean $(x_1,t_1),(x_2,t_2) \in \text{epi}(f)$ y sea $\phi \in [0,1]$

$$f((1-\phi x_1)+\phi x_2) \leq (1-\phi)f(x_1)+\phi f(x_2) \leq (1-\phi)t_1+\phi t_2$$

Para la condición suficiente basta darse cuenta de que dados $x_1, x_2 \in X$, entonces $(x_1, f(x_1)), (x_2, f(x_2)) \in$epi($f$).

□

Se darán a continuación dos teoremas claves para el desarrollo de algunos métodos de optimización convexa (Veánse, por ejemplo, el método del centro de gravedad, el método del elipsoide o el método del plano de corte de Vaidya en [4]). No se aplican directamente en el

descenso por gradiente pero se recurre a ellos para demostrar la proposición 2.7, que garantiza la existencia de subgradientes cuando la función es convexa y además ofrece otra caracterización de una función convexa a partir de la existencia de subgradientes.

Teorema 2.5 (Teorema del hiperplano separante) Sea $X \subset \mathbb{R}^n$ convexo, no vacío y cerrado y sea $y \notin X$. Existen $p \in \mathbb{R}^n, p \neq \vec{0}$ y $\alpha \in \mathbb{R}$ tales que $p^T x \leq \alpha,\ \forall x \in X,\ p^T y > \alpha$ $(p^T x < p^T y)$.

El vector p dado por el teorema es el vector normal a un hiperplano que divide el espacio ambiente en dos subespacios, dejando de un lado el punto exterior y y del otro los puntos del conjunto X.

Dem: Sea $\Pi_X(y)$ la proyección de y sobre X. Se verifica por el teorema de la proyección

$$(\Pi_X(y) - y)^T(\Pi_X(y) - x) \leq 0, \quad \forall x \in X$$

Se definen $p := y - \Pi_X(y)$ y $\alpha = p^T \Pi_X(y)$. Como $y = p + \Pi_X(y)$, $p^T y = ||p||^2 + \alpha > \alpha$ ($||p||^2 \neq 0$ pues $y \notin X$). Por otro lado, por el teorema de la proyección

$$p^T x = (y - \Pi_X(y))^T (x - \Pi_X(y)) + \alpha \leq \alpha$$

□

Esta no es más que una versión elemental de un resultado mucho más general para en espacio vectorial cualquiera, como es el Teorema de Hahn-Banach. Sin embargo, una demostración del Teorema de Hahn Banach requeriría del Lema de Zorn, en cambio, esta versión emplea únicamente resultados elementales.

Teorema 2.6 (Teorema del hiperplano soporte) Sea $X \subset \mathbb{R}^n$ convexo y sea $\bar{x} \in \delta X$. $\exists p \in \mathbb{R}^n$, $p \neq \vec{0}$ / $p^T(x - \bar{x}) \leq 0, \quad \forall x \in X$ $(p^T x \leq p^T \bar{x})$.

La interpretación es similar a la del teorema anterior, solo que ahora el punto $\bar{x}$ pertenece al propio hiperplano y el resto de puntos del conjunto quedan estrictamente por encima de él.

Dem: Se hará uso de argumentos topológicos, así como del teorema del hiperplano separador para la prueba de este teorema.

$$\bar{x} \notin \text{Int}(X) = \text{Int}(\overline{X}) \Rightarrow \forall k \in \mathbb{N}, \quad \text{existe } y_k \in B\left(\bar{x}, \frac{1}{k}\right) \cap \overline{X}^c.$$

Gracias al teorema anterior se puede suponer que $\exists p_k \in \mathbb{R}^n, ||p_k|| = 1$ (sin pérdida de generalidad), con $p_k^T y_k > p_k^T x,\ \forall x \in \overline{X}$ y como la sucesión $(p_k)_k$ es acotada existe una subsucesión suya que converge a $p \in \mathbb{R}^n$. Tomando el límite cuando $k \longrightarrow \infty$ en la desigualdad $p_k^T y_k > p_k^T x$ se concluye la prueba.

□

Proposición 2.7 (Existencia de subgradientes) Dados $X \subset \mathbb{R}^n$ convexo y $f : X \to \mathbb{R}$. Si f es convexa entonces $\partial f(x) \neq \emptyset,\ \forall x \in int(X)$. Por otro lado, si $\forall x \in X,\ \partial f(x) \neq \emptyset$, entonces f es convexa. Además si f es convexa y diferenciable en X entonces $\nabla f(x) \in \partial f(x)$.

Dem:

1. $\forall x \in X$, $\partial f(x) \neq \emptyset \Rightarrow f$ convexa.

   Sean $x, y \in X$ y sea $\xi \in \partial f((1-\phi)x + \phi y)$, por definición de subgradiente

   $$f((1-\phi)x + \phi y) \leq f(x) + \phi \xi^T (y - x) \quad (1)$$
   $$f((1-\phi)x + \phi y) \leq f(y) + (1-\phi) \xi^T (x - y) \quad (2)$$

   Basta operar $(1-\phi) \cdot (1) + \phi \cdot (2)$ para concluir que f es convexa.

2. f convexa $\Rightarrow \forall x \in \text{Int}(X)$, $\partial f(x) \neq \emptyset$.

   Sea $x \in \text{Int}(X)$. Como f es convexa también lo es su epígrafe. Además claramente $(x, f(x)) \in \text{epi}(f)$. De hecho, $(x, f(x)) \in \delta\ \text{epi}(f)$. Usando el teorema del hiperplano soporte tenemos garantizada la existencia de $(a, b) \in \mathbb{R}^n \times \mathbb{R}$, $(a, b) \neq (\vec{0}, 0)$ tales que

   $$\begin{aligned} (a,b)^T (x, f(x)) &\geq (a,b)^T (y,t)\ \ \forall (y,t) \in \text{epi}(f) \text{ o sea} \\ a^T x + b f(x) &\geq a^T y + bt \end{aligned}$$

   Haciendo $t \to \infty$ se ve claramente que $b \leq 0$ y, como para ϵ suficientemente pequeño $x + \epsilon a \in X$, si se toma $y = x + \epsilon a$ se llega a $bf(x) \geq \epsilon ||a||^2 + bt$. De aquí se deduce que además $b < 0$. Por lo tanto

   $$f(x) - f(y) \leq \frac{1}{|b|} a^T (x - y) \Rightarrow \frac{1}{|b|} a \in \partial f(x)$$

3. Si f diferenciable, $\nabla f(x) \in \partial f(x)$.

   La prueba se ha dado en la proposición 1.2

□

Proposición 2.8 Sea $X \subset \mathbb{R}^n$ convexo, abierto y no vacío, y sea $f : X \to \mathbb{R}$ convexa y diferenciable en un punto $x_0 \in \text{Int}(X)$. Entonces $\partial f(x_0) = \{\nabla f(x_0)\}$.

Dem: Como f es convexa $\partial f(x_0) \neq \emptyset$. Sea $\xi \in \partial f(x_0)$, $d \in \mathbb{R}^n$ y $\delta > 0$ suficientemente pequeño. Por definición de subgradiente

$$\frac{f(x_0 + \delta d) - f(x_0)}{\delta} \geq \xi^T d$$

Haciendo $\delta \to 0$, dado que f es diferenciable, se obtiene $\nabla f(x_0)^T d \geq \xi^T d$ para toda dirección $d \in \mathbb{R}^n$. Solo cabe que $\xi = \nabla f(x_0)$.

□

La proposición 2.8 deja claro que el subgradiente no es más que un concepto que generaliza el de gradiente de una función. Cuando la función en cuestión es diferenciable el subdiferencial es un conjunto unipuntual cuyo único elemento es el gradiente. Como se adelantaba en el capítulo 1 el gradiente es en principio un fenómeno local si lo interpretamos como el vector normal a un hiperplano que pasa por un punto del grafo de la función y queda por debajo de este en un

entorno de dicho punto. En el caso en que se introduzca la convexidad, tanto en la función como en su dominio, pasa a convertirse en un fenómeno global. El hiperplano definido por el gradiente no es solo inferior al grafo de la función alrededor del punto soporte, sino en toda la función. Si la función no es diferenciable no podemos hablar de gradiente pero, supuesta la convexidad, la proposición 2.7 nos garantiza la existencia de subgradientes que juegan un papel análogo al definir hiperplanos soporte que dejan el grafo de la función sobre ellos.

El grafo de una función no es generalmente un conjunto convexo, pero sí su epígrafo. Es más, $\mathrm{G}(f) \subset \mathrm{epi}(f)$, de hecho $\mathrm{G}(f) = \partial\mathrm{epi}(f)$ y por lo tanto el epígrafo de una función convexa es un conjunto convexo y cerrado. En virtud del teorema 2.6 existe un hiperplano soporte para cada punto de la frontera de $\mathrm{epi}(f)$, es decir, para cada punto de $\mathrm{G}(f)$ y podemos dar la expresión analítica para tantos hiperplanos soporte como subgradientes conozcamos. Sea $(x, f(x)) \in \mathrm{G}(f)$ y sea $\xi \in \partial f(x)$ entonces $p = (\xi, -1)$ es el vector normal a un hiperplano soporte para $(x, f(x))$; ya que si $(y, t) \in \mathrm{epi}(f)$.

$$\begin{aligned}(\xi, -1)^T((y,t) - (x, f(x))) &\leq \xi^T(y - x) + f(x) - f(y) \\ &\leq \xi^T(y - x) + \xi^T(x - y) = 0\end{aligned}$$

Estos resultados permiten adaptar el descenso por gradiente a dos nuevos tipos de situaciones: restricciones en el espacio ambiente teniendo que optimizar en un subespacio convexo de este y funciones no diferenciables para las que no es posible entonces utilizar el gradiente en (2.1.1). El resultado es un método más general conocido como descenso por gradiente proyectado, que solventa simultáneamente ambas situaciones partiendo de las hipótesis siguientes (a las que se hará referencia como *hipótesis generales*)

Se supone $X \subsetneq \mathbb{R}^n$ contenido en una bola de centro el iterante inicial x_1 y de radio $R > 0$, $f : X \to \mathbb{R}$ una función tal que para todo $x \in X$, $\partial f(x) \neq \emptyset$ (lo cual implica directamente que f es convexa, proposición 2.7) y además si $\xi \in \partial f(x)$ entonces $||\xi|| \leq L$ para cierto $L > 0$. Esta última suposición implica en particular que f es L-Lipschitziana, basta aplicar la definición de subgradiente y Cauchy-Schwartz.

$$|f(x) - f(y)|^2 \leq ||\xi||^2||x - y||^2 \leq L^2||x - y||^2 \Rightarrow |f(x) - f(y)| \leq L||x - y||$$

Partiendo del iterante inicial $x_1 \in \mathbb{R}^n$ el nuevo método avanza de la siguiente manera para $t \geq 1$

$$y_{t+1} = x_t - \eta_t \xi_t \text{ con } \xi_t \in \partial f(x) \tag{2.1.4}$$
$$x_{t+1} = \Pi_X(y_{t+1}) \tag{2.1.5}$$

(2.1.4) coincide con (2.1.1) cuando f es diferenciable en virtud de la proposición 2.8 pero también admite que f no sea diferenciable y en ese caso toma una dirección que guarde cierta relación con $\nabla f(x)$, un subgradiente $\xi_x \in \partial f(x)$. Posteriormente se analizará cómo el subgradiente aporta un descenso hacia el mínimo o minimizante con la elección de un paso adecuado. En (2.1.5) se

proyecta en X el punto obtenido.

Utilizando lo desarrollado hasta ahora, en particular de nuevo la definición de subgradiente y la segunda consecuencia de el Teorema 2.1 sobre la proyección convexa, obtenemos el primero de una serie de resultados sobre la complejidad asociado al DG y sus variantes (en este caso, al DGP).

Teorema 2.9 El método del descenso por gradiente proyectado bajo las hipótesis generales y con $\eta = \frac{R}{L\sqrt{t}}$ satisface

$$f\left(\frac{1}{t}\sum_{s=1}^{t} x_s\right) - f(x^*) \leq \frac{RL}{\sqrt{t}} \tag{2.1.6}$$

Dem: Se hará uso de la siguiente igualdad $||a||^2 + ||b||^2 - ||a-b||^2 = 2a^T b$ que se obtiene desarrollando $||a-b||^2$ como producto escalar

$$\begin{aligned} f(x_s) - f(x^*) &\leq \xi_s^T (x_s - x^*) \\ &= \frac{1}{\eta}(x_s - y_{s+1})^T (x_s - x^*) \\ &= \frac{1}{2\eta}(||x_s - y_{s+1}||^2 + ||x_s - x^*||^2 - ||y_{s+1} - x^*||^2) \\ &= \frac{\eta}{2}||\xi_s||^2 + \frac{1}{2\eta}(||x_s - x^*||^2 - ||y_{s+1} - x^*||^2) \end{aligned}$$

donde se ha utilizado la definición de subgradiente para la desigualdad, (2.1.4) para la primera y la tercera igualdad y $||a||^2 + ||b||^2 - ||a-b||^2 = 2a^T b$ para la segunda.

Por el teorema 2.1, dado que x_{t+1} es la proyección convexa de y_{t+1}, se comprueba $||x_{s+1} - x^*||^2 \leq ||y_{s+1} - x^*||^2$. Utilizando la convexidad de f, lo obtenido anteriormente, esta última desigualdad, las propiedades de las sumas telescópicas y que $||x_1 - x^*|| \leq R$ deducimos finalmente el resultado tras reemplazar en el último paso el valor de η

$$\begin{aligned} f(\tfrac{1}{t}\sum_{s=1}^{t} x_s) \quad & -f(x^*) \leq \frac{1}{t}\sum_{s=1}^{t}(f(x_s) - f(x^*)) \\ \leq \quad & \frac{1}{t}\sum_{s=1}^{t}\left(\frac{\eta}{2}||\xi_s||^2 + \frac{1}{2\eta}\left(||x_s - x^*||^2 - ||y_{s+1} - x^*||^2\right)\right) \\ \leq \quad & \frac{\eta L^2}{2} + \frac{1}{2t\eta}R^2 \\ \leq \quad & \frac{RL}{\sqrt{t}} \end{aligned} \tag{2.1.7}$$

□

Como se adelantaba en el capítulo inicial el interés por esta variante del descenso por gradiente radica en que la dimensión del espacio en el cual se trabaja no juega ningún papel en

la convergencia del método como se puede observar en (2.1.6). Esto es importante ya que actualmente en muchos casos se tienen en cuenta un número elevado de características de un determinado proceso o fenómeno.

En cualquier caso se desea acelerar este método. Si se analiza (2.1.6) en profundidad se observa que realizando al menos $\frac{(RL)^2}{\epsilon^2}$ iteraciones se obtendrá un aproximante a dcon un exceso menor que ϵ del mínimo de la función, es decir un ϵ-aproximante. Esta cota es demasiado elevada en comparación con lo que es posible obtener. En [4] se analiza el método de plano de corte de Vaidya, un algoritmo black-box que a pesar de que no cuenta con indenpendencia de dimensión alcanza un ϵ-aproximante en $O(nln(\frac{n}{\epsilon}))$ iteraciones. Así en lo que resta de sección se verá que pidiendo una mayor regularidad a la función objetivo se pueden acelerar notablemente el descenso por gradiente y su versión proyectada hasta llegar una complejidad de $O(ln(\frac{1}{\epsilon}))$. Por otra parte en la sección 2.3 de [4] se dan cotas inferiores para el número de iteraciones que indican que esta complejidad es prácticamente la mejor que podemos esperar de un algoritmo de tipo black box con las exigencias que se le requerirán a f.

Retomando el teorema 2.9, se observa que η es dependiente, en este caso, del número total de iteraciones que vayamos a realizar y esto puede resultar incómodo si de partida no se tiene claro este número. Se puede sustituir por $\eta_s = \frac{R}{L\sqrt{s}}$ para cada iteración sin apenas alterar la cota dada por el teorema 2.9.

Corolario 2.10 En las condiciones del teorema anterior, el descenso por gradiente proyectado con $\eta_s = \frac{R}{L\sqrt{s}}$ satisface

$$f\left(\frac{1}{t}\sum_{s=1}^{t} x_s\right) - f(x^*) \leq \frac{3RL}{2\sqrt{t}}$$

Dem: Se parte de la expresión (2.1.7) de la demostración del teorema 2.9 con η dependiente de s: η_s. El segundo sumando se acota de la misma manera teniendo en cuenta que $\frac{1}{\eta_t} > \frac{1}{\eta_s}$ para todo $s < t$ y que $\eta_t = \frac{R}{L\sqrt{t}}$; en el primero ahora se tiene una serie no constante

$$\begin{aligned}
(2.1.7) \quad &= \frac{1}{t}\sum_{s=1}^{t}\left(\frac{\eta_s}{2}||\xi_s||^2 + \frac{1}{2\eta_s}(||x_s - x^*||^2 - ||y_{s+1} - x^*||^2)\right) \\
&\leq \frac{RL}{2t}\sum_{s=1}^{t}\frac{1}{\sqrt{s}} + \frac{L}{2R\sqrt{t}}(||x_1 - x^*||^2 - ||y_{t+1} - x^*||^2) \\
&\leq \frac{3RL}{2\sqrt{t}}
\end{aligned}$$

Ahora bien, $\sum_{s=1}^{t}\frac{1}{\sqrt{s}} \leq \int_0^t \frac{1}{\sqrt{s}}ds\ (= 2\sqrt{t})$ con lo cual se obtiene una cota casi idéntica a la dada en (2.1.6)

□

Ejemplo 2.11 Con objeto de ilustrar los métodos de descenso por gradiente se supondrá que se desea optimizar la función $f : \mathbb{R}^2 \to \mathbb{R}$ definida por $f(x) = ||x||^2$. Su mínimo se alcanza en el origen y su valor es nulo. Para el caso restringido se tomará $X = \bar{B}((0,2),1)$, en el cual el minimizador pasaría a ser $x^* = (0,1)$ y el correspondiente mínimo $f(x^*) = 1$. En este conjunto

f es L-lipschitziana con $L = 6$.

$$\begin{aligned} |f(x) - f(y)| &= |\,||x||^2 - ||y||^2\,| = |\,(||x|| + ||y||)(||x|| - ||y||)\,| \\ &\leq ||x - y||(||x|| + ||y||) \leq 6||x - y|| \end{aligned}$$

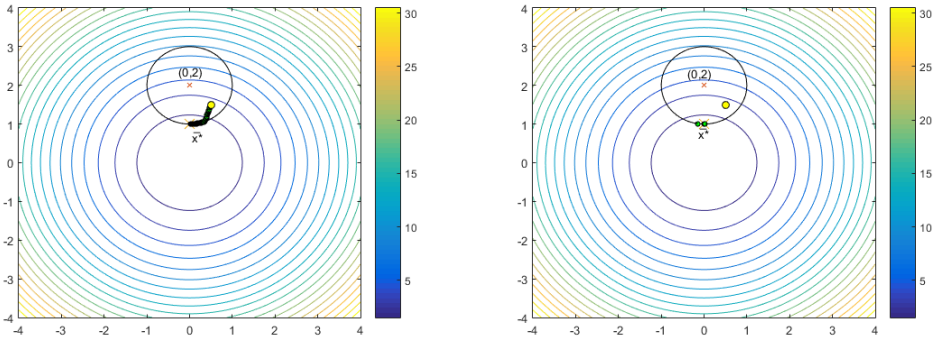
Proyectar un punto $x \in \mathbb{R}^2$ sobre X es algo sencillo pues basta trasladar el punto, proyectarlo sobre la bola centrada en el origen y de radio unidad (como se hizo en el ejemplo 1.3) y retrasladar esta proyección, es decir

$$\Pi_X(x) = (0,2) + \Pi_{B(\vec{0},1)}(x - (0,2))) = \begin{cases} x & si \quad x \in \bar{B}((0,2),1) \\ \frac{x-(0,2)}{||x-(0,2)||} + (0,2) & si \quad x \notin B((0,2),1) \end{cases}$$

De acuerdo con (2.1.4) y (2.1.5) y dado que $\nabla f(x) = 2x$, el descenso por gradiente proyectado aplicado a f produce en cada paso un iterante

$$\begin{aligned} y_{t+1} &= x_t(1 - 2\eta_t) \\ x_{t+1} &= \Pi_X(y_{t+1}) \end{aligned}$$

Así pues el teorema 2.9 garantiza que el método del gradiente proyectado alcanzará tras 100 iteraciones, y partiendo de un iterante inicial $x_1 = (0.5, 1.5)$, un punto promedio $\frac{1}{t}\sum_{s=1}^{t} x_s$ cuya imagen se encuentra a distancia $\frac{R \cdot 6}{\sqrt{100}} = 0.9$ del verdadero mínimo restringido, siempre y cuando se elija una longitud de paso fija igual a $\eta = \frac{R}{6 \cdot \sqrt{100}} = 0.025$. Se ha tomado $R = 1.5$, que constituye el valor más pequeño posible de tal manera que X esté contenido en la bola de centro x_1 y radio R. Tras implementar el método se obtiene (0.06701.0239), cuya imagen dista 0.0529 de x^*.



(a) Paso fijo $\eta = \frac{1}{100}$ (b) Paso variable $\eta_t = \frac{1}{\sqrt{t}}$

Figura 2.2: Descenso por gradiente proyectado para $f(x) = ||x||^2$

Usando el paso variable $\eta_t = \frac{1}{\sqrt{t}}$ ocurre que $\eta_4 = \frac{1}{2}$ y por lo tanto en el cuarto paso el algoritmo alcanza el verdadero mínimo restringido a X.

$$x_4 = \Pi_X(x_3\,(1-1)) = \Pi_X(\vec{0}) = x^*$$

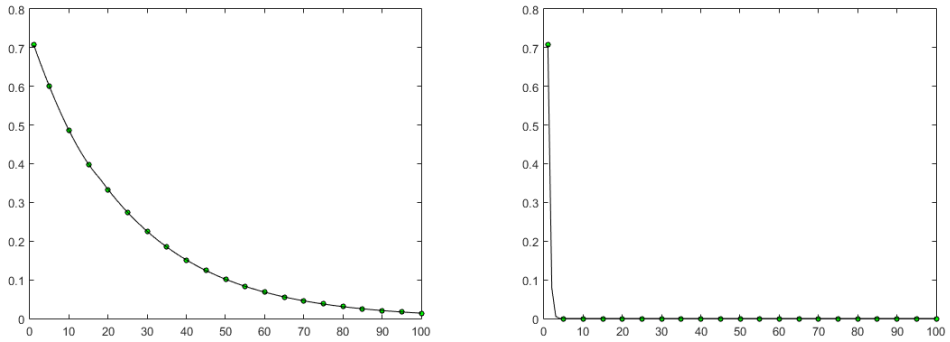
$$\text{y para } t > 4\,,\, x_{t+1} = \Pi_X\left(x^*(1-2\eta_t)\right) = x^* \text{ ya que } 2\eta_t \in (0,1)$$

.

Este hecho provoca que el punto promedio para el caso restringido este más próximo a x^* a pesar de que la cota teórica proporcionada por el corolario 3.10 sea igual a 1.35. El algoritmo produce un punto promedio $(0.0038, 1.0051)$ a distancia 0.0101. El funcionamiento del algoritmo puede verse en la Figura 2.2.

Al no producirse esta salvedad para el algoritmo con longitud de paso fija ocurre, tal y como se aprecia en la Figura 2.2 y más claramente en la Figura 2.3, que la distancia entre el iterante y el verdadero mínimo desciende con más demora quizá de la que se desearía y de ahí la posibilidad de plantearse un cambio en la longitud de paso que de alguna manera explote todas las características de regularidad que pudiera ofrecer la función objetivo.

Este ejemplo se retomará al final de la subsección 2.1.4 después de analizar la convergencia del



(a) Paso fijo $\eta = \frac{1}{100}$ (b) Paso variable $\eta_t = \frac{1}{\sqrt{t}}$

Figura 2.3: Distancia entre los iterantes y el minimizador en el descenso por gradiente proyectado para $f(x) = ||x||^2$

descenso por gradiente y el descenso por gradiente proyectado cuando, se exploran esas nuevas longitudes de paso en función de la estructura de f.

2.1.1. Funciones β-suaves

La primera particularidad que se pedirá a las funciones que se van a optimizar para acelerar la convergencia viene dada por la siguiente definición.

Definición 2.12 Sea $f : X \subset \mathbb{R}^n \to \mathbb{R}$, $f \in C^1(X)$. Diremos que f es β-suave si

$$||\nabla f(x) - \nabla f(y)|| \leq \beta||x-y|| \;\forall x, y \in X$$

A fin de cuentas se trata de pedir que el gradiente también sea lipschitziano (con constante de Lipschitz igual a β). Veáse como esta propiedad influye directamente en el descenso por gradiente.

Lema 2.13 Sea f una función convexa en $\mathbb{R}^n$, entonces f es β-suave, si y solo si, para todo $x, y \in X$

$$0 \leq f(x) - f(y) - \nabla f(y)^T(x-y) \leq \frac{\beta}{2}||x-y||^2 \tag{2.1.8}$$

Dem: La primera desigualdad se deduce directamente de la definición de subgradiente por lo que basta probar la segunda.

La condición necesaria se prueba escribiendo $|f(x) - f(y) - \nabla f(y)^T(x-y)|$ en la forma integral $|\int_0^1 \nabla f(y+t(x-y))^T(x-y)dt - \nabla f(y)^T(x-y)|$ y utilizando que f es β-suave.

$$\begin{aligned}\left|\int_0^1 \nabla f(y+t(x-y))^T(x-y)dt - \nabla f(y)^T(x-y)\right| &\leq \int_0^1 ||\nabla f(y+t(x-y)) - \nabla f(y)||||x-y||dt \\ &\leq \int_0^1 \beta t||x-y||^2 dt = \frac{\beta}{2}||x-y||^2\end{aligned}$$

La condición suficiente se deduce tomando $z = y - \frac{1}{\beta}(\nabla f(y) - \nabla f(x))$ y escribiendo $f(x) - f(y) = (f(x) - f(z)) + (f(z) - f(y))$ se utiliza que $\nabla f(x) \in \delta f(x)$ para desarrollar el primer sumando y la desigualdad (2.1.8) para el segundo.

$$\begin{aligned}f(x) - f(y) &= f(x) - f(z) + f(z) - f(y) \leq \nabla f(x)^T(x-z) + \nabla f(y)^T(z-y) \\ &\quad + \frac{\beta}{2}||z-y||^2 \\ &= \nabla f(x)^T(x-y) - \frac{1}{\beta}(\nabla f(y) - \nabla f(x))^T(y-z) + \frac{1}{2\beta}||\nabla f(x) - \nabla f(y)||^2 \\ &= \nabla f(x)^T(x-y) - \frac{1}{2\beta}||\nabla f(x) - \nabla f(y)||^2 \end{aligned} \tag{2.1.9}$$

Ahora reagrupando los términos convenientemente se cumple por (2.1.8)

$$\frac{1}{2\beta}||\nabla f(x) - \nabla f(y)||^2 \leq f(y) - f(x) - \nabla f(x)^T(y-x) \leq \frac{\beta}{2}||x-y||^2$$

de donde se deduce que $||\nabla f(x) - \nabla f(y)|| \leq \beta||x-y||$ y por lo tanto f es β-suave.

□

El lema anterior, en particular (2.1.8) va a marcar el camino para elegir el mejor η en la descripción del método. Obsérvese que si en (2.1.8) se toma $x = x_{s+1} = x_s - \frac{1}{\beta}\nabla f(x_s)$ e $y = x_s$ se obtiene

$$f(x_{s+1}) - f(x_s) \leq \frac{-1}{2\beta}||\nabla f(x_s)||^2 \tag{2.1.10}$$

y así en cada iteración se garantiza un acercamiento al mínimo de la función a un paso marcado por el gradiente del iterante inmediatamente anterior.

Se llega por lo tanto al siguiente resultado de convergencia del descenso por gradiente para funciones β-suaves.

Teorema 2.14 El método de descenso por gradiente para funciones β-suaves con $\eta = \frac{1}{\beta}$ satisface

$$f(x_t) - f(x^*) \leq \frac{2\beta||x_1 - x^*||^2}{t-1}$$

Dem: Se definen $\delta_s := f(x_s) - f(x^*)$ y $w := \frac{1}{2\beta||x_1 - x^*||^2}$. Como $\delta_{s+1} - \delta_s = f(x_{s+1}) - f(x_s)$ entonces se verifica por (2.1.10)

$$\delta_{s+1} \leq \delta_s - \frac{1}{2\beta}||\nabla f(x_s)||^2$$

Y se verifica por definición de subgradiente $\delta_s \leq \nabla f(x_s)^T(x_s - x^*) \leq ||x_s - x^*||\,||\nabla f(x_s)||$ deduciéndose $||\nabla f(x_s)|| \geq \frac{\delta_s}{||x_s - x^*||}$. Esta última desigualdad junto con la anterior nos llevan a

$$\delta_{s+1} \leq \delta_s - \frac{1}{2\beta||x_1 - x^*||^2}\delta_s^2$$

Nótese que en la última expresión se ha escrito $||x_1 - x^*||^2$ y no $||x_s - x^*||^2$ porque $||x_s - x^*||$ es decreciente (se probará al final de la demostración).

Sustituyendo $\frac{1}{2\beta||x_1 - x^*||^2}$ por w, dividiendo ambos lados de la desigualdad por δ_{s+1} y teniendo en cuenta que $\delta_{s+1} \leq \delta_s$ se llega a que $w\frac{\delta_s}{\delta_{s+1}} \leq \frac{1}{\delta_{s+1}} - \frac{1}{\delta_s}$ y en particular $w \leq \frac{1}{\delta_{s+1}} - \frac{1}{\delta_s}$. Para acabar basta sumar desde 1 hasta $t-1$ en ambos lados y tener en cuenta que la segunda serie es telescópica.

$$\begin{aligned}
\sum_{s=1}^{t-1}\left(\frac{1}{\delta_{s+1}} - \frac{1}{\delta_s}\right) \geq \sum_{s=1}^{t-1} w \quad &\Rightarrow \quad \frac{1}{\delta_t} - \frac{1}{\delta_1} \geq (t-1)w \\
&\Rightarrow \quad \frac{1}{\delta_t} \geq (t-1)w \\
&\Rightarrow \quad f(x_t) - f(x^*) \leq \frac{2\beta||x_1 - x^*||^2}{t-1}
\end{aligned}$$

Como colofón veáse que $||x_s - x^*||$ es decreciente. En el lema 3.12 se llegaba a la desigualdad 2.1.9, $f(x) - f(y) \leq \nabla f(x)^T(x-y) - \frac{1}{2\beta}||\nabla f(x) - \nabla f(y)||^2$. Si en esta se toman $y = x^*$, $x = x_s$ se obtiene $0 \leq f(x_s) - f(x^*) \leq \nabla f(x_s)^T(x_s - x^*) - \frac{1}{2\beta}||\nabla f(x_s)||^2$ y en particular se cumple

$$\frac{-2}{\beta}\nabla f(x_s)^T(x_s - x^*) + \frac{1}{\beta^2}||\nabla f(x_s)||^2 \leq 0$$

Basta observar

$$\begin{aligned} ||x_{s+1} - x^*||^2 &= ||x_s - \frac{1}{\beta}\nabla f(x_s) - x^*||^2 \\ &= ||x_s - x^*||^2 - \frac{2}{\beta}\nabla f(x_s)^T(x_s - x^*) + \frac{1}{\beta^2}||\nabla f(x_s)||^2 \\ &\leq ||x_s - x^*||^2 \end{aligned}$$

□

Se comprueba de manera inmediata que $\frac{2\beta||x_1-x^*||^2}{\epsilon} + 1$ iteraciones bastan para alcanzar un ϵ-aproximante. Por lo tanto, si f es β-suave es posible garantizar una complejidad lineal superando la cuadrática que se tenía previamente. Nótese que en este caso cobra cierta importancia la elección del iterante inicial en la cota. Una buena elección puede acelerar la búsqueda del óptimo.

En la versión proyectada se tienen un lema y un teorema análogos a los anteriores, con las particularidades que repercuten de hallarse en un conjunto X restringido y de utilizar la proyección y el subgradiente en lugar del punto original y el gradiente.

Lema 2.15 Sea $X \subset \mathbb{R}^n$ convexo y sea $f : X \to \mathbb{R}$ convexa y β-suave. Sean $x, y \in X$, $x^+ := \Pi_X(x - \frac{1}{\beta}\nabla f(x))$ y $\xi_X(x) := \beta(x - x^+)$. Se cumple la siguiente desigualdad

$$f(x^+) - f(y) \leq \xi_X(x)^T(x - y) - \frac{1}{2\beta}||\xi_X(x)||^2$$

Dem: Para probarlo primero se observa que es válida la siguiente equivalencia y que además la segunda parte de esta se cumple por la segunda consecuencia del teorema 2.1 de la proyección convexa .

$$\nabla f(x)^T(x^+ - y) \leq \xi_X(x)^T(x^+ - y) \Leftrightarrow \left(x^+ - (x - \frac{1}{\beta}\nabla f(x))\right)^T (x^+ - y) \leq 0$$

Ahora basta proceder como en la condición de suficiencia del lema 2.13 utilizando la primera parte de la equivalencia, la condición β-suave de la función y (2.1.8).

$$\begin{aligned} f(x^+) - f(y) &= f(x^+) - f(x) + f(x) - f(y) \\ &\leq \nabla f(x)^T(x^+ - x) + \frac{\beta}{2}||x^+ - x||^2 + \nabla f(x)^T(x - y) \\ &= \nabla f(x)^T(x^+ - y) + \frac{\beta}{2}||x^+ - x||^2 \\ &\leq \xi_X(x)^T(x^+ - y) + \frac{1}{2\beta}||\xi_X(x)||^2 \\ &\leq \xi_X(x)^T(x - y) + \frac{1}{2\beta}||\xi_X(x)||^2 \end{aligned}$$

Esto último se debe a que $\xi_X(x)^T(x^+ - y) \leq \xi_X(x)(x - y)$ pues obviamente $0 \geq \xi_X(x)^T(x^+ - x) (= -\beta||x^+ - x||^2)$.

□

También de manera muy similar al teorema 2.14 *mutatis mutandis* se obtiene el siguiente resultado de convergencia.

Teorema 2.16 El método de descenso por gradiente proyectado satisface para una función β-suave con $\eta = \frac{1}{\beta}$

$$f(x_t) - f(x^*) \leq \frac{3\beta||x_1 - x^*||^2 + f(x_1) - f(x^*)}{t}$$

A pesar de que el método avanza más despacio que el descenso por gradiente tradicional debido a la ralentización que provoca la proyección convexa, mantenemos una complejidad de $O(\frac{1}{\epsilon})$. Tanto en la versión proyectada como en la original se ha logrado acelerar notoriamente el método aunque se considera la posibilidad de optimizarlo aún más con nuevas restricciones a la función objetivo.

2.1.2. Funciones fuertemente convexas

El otro concepto que va a acelerar la búsqueda del mínimo en los métodos derivados del descenso por gradiente es el de convexidad fuerte. Todas las funciones que aparezcan en esta subsección se supondrán diferenciables, si bien, sería posible omitir la diferenciabilidad de las funciones y ofrecer una versión de las definiciones y resultados utilizando el subgradiente en lugar del gradiente.

Definición 2.17 Sea $X \subset \mathbb{R}^n$ convexo, $f : X \to \mathbb{R}^n$ es α-fuertemente convexa si para algún $\alpha > 0$ satisface

$$f(x) - f(y) \leq \nabla f(x)^T(x - y) - \frac{\alpha}{2}||x - y||^2, \quad \forall x, y \in X \tag{2.1.11}$$

Nótese que esto supone una mejora de la cota dada por la definición de gradiente (o subgradiente).

Proposición 2.18 Sean $X \subset \mathbb{R}^n$ convexo y $f : X \to \mathbb{R}^n$, f α-fuertemente convexa $\Rightarrow$ f convexa.

Dem: De que f sea α-fuertemente convexa se deduce que en particular para todo $x \in X$ existe $\xi \in \mathbb{R}^n$ tal que $f(x) - f(y) \leq \xi^T(x - y)$ $y \in X$, es decir que para todo x existe un subgradiente ($\xi = \nabla f(x)$). La proposición 3.7 garantiza que f es convexa.

□

Proposición 2.19 Sean $X \subset \mathbb{R}^n$ convexo y $f : X \to \mathbb{R}^n$, f es α-fuertemente convexa $\Leftrightarrow$ la función $f(x) - \frac{\alpha}{2}||x||^2$ es convexa.

Dem: Sea $g : X \to \mathbb{R}^n$ definida por $g(x) = f(x) - \frac{\alpha}{2}||x||^2$

$\Rightarrow$ Basta tener en cuenta que g es diferenciable y por lo tanto para todo $x \in X$ existe un subgradiente (el propio gradiente $\nabla g(x)$).

$\Leftarrow$

$$\begin{aligned} f(x) - f(y) - \frac{\alpha}{2}(||x||^2 - ||y||^2) &= g(x) - g(y) \leq \nabla g(x)^T(x-y) \\ &= \nabla f(x)^T(x-y) - \alpha x^t(x-y) \end{aligned}$$

Se ha tenido en cuenta que $\nabla g(x) \in \partial g(x)$, por ser g convexa, y también $\nabla g(x) = \nabla f(x) - \alpha x$. Se deduce

$$f(x) - f(y) \leq \nabla f(x)^T(x-y) - \frac{\alpha}{2}||x||^2 - \frac{\alpha}{2}||y||^2 + \alpha x^T y = \nabla f(x)(x-y) - \frac{\alpha}{2}||x-y||^2$$

□

El valor de α de la definición resulta ser una medida de la curvatura de la función. Es conveniente que este valor sea lo más elevado posible ya que de esta manera si el iterante actual está alejado del minimizador global de la función el valor del gradiente será elevado en norma y por lo tanto también lo será el paso que demos en esa dirección en vistas de (2.1.11)

$$(2.1.11) \Rightarrow ||\nabla f(x)|| \geq \frac{f(x) - f(y)}{||x-y||} + \frac{\alpha}{2}||x-y||$$

Cabe destacar que así como la condición β-suave de una función garantiza la existencia de una función cuadrática

$$q_y^+(x) = f(y) + \nabla f(y)^T(x-y) + \frac{\beta}{2}||x-y||^2$$

que constituye una cota superior de f que pasa por $(y, f(y))$ (es decir, $q_y^+(x) \geq f(x)\ \forall x \in X$ y $q_y^+(y) = f(y)$); la condición α-fuertemente convexa hace lo propio con una función cuadrática

$$q_y^-(x) = f(y) + \nabla f(y)^T(x-y) + \frac{\alpha}{2}||x-y||^2$$

que constituye una cota inferior de f que pasa por $(y, f(y))$ (es decir, $q_y^-(x) \leq f(y)\ \forall x \in X$ y $q_y^-(y) = f(y)$), lo que se podría ver como una especie de dualidad entre las dos definiciones.

Añadiendo la convexidad fuerte a las funciones a optimizar se consigue una convergencia del mismo orden que en el Teorema 3.16 para el descenso por gradiente proyectado, como se apreciará en el siguiente resultado. Se hace uso de la desigualdad de Jensen para su demostración (ver, por ejemplo, [9]):

Dada una función convexa ϕ, una serie de puntos de su dominio, $x_1, \ldots, x_n$ y un mismo número de números reales $a_1, \ldots, a_n$ se cumple

$$\phi\left(\frac{\sum_{i=1}^n a_i x_i}{\sum_{i=1}^n a_i}\right) \leq \frac{\sum_{i=1}^n a_i \phi(x_i)}{\sum_{i=1}^n a_i}$$

Teorema 2.20 El método de descenso por gradiente proyectado bajo las hipótesis generales y para una función α-fuertemente convexa con $\eta_s = \frac{2}{\alpha(s+1)}$ satisface

$$f\left(\sum_{s=1}^t \frac{2s}{t(t+1)} x_s\right) - f(x^*) \leq \frac{2L^2}{\alpha(t+1)}$$

Dem: Se procede como en el teorema 2.9 teniendo en cuenta un sumando más, $-\frac{\alpha}{2}||x_s - x^*||^2$, que al ser negativo va a proporcionar una mejora de la cota. Nótese que con la hipótesis de que f es diferenciable se tiene que $\xi_s = \nabla f(x_s)$ y se puede aplicar (2.1.11) para añadir este nuevo término. Se parte por lo tanto de

$$f(x_s) - f(x^*) \leq \frac{\eta_s}{2}L^2 + (\frac{1}{2\eta_s} - \frac{\alpha}{2})||x_s - x^*||^2 - \frac{1}{2\eta_s}||x_{s+1} - x^*||^2$$

. Se aplica ahora la desigualdad de Jensen y se tienen en cuenta la expresión de η_s y que la suma de los primeros t naturales es $\frac{t(t+1)}{2}$.

$$\begin{aligned}
f(\sum_{s=1}^{t}\frac{2s}{t(t+1)}x_s) - f(x^*) &\leq \frac{2}{t(t+1)}\sum_{s=1}^{t}[s(f(x_s) - f(x^*))] \\
&= \frac{2}{t(t+1)}\sum_{s=1}^{t}[s\frac{L^2}{\alpha(s+1)} + \frac{\alpha}{4}(s(s-1)||x_s - x^*||^2 \\
&\quad -s(s+1)||x_{s+1} - x^*||^2)] \\
&\leq \frac{2L^2}{\alpha(t+1)} + \frac{\alpha}{4}\left(\sum_{s=1}^{t}[s^2(||x_s - x^*||^2 - ||x_{s+1} - x^*||^2)]\right. \\
&\quad \left. - \sum_{s=1}^{t}[s(||x_s - x^*||^2 + ||x_{s+1} - x^*||^2)]\right) \\
&\text{‘} \leq \frac{2L^2}{\alpha(t+1)} + \frac{\alpha}{4}(||x_1 - x^*||^2 - t(t+1)||x_{t+1} - x^*||^2 \\
&\quad -||x_1 - x^*||^2) \\
&\leq \frac{2L^2}{\alpha(t+1)}
\end{aligned}$$

□

Sin embargo, esta cota no logra alcanzar el $O(\ln\frac{1}{\epsilon})$ deseado y se queda de nuevo en algo del tipo $O(\frac{1}{\epsilon})$. Es posible, no obstante pedir que la función a optimizar sea simultáneamente β-suave y α-fuertemente convexa. En la siguiente subsección se prueban las cotas de convergencia para este caso.

2.1.3. Funciones β-suaves y α-fuertemente convexas

En la sección anterior se analizaba que el hecho de que una función sea suave implica la existencia de una función cuadrática que acota superiormente a esta primera coincidiendo con ella en un punto. Asímismo se afirmaba que con la condición de convexidad fuerte ocurre precisamente lo contrario: existe una función cuadrática que acota inferiormente a la dada y que coincide con ella en un punto. Por consiguiente contar con las nociones de suavidad y de convexidad fuerte simultáneamente no es algo que se dé naturalmente. Se trata de una condición bastante fuerte pues supone que la función se comporta básicamente como una función cuadrática acorralada

entre otras dos funciones cuadráticas. Cuanto más cercanos a cero son los valores de α y de β la función dispone de menor margen de movimiento. Se trata por tanto de una situación francamente favorable que se puede controlar sin dificultad.

El siguiente resultado se usará en la demostraciòn del lema 2.22.

Lema 2.21 Sea $f : X \subset \mathbb{R}^n \to \mathbb{R}$ β-suave. Entonces se cumple

$$(\nabla f(x) - \nabla f(y))^T (x - y) \geq \frac{1}{\beta} ||\nabla f(x) - \nabla f(y)||^2$$

Dem: Procediendo como en la demostración de la suficiencia del lema 2.13 se llega a (2.1.9)

$$f(x) - f(y) \leq \nabla f(x)^T (x - y) - \frac{1}{2\beta} ||\nabla f(x) - \nabla f(y)||^2$$

Invirtiendo los papeles de x e y también es cierto

$$f(y) - f(x) \leq \nabla f(y)^T (y - x) - \frac{1}{2\beta} ||\nabla f(x) - \nabla f(y)||^2$$

Sumando ambas expresiones se obtiene el resultado.

□

Lema 2.22 Sea $X \subset \mathbb{R}^n$ convexo. Una función $f : X \subset \mathbb{R}^n \to \mathbb{R}$ simultáneamente β-suave y α-fuertemente convexa cumple la siguiente desigualdad

$$(\nabla f(x) - \nabla f(y))^T (x - y) \geq \frac{\alpha\beta}{\beta + \alpha} ||x - y||^2 + \frac{1}{\beta + \alpha} ||\nabla f(x) - \nabla f(y)||^2$$

Dem: Sea $\phi(x) := f(x) - \frac{\alpha}{2}||x||^2$, convexa por la proposición 2.19
El hecho de que f sea β-suave nos permite hacer uso de (2.1.8) y teniendo en cuenta que $\nabla\phi(x) = \nabla f(x) - \alpha x$ se deduce que para todo $x, y \in X$

$$\phi(x) - \phi(y) - \nabla\phi(y)^T (x - y) \leq \frac{\beta}{2} ||x - y||^2 + \frac{\alpha}{2} ||y||^2 - \frac{\alpha}{2} ||x||^2 + \alpha y^T x - \alpha ||y||^2$$

La segunda parte de la desigualdad coincide con $(\frac{\beta}{2} - \frac{\alpha}{2})||x - y||^2$ con lo cual ϕ es $(\beta - \alpha)$-suave a efectos del lema 2.13. Ahora en virtud del lema inmediatemente anterior

$$(\nabla\phi(x) - \nabla\phi(y))^T (x - y) \geq \frac{1}{\beta - \alpha} ||\nabla\phi(x) - \nabla\phi(y)||^2$$

En términos de f esto equivale a

$$\begin{aligned} (\nabla f(x) - \nabla f(y))^T (x - y) - \alpha ||x - y||^2 \quad \geq \quad & \frac{1}{\beta - \alpha} ||\nabla f(x) - \nabla f(y)||^2 \\ & - \frac{2\alpha}{\beta - \alpha} (\nabla f(x) - \nabla f(y))^T (x - y) \\ & + \frac{\alpha^2}{\beta - \alpha} ||x - y||^2. \end{aligned}$$

Agrupando los términos se deduce el lema de manera sencilla.

□

Los próximos teoremas ofrecen una mejora importante en el orden de convergencia del descenso por gradiente y descenso por gradiente proyectado.

Teorema 2.23 El método de descenso por gradiente con $\eta = \frac{2}{\alpha+\beta}$ para una función β-suave y α-fuertemente convexa satisface

$$f(x_{t+1}) - f(x^*) \leq \frac{\beta}{2} \exp\left(\frac{-4t}{k+1}\right) ||x_1 - x^*||^2$$

donde k representa el cociente $\frac{\beta}{\alpha}$.

Dem:

$$\begin{aligned} ||x_{t+1} - x^*||^2 &= ||x_t - \eta \nabla f(x_t) - x^*||^2 = ||x_t - x^*||^2 - 2\eta \nabla f(x_t)^T (x_t - x^*) \\ &\quad + \eta^2 ||\nabla f(x_t)||^2 := A(x^*, x_t, \eta) \end{aligned}$$

Se aplica el lema 2.22 con $x = x_t$ e $y = x^*$ teniendo en cuenta que $\nabla f(x^*) = 0$ por ser un extremo relativo de f.

$$\nabla f(x_t)^T (x_t - x^*) \geq \frac{\alpha\beta}{\beta+\alpha} ||x_t - x^*||^2 + \frac{1}{\beta+\alpha} ||\nabla f(x_t)||^2$$

La desigualdad anterior es equivalente a esta otra

$$\frac{1}{2\eta} A(x^*, x_t, \eta) + \left(\frac{\alpha\beta}{\beta+\alpha} - \frac{1}{2\eta}\right) ||x_t - x^*||^2 + \left(\frac{1}{\beta+\alpha} - \frac{\eta}{2}\right) ||\nabla f(x_t)||^2 \leq 0$$

de la cual, dado que $\eta = \frac{2}{\alpha+\beta}$, se deduce una cota superior para $A(x, x_t, \eta) (= ||x_{t+1} - x^*||^2)$.

$$\begin{aligned} ||x_{t+1} - x^*||^2 &\leq (1 - \frac{4\alpha\beta}{(\alpha+\beta)^2}) ||x_t - x^*||^2 \\ &= (\frac{k-1}{k+1})^2 ||x_t - x^*||^2 \\ &= (1 - \frac{2}{k+1})^{2t} ||x_1 - x^*||^2 \\ &\leq \exp(-\frac{4t}{k+1}) ||x_1 - x^*||^2 \end{aligned}$$

Para el último paso se ha tenido en cuenta la desigualdad $(1 - x) \leq \exp -x, \;\; x > 0$. Por último, como f es β-suave, por el lema 2.13

$$f(x_{t+1}) - f(x^*) + \nabla f(x^*)^T (x_{t+1} - x^*) \leq \frac{\beta}{2} ||x_{t+1} - x^*||^2$$

. $\nabla f(x^*) = 0$ y basta sustituir $||x_{t+1} - x^*||^2$ por su cota.

□

El número $k = \frac{\beta}{\alpha}$ empleado en el teorema recibe el nombre de *número de condición*. Para que la convergencia sea lo más rápida posible interesa que este número esté próximo a cero.

De la expresión anterior se deduce que bastan $t \geq \frac{k+1}{4} \ln(\frac{\beta}{2\epsilon})$ iteraciones para alcanzar un ϵ-aproximante, es decir, se llega al fin a una complejidad de $\ln(k\frac{1}{\epsilon})$. Esta mejora es francamente

buena, pues supone una convergencia exponencial hacia el mínimo de la función. Asímismo merece la pena destacar que la longitud de paso elegida para llegar a esta cota es constante en cada iteración y depende exclusivamente del grado de suavidad y de convexidad fuerte de la función objetivo.

Para el descenso por gradiente proyectado se alcanzará también una cota de convergencia exponencial, si bien esta cota se da en términos del minimizador de la función objetivo y no del mínimo. Esto puede ser beneficioso o no en función de lo que se desee aproximar. En cualquier caso en lo que concierne a la cota en sí misma, la diferencia principal con la dada por el teorema 2.24 es el factor -4 que multiplica al número de iteraciones t dentro de la función exponencial. En la versión proyectada perdemos este factor que puede acelerar sensiblemente la convergencia.

Teorema 2.24 El método de descenso por gradiente proyectado con $\eta = \frac{1}{\beta}$ para una función smooth y fuertemente convexa satisfce

$$||x_{t+1} - x^*|| \leq \exp\left(\frac{-t}{k}\right) \ ||x_1 - x^*||^2$$

Dem: Añadiendo al lema 2.15 la condición de convexidad fuerte (2.1.11) y manteniendo la notación de entonces se obtiene la desigualdad

$$f(x^+) - f(y) \leq \xi_X(x)^T(x - y) - \frac{1}{2\beta}||\xi_X(x)||^2 - \frac{\alpha}{2}||x - y||^2$$

En particular para $y = x^*$ se tiene

$$f(x^+) - f(x^*) \leq \xi_X(x)(x - x^*) - \frac{1}{2\beta}||\xi_X(x)||^2 - \frac{\alpha}{2}||x - x^*||^2$$

De las igualdades $x_{t+1} = x_t^+$ y $\frac{1}{\beta}\xi_X(x_t) = x_t - x_t^+$, de la desigualdad anterior y de $f(x^*) - f(x_{t+1}) \leq 0$ respectivamente se deduce

$$\begin{aligned}
||x_{t+1} - x^*||^2 &\leq ||x_t^+ - x^*||^2 = ||x_t - \frac{1}{\beta}\xi_X(x_t) - x^*||^2 \\
&= ||x_t - x^*||^2 - \frac{2}{\beta}\xi_X(x_t)(x_t - x^*) + \frac{1}{\beta^2}||\xi_X(x_t)||^2 \\
&= ||x_t - x^*||^2 - \frac{2}{\beta}(\xi_X(x_t)^T(x_t - x^*) - \frac{1}{2\beta}||\xi_X(x_t)||^2) \\
&\leq ||x_t - x^*||^2 + \frac{2}{\beta}(f(x^*) - f(x_{t+1}) - \frac{\alpha}{2}||x_t - x^*||^2) \\
&\leq \left(1 - \frac{1}{k}\right)||x_t - x^*||^2 \\
&\leq \left(1 - \frac{1}{k}\right)^t ||x_1 - x^*||^2 \\
&\leq \exp\left(-\frac{t}{k}\right)^t ||x_1 - x^*||^2
\end{aligned}$$

□

Se retoma aquí el **Ejemplo 2.11**, en el cual se procedía a optimizar la función $f(x) = ||x||^2$ tanto en todo el espacio $\mathbb{R}^n$ como en un subespacio restringido $X = \bar{B}((0,2),1)$ donde, como se ha visto, no supone demasiado esfuerzo el proceso de proyectar los puntos exteriores al conjunto en cuestión. Dado $\nabla f(x) = 2x$ se puede comprobar que f es β-suave con $\beta = 2$. No obstante, a la hora de aplicar los métodos se evitará usar esta información, ya que para una longitud de paso $\eta = \frac{1}{\beta}$ (la exigida en los teoremas de convergencia de la subsección 2.1.2) se llegaría al verdadero mínimo inmediatamente tras la primera iteración. Esto se debe al excelente comportamiento que ofrecen las funciones cuadráticas a la hora de llevar a cabo este tipo de optimización.

$$\begin{aligned} f(x) &= a||x||^2 + bx^T\vec{1} + c \\ \nabla f(x) = 2ax + b\vec{1} &\Rightarrow x^* = -\frac{b}{2a}\vec{1} \\ ||\nabla f(x) - \nabla f(y)|| &= 2a||x-y|| \end{aligned}$$

Tomando entonces $\eta = \frac{1}{\beta} = \frac{1}{2a}$ se tiene en 2.1.1 para la primera iteración $x_2 = x_1 - (x_1 + \frac{b}{2a}\vec{1}) = x^*$.

Se comprueba fácilmente que $f(x) = ||x||^2$ es α-fuertemente convexa con $\alpha = 1$

$$f(x) - f(y) - \nabla f(x)(x-y)^T = ||x||^2 - ||y||^2 - 2x^T(x-y) = -2||x-y||^2$$

Se procede a implementar el descenso por gradiente (DG) y el descenso por gradiente proyectado (DGP) para cuatro casos distintos donde en cada uno de los cuales se suponen conocidas determinadas condiciones de regularidad de la función. No obstante, el ejemplo elegido corresponde con una función ideal para analizar teóricamente cómo funcionan estos métodos pero que carece de todo interés práctico dado que la expresión de $\nabla f(x)$ es fácilmente calculable. Lo que sí es frecuente cuando uno se enfrenta a la optimización de una función arbitraria es que no sea posible precisar los valores de las constantes de regularidad L, α y β, con lo que es necesario estimarlas de alguna manera. Por ello cada uno de los casos se ejecutará varias veces. En la primera ocasión suponiendo conocidas las verdaderas constantes de regularidad, salvo la de suavidad que se tomará $\beta = 3$ por las razones que se han explícitado. En una segunda ocasión se supondrá que de alguna manera se han estimado unas constantes de regularidad correctas (es decir, con tales constantes se cumplen las definiciones de L-lipschtz, β-suave y α-fuertemente convexa de f), pero ciertamente alejadas de su valor óptimo. Se tomarán,como ejemplo, L= 20, $\beta = 5$ y $\alpha = \frac{1}{2}$. Por último se considerarán unas estimaciones erróneas que dan lugar a unos valores L= 3, $\beta = 1$ y $\alpha = 2$. De acuerdo con estos valores para las constantes de regularidad se elegirán las longitudes de paso oportunas según los teoremas de convergencia expuestos y se estudiará la distancia entre el aproximante que se especifica en el teorema y el valor que se trata de aproximar. Nótese que lo usual es pensar en la distancia entre la imagen por f del último iterante y el verdadero valor del mínimo de la función objetivo, pero en muchos de los teoremas el aproximante que se toma es un punto promedio de todos los iterantes, e incluso en el teorema 2.24 se estudia la distancia entre el último iterante y el minimizador de f. Para este se ha tomado la distancia entre la imagen del último iterante y el mínimo, a fin de llevar a cabo una comparación justa con el resto de casos.

La información sobre la regularidad de f conocida en cada caso viene especificada al pie de la tabla. El valor t corresponde al número de iteraciones que se han llevado a cabo. Los valores $\hat{\eta_s}$ y η_s corresponden respectivamente a la longitud de paso dada en las hipótesis del teorema en cuestión a partir de los verdaderos valores de las constantes de regularidad (salvo β=3) y la

longitud de paso utilizada a la hora de llevar a cabo el método según las estimaciones que se hayan realizado. $d(\epsilon_{\text{aprox}}, v_{\text{exact}})$ indica la distancia entre el aproximante y el verdadero valor que este aproxima.

Francamente en la práctica serán pocos los casos en los que el funcionamiento de DG y DGP sean

	MÉTODO	t	$\hat{\eta_s}$	η_s	$d(\epsilon_{\text{aprox}}, v_{\text{exact}})$
0	DGP	100	$\frac{R}{L\sqrt{t}} = 0.025$	0.025	0.0529
	DGP	100	$\frac{R}{L\sqrt{t}} = 0.025$	$\frac{1}{\sqrt{s}}$	0.0101
	DGP	100	$\frac{R}{L\sqrt{t}} = 0.025$	$\frac{1.5}{20 \cdot 10} = 0.0075$	0.1978
	DGP	100	$\frac{R}{L\sqrt{t}} = 0.025$	$\frac{1.5}{3 \cdot 10} = 0.05$	0.0275
1	DG	10	$\frac{1}{\beta} = \frac{1}{3}$	$\frac{1}{3}$	$6.4529 \cdot 10^{-9}$
	DG	10	$\frac{1}{\beta} = \frac{1}{2}$	$\frac{1}{5}$	$2.539 \cdot 10^{-4}$
	DG	10	$\frac{1}{\beta} = \frac{1}{2}$	1	2.5
	DGP	10	$\frac{1}{\beta} = \frac{1}{2}$	$\frac{1}{3}$	$1.6009 \cdot 10^{-13}$
	DGP	10	$\frac{1}{\beta} = \frac{1}{2}$	$\frac{1}{5}$	$1.8321 \cdot 10^{-7}$
	DGP	10	$\frac{1}{\beta} = \frac{1}{2}$	1	$9.1998 \cdot 10^{-10}$
2	DG	10	$\frac{\alpha}{s+1} = \frac{2}{s+1}$	$\frac{2}{s+1}$	0.0185
	DG	10	$\frac{\alpha}{s+1} = \frac{2}{s+1}$	$\frac{4}{(s+1)}$	0.0197
	DG	10	$\frac{\alpha}{s+1} = \frac{2}{s+1}$	$\frac{1}{(s+1)}$	0.0192
3	DG	5	$\frac{2}{\alpha+\beta} = \frac{2}{3}$	$\frac{2}{1+2} = \frac{1}{3}$	$6.4529 \cdot 10^{-9}$
	DG	5	$\frac{2}{\alpha+\beta} = \frac{2}{3}$	$\frac{2}{0.5+5} = 0.363$	$1.742 \cdot 10^{-10}$
	DG	5	$\frac{2}{\alpha+\beta} = \frac{2}{3}$	$\frac{2}{2+1} = \frac{2}{3}$	$6.4529 \cdot 10^{-9}$
	DGP	5	$\frac{1}{\beta} = \frac{1}{2}$	$\frac{1}{3}$	$1.6009 \cdot 10^{-13}$
	DGP	5	$\frac{1}{\beta} = \frac{1}{2}$	$\frac{1}{5}$	$1.8321 \cdot 10^{-7}$
	DGP	5	$\frac{1}{\beta} = \frac{1}{2}$	1	$9.1998 \cdot 10^{-10}$

Cuadro 2.1: Métodos de descenso por gradiente para $f(x) = ||x||^2$
0: f convexa y L-lipschitziana
1: f convexa y β-suave
2: f α-fuertemente convexa y L-lipschitziana
3: f α-fuertemente convexa y β-suave

tan eficientes como lo son para esta función. Basta observar que en el caso 3 se han llevado a cabo tan solo 5 iteraciones para obtener un aproximante prácticamente idéntico al verdadero valor. Se ha mencionado ya que si una función es simultáneamente β-suave y α-fuertemente convexa entonces está acotada superiormente por una función cuadrática con la misma constante de suavidad, coincidiendo ambas en un punto; e inferiormente por una función cuadrática con la misma constante de convexidad fuerte, coincidiendo también ambas en un punto. Por lo tanto, las condiciones de regularidad exigidas en el caso 3 se dan casi exclusivamente en las funciones cuadráticas, o en funciones cuyo comportamiento se prácticamente el de estas

Es curioso que en todos los ejemplos recogidos en 2.1 el método de descenso por gradiente proyectado ha alcanzado un ϵ-aproximante más cercano al valor real que el ϵ-aproximante obtenido por el descenso por gradiente clásico. Esto pone de manifiesto que el hecho de que las cotas proporcionadas por los teoremas de convergencia sean más ajustadas para el

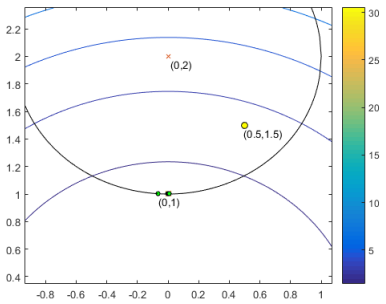


Figura 2.4: Descenso por gradiente para $f(x) = ||x||^2$ con $\eta = \frac{2}{3}$

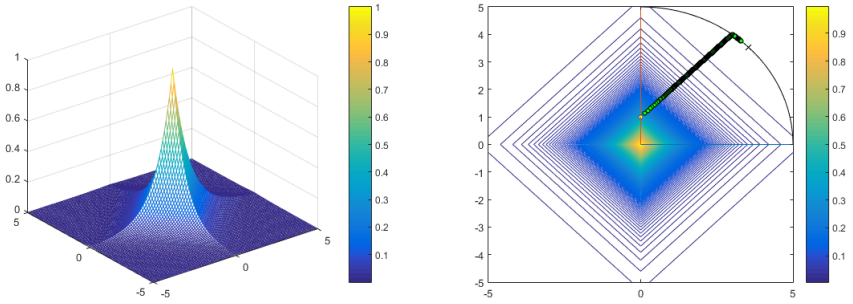
método de descenso por gradiente clásico no quiere decir que este finalmente obtenga un ϵ-aproximante asímismo más ajustado. Puede ocurrir, como en este caso, que la proyección aporte un acercamiento extra hacia el mínimo de la función. Si bien el cálculo de esta proyección siempre acarreará un costo computacional adicional que no siempre es asumible.

En cuanto a los resultados obtenidos para cada 3-upla de valores de las constantes de regularidad, se puede observar que la distancia entre aproximante y valor real es muy sensible frente a cambios en el valor de β, mientras estimaciones desproporcionadas de α no afectan tanto a la convergencia del método. En el caso 1, dando por conocido que f es β-suave con valor estimado $\beta = 1$ el método nunca convergerá. Analizando (2.1.1) ocurre que para cada iteración $s \geq 1$, $x_{s+1} = x_s \cdot (1 - 2\eta) = -x_s$. Por lo tanto, los iterantes oscilan entre x_1 y $-x_1$ y la distancia 2.5 corresponde finalmente a $\mathrm{d}(f(x_1),f(\vec{0}))$. En el caso 2, las diferentes estimaciones de α dan lugar a distancias de aproximación muy similares. Esto puede deberse a que el paso idóneo en estos métodos no es tan solo dependiente de α, sino también de la iteración actual s, mientras que el paso idóneo en el caso 1 es estrictamente dependiente de β.

Ejemplo 2.25 Antes de introducir la siguiente sección veáse un ejemplo en el que los métodos de descenso por gradiente no resuelven tan eficazmente el problema de optimización. Dada una función convexa β-suave en $X \subset \mathbb{R}^n$ se aplica el método de descenso por gradiente proyectado con el objetivo de encontrar un minimizador de f. Por lo tanto, haciendo uso de los resultados teóricos se observa que si f estuviera dotada además de la condición de convexidad fuerte la convergencia sería muy rápida siempre y cuando se tomara una longitud de paso igual a $\frac{1}{\beta}$. Se supone que esta hipótesis, sin embargo, no se tiene. El teorema 3.17 sobre descenso por gradiente proyectado para funciones β-suaves no proporciona a priori una cota para el minimizador por lo que la situación no está controlada como en otros casos.

Veáse qué sucede cuando se procede a optimizar la función $f(x) = \exp(-||x||_1)$ en el conjunto convexo $X = \{x = (x_1, x_2) \in \mathbb{R}^2 : x_1 \geq 0, x_2 \geq 0\} \cap \bar{B}(\vec{0}, 5)$. f es diferenciable en el interior de este conjunto; alcanza su mínimo en el punto $x^* = (\frac{\sqrt{2}\cdot 5}{2}, \frac{\sqrt{2}\cdot 5}{2})$; es convexa, se puede observar en Figura 2.5 y se justificará a continuación; y además es $\beta-$suave con $\beta = 2\sqrt{2}$, como también se justificará.

Si f es convexa en $\{x = (x_1, x_2) \in \mathbb{R}^2 : x_1 \geq 0, x_2 \geq 0\}$ también lo será en X. El hecho de que f sea diferenciable en el interior de este conjunto garantiza que $\partial f(x) = \{\nabla f(x)\} \neq \emptyset$



(a) $f(x) = \exp(-||x||_1)$ (b) DGP $\eta = \frac{1}{2\sqrt{2}}$

Figura 2.5: Descenso por gradiente proyectado para $f(x) = \exp -||x||_1$ con $\eta = \frac{1}{2\sqrt{2}}$ en $X = \{x = (x_1, x_2) \in \mathbb{R}^2 : x_1 \geq 0, x_2 \geq 0\} \cap \bar{B}(\vec{0}, 5)$

para todo punto del interior. Si x, en cambio pertenece a la frontera, también se deduce la existencia de subgradientes a partir de la convexidad de la función exponencial en el caso real. Sin pérdida de generalidad se prueba la existencia de subgradientes para un punto cualquiera de $\{x = (0, x_2) : x_2 \geq 0\}$. El caso $\{x = (x_1, 0) : x_1 \geq 0\}$ es completamente análogo. Sean $x = (0, x_2), y = (y_1, y_2)$,

$$f(x) - f(y) = e^{-x_2} - e^{-y_1-y_2} \leq -e^{-x_2}(x_2 - y_1 - y_2) = (e^{-x_2}, -e^{-x_2})^T(x - y)$$

luego $(e^{-x_2}, -e^{-x_2}) \in \partial f(x)$ y se puede concluir por la proposición 3.7 que f es convexa en X dado que existen subgradientes en cada uno de sus puntos.

Se hace uso de nuevo de la convexidad de la función exponencial en el caso real, de la definición 3.2 de subgradiente y de la proposición 3.7 para deducir que $|e^x - e^y| \leq e^x|x - y|$ y probar la suavidad de f. Dados $x = (x_1, x_2), y = (y_1, y_2) \in X$

$$\begin{aligned} ||\nabla f(x) - \nabla f(y)||_2 &\leq ||\nabla f(x) - \nabla f(y)||_1 = 2|f(y) - f(x)| \\ &\leq 2\exp(-y_1 - y_2)|x_1 - y_1 + x_2 - y_2| \leq 2 \max_{x \in B(0,5)} \exp(-x_1 - x_2)||x - y||_1 \\ &\leq 2\sqrt{2}||x - y||_2 \end{aligned}$$

Tomando por lo tanto una longitud de paso $\eta = \frac{1}{2\sqrt{2}}$ el método de descenso por gradiente proyectado encuentra tras 10000 iteraciones un punto cuya imagen dista $1.3765 \cdot 10^{-5}$ del mínimo de f. Esto no es nada extraordinario dado el elevado número de iteraciones que se han llevado a cabo y lo peor es que si el interés se centra en el minimizador, este aún se encuentra a una distancia de 0.3372 respecto al valor real. La razón está en que el valor del módulo del gradiente se hace excesivamente pequeño tras pocas iteraciones con lo que, para una longitud de paso fija, el método avanza mucho más despacio de lo deseado. Además, se observa en la figura 2.5, no lo hace en la dirección óptima hasta que empieza a actuar la proyección, dado que en el espacio ambiente $\mathbb{R}^2$ la función diverge hacia $-\infty$ en cualquier dirección.

2.2. El método de Frank-Wolfe

Es satisfactorio el logro de una complejidad $O(\ln \frac{1}{\epsilon})$ a la hora de minimizar una función en un subconjunto de $\mathbb{R}^n$, sin embargo el problema de calcular la proyección convexa en cada una de las iteraciones puede resultar verdaderamente costoso. El método que se va a exponer a continuación sustituye esta proyección convexa por un optimización lineal obteniendo un orden de convergencia de $O(\frac{1}{\epsilon})$. La conveniencia de este nuevo algoritmo estará supeditada a los casos en los que la optimización lineal sea más sencilla que la optimización convexa haciendo finalmente útil la pérdida en el orden de convergencia.

Sea $X \subset \mathbb{R}^n$ compacto y convexo y sea $f : X \to \mathbb{R}$. Sean también $(\phi_s)_{s\geq 1}$ una secuencia fija de números reales y $x_1 \in \mathbb{R}^n$ un punto inicial. El método de descenso por gradiente condicional realiza la siguiente iteración para $t \geq 1$.

$$y_t \in argmin_{y\in X} \nabla f(x_t)^T y \tag{2.2.1}$$
$$x_{t+1} = (1-\phi_t)x_t + \phi_t y_t \tag{2.2.2}$$

La razón que hay detrás de (2.2.1) es exactamente la misma que había en (2.1.2) para el descenso por gradiente: y_t es la dirección de máximo descenso para el polinomio de Taylor de f en cada iteración. Basta observar que minimiza la siguiente expresión por su propia definición.

$$\begin{aligned} f(x_{t+1}) - f(x_t) &= f((1-\phi_t)x_t + \phi_t y_t) - f(x_t) \\ &= \nabla f(x_t)\phi_t(x_t - y_t) + \mathrm{O}(\phi_t^2 ||y_t - x_t||^2) \end{aligned}$$

Algunas suposiciones son necesarias. En primer lugar ya se ha dado por supuesta la existencia de $\nabla f(x_t)$ para cada $t \geq 1$, lo cual implica que se admitirá que f sea diferenciable. Se considera también que el conjunto X es acotado, en particular se denotará por $R := \sup_{x,y\in X} ||x-y||$ el diámetro de X. Por último se redefinirá el concepto de suavidad de una función a partir de la norma dual.

Definición 2.26 Sea $||\cdot||$ una norma cualquiera. Se define la norma dual $||\cdot||_*$ como $||g||_* = \sup_{\substack{x\in\mathbb{R}^n \\ ||x||\leq 1}} g^T x, \quad \forall\, g \in \mathbb{R}^n$.

Si $||\cdot|| = ||\cdot||_2$ entonces $||\cdot|| = ||\cdot||_*$ pues debido a la desigualdad de Cauchy-Schwarz $||g||_2 = \sup_{\substack{x\in\mathbb{R}^n \\ ||x||\leq 1}} g^T x$.

Definición 2.27 Sea $f : X \subset \mathbb{R}^n \to \mathbb{R}$ diferenciable. f es β-suave si para todo $x, y \in \mathbb{R}^n$

$$||\nabla f(x) - \nabla f(y)||_* \leq \beta ||x - y||$$

Esta definición no es más que una generalización de la dada en las secciones anteriores, (pues si $||\cdot|| = ||\cdot||_2$ ambas definiciones coinciden) que es útil para adaptar el concepto de suavidad a cualquier norma más allá de la euclídea.

Teorema 2.28 El descenso por gradiente condicional con $\phi_s = \frac{2}{s+1}$ para funciones convexas y β-suaves satisface

$$f(x_t) - f(x^*) \leq \frac{2\beta R^2}{t+1}$$

Dem: La definicición de suavidad se cumple para $||.||$ arbitraria. Entonces usando (2.1.8), $x_{s+1} - x_s = \phi_s(y_s - x_s)$, $y_s \in argmin_{y\in X} \nabla f(x)^T y$ y $\nabla f(x_s) \in \partial f(x_s)$,en este orden, se llega a la siguiente cadena de desigualdades

$$\begin{aligned} f(x_{s+1}) - f(x_s) &\leq \nabla f(x_s)^T(x_{s+1} - x_s) + \frac{\beta}{2}||x_{s+1} - x_s||^2 \\ &\leq \phi_s \nabla f(x_s)^T(y_s - x_s) + \frac{\beta}{2}\phi_s^2 R^2 \\ &\leq \phi_s \nabla f(x_s)^T(x^* - x_s) + \frac{\beta}{2}\phi_s^2 R^2 \\ &\leq \phi_s(f(x^*) - f(x_s)) + \frac{\beta}{2}\phi_s^2 R^2 \end{aligned}$$

Sea $\delta_s := f(x_s) - f(x^*)$. Lo deducido antes equivale a lo siguiente

$$\begin{aligned} \delta_{s+1} &\leq (1 - \phi_s)\delta_s + \frac{\beta}{2}\phi_s^2 R^2 \\ &\leq (1 - \frac{2}{s+1})\delta_s + \frac{\beta}{2}(\frac{2}{s+1})^2 R^2 \end{aligned}$$

El resultado se concluye probando por inducción $\delta_s \leq \frac{\beta}{s}R^2, \ \forall s \geq 2$. Para $s = 2$ es cierto por (2.1.8) y el hecho de que $\nabla f(x^*) = 0$. Suponiendo como hipótesis de inducción que también lo es para s se prueba para $s + 1$

$$\begin{aligned} \delta_{s+1} &\leq (1 - \frac{2}{s+1})\frac{\beta R^2}{s} + \frac{\beta}{2}\frac{4}{(s+1)^2}R^2 \\ &= \frac{(s^2-1)\beta R^2 + 2s\beta R^2}{s(s+1)^2} = \frac{(s-1)^2\beta R^2 - 2\beta R^2}{s(s+1)^2} \\ &\leq \frac{(s-1)^2\beta R^2}{s(s+1)^2} \leq \frac{\beta R^2}{s+1} \end{aligned}$$

□

En suma a la libertad en la elección de la norma y al desapego de la proyección convexa, existe otra propiedad que ofrece el método de Frank Wolfe. Bajo la suposición de que el conjunto de restricciones X es un politopo (y por ende convexo) e iniciando el algoritmo desde uno de sus vértices se llega a una representación dispersa de los iterantes en función de los vértices del politopo. Basta observar (2.2.1) y (2.2.2). La optimización lineal dada en el primer paso del método conduce a un vértice del politopo v_{t+1}, con lo que el iterante $t+1$ es combinación convexa de este nuevo vértice y del iterante x_t. Si, por hipótesis de inducción, x_t es combinación convexa de t vértices del polítopo $x_t = \sum_{i=1}^t \lambda_i v_i$ (con $\sum_{i=1}^t \lambda_i = 1$) , entonces en nuevo iterante $t + 1$ lo será de $t + 1$ vértices.

$$x_{t+1} = (1 - \phi_t)\sum_{i=1}^t \lambda_i v_i + \phi_t v_{t+1} \text{ con } (1 - \phi_t)\sum_{i=1}^t \lambda_i + \phi_t = 1$$

En algunos casos la propiedad de dispersión puede resultar verdaderamente conveniente. Imagínese que el conjunto A es el símplice dado por la restricción $||x||_1 \leq 1$ (se podría incluso generalizar y hablar de $||x||_1 \leq k$ para cualquier $k > 0$). Los vértices correspondientes son exactamente los vectores de la base canónica, luego el iterante x_t se escribirá como combinación lineal convexa de tan solo t vectores de la base canónica, es decir, constará de a lo sumo t coordenadas no nulas que además suman 1. Si iteramos el algoritmo un número de veces claramente inferior a la dimensión del espacio ambiente los iterantes tendrán un número considerable de coordenadas no nulas. Este hecho, unido a que el orden de convergencia es independiente de la dimensión del espacio ambiente, puede reducir notoriamente el costo computacional del algoritmo. Por ejemplo en el problema LASSO. Formulado en la forma no Lagrangiana según [5].

$$\min_{\beta \in \mathbb{R}^p} \left\{ \frac{1}{2n} ||y - X\beta||_2^2 + \lambda ||\beta||_1 \right\}$$

donde $X \in \mathbb{R}^{n \times p}$ e $y \in \mathbb{R}^n$ son conocidos.

El inconveniente principal de esta forma de escribir el problema es que $g : \mathbb{R}^p \to \mathbb{R}$ definifa por $g(\beta) = \frac{1}{2n} ||y - X\beta||_2^2 + \lambda ||\beta||_1$ no es β-suave para ningún valor de β. En todo caso es suma de un función β-suave $g_1(\beta) = \frac{1}{2n} ||y - X\beta||_2^2$ y de otro término $g_2(\beta)$. No obstante, existe una analogía entre la forma langrangiana y la forma restringida del problema LASSO.

$$\min_{\beta \in \mathbb{R}^p} \left\{ \frac{1}{2n} ||y - X\beta||_2^2 \right\} \text{ sujeto a } ||\beta||_1 \leq s$$

Se puede suponer además $s = 1$ pues bastaría reescalar la solución obtenida para un s arbitrario. En esta forma la función objetivo sería $f : A \to \mathbb{R}$, $f(\beta) = \frac{1}{2n} ||y - X\beta||_2^2$ donde A es el símplice dado por la restricción $||\beta||_1 \leq 1$. Es decir, el término g_1 anterior restringido al conjunto A y por lo tanto se está llevando a cabo la optimización de una función convexa y suave para algún valor de β.

La dirección de descenso estará dada por $\nabla f(\beta) = X^T(X\beta - y)$. Así pues, si para la iteración $t \geq 1$ se supone calculado $z_t = X\beta_t - y$, (2.2.1) es equivalente a $\max_{i \in 1,..,p} |[\nabla f(\beta_t)](i)|$ que a su vez es equivalente a $\max_{i \in 1,..,p} |x_i z_t|$ (donde x_i es la columna i-ésima de X) gracias a la estructura del conjunto de restricciones.

Como ya se ha mencionado, la clave del método de Frank Wolfe radica en que la optimización lineal realizada en (2.2.1) sea asumiblemente calculable. En este caso se va a conceder la hipótesis de que el problema $\max_{i \in 1,..,p} |x_i y|$ se puede llevar a cabo en $\mathrm{O}(\mathrm{P}(p))$ operaciones para cualquier $y \in \mathbb{R}^n$, es decir, el costo operacional es polinomial respecto a la dimensión p.

Asumida esta hipótesis ahora entra en juego la dispersión de los iterantes. En (2.2.2) se obtiene x_{t+1} tras $\mathrm{O}(t)$ operaciones gracias a que β_t es combinación de t vértices y por lo tanto de t vectores unitarios que forman la base canónica y el minimizador v_{t+1} de (2.2.1) pertenece a la base canónica. Por último, para obtener z_{t+1} solo nos queda operar Xv_{t+1}. Al tratarse de un vector de la base canónica esto conlleva tan solo $\mathrm{O}(p)$ operaciones. En resumen, tras t iteraciones se habrán realizado $\mathrm{O}(t\mathrm{P}(p) + t^2)$ operaciones.

Por otra parte se puede probar que f es β-suave y así utilizar el teorema de convergencia 2.26. Sea $m := \max_{1 \leq i \leq p} ||x_i||_2$. Dado que $||\nabla f(y) - \nabla f(z)||_* = ||\nabla f(y) - \nabla f(z)||_2 \leq ||\nabla f(y) - \nabla f(z)||_\infty$

y dada la desiguladad

$$\begin{aligned} ||\nabla f(y) - \nabla f(z)||_\infty &= ||X^T X(y-z)||_\infty = \max_{1\leq i\leq p} \left| x_i^T \left(\sum_{j=1}^{p} x_j (y(j) - z(j)) \right) \right| \\ &\leq m^2 \sum_{j=1}^{p} |y(j) - z(j)| = m^2 ||y - z||_1 \end{aligned}$$

f es β-suave con $\beta = m^2$. Obsérvese que la definición de suavidad se cumple para la norma 1, luego la libertad de norma que ofrece el método de Frank Wolfe también hace su aparición para este ejemplo. Así pues el teorema 2.28 proporciona un orden de convergencia

$$f(x_t) - f(x^*) \leq \frac{8m^2}{t+1}$$

Usando los cálculos anteriores esto equivale a afirmar que es posible obtener un aproximante a distancia menor que ϵ del verdadero mínimo de la función objetivo con un costo computacional de $\mathrm{O}\left(m^2 \frac{\mathrm{P}(p)}{\epsilon} + \frac{m^4}{\epsilon^2}\right)$ operaciones.

2.3. Cotas inferiores

Lo que se ha hecho hasta ahora una vez descritos los diversos métodos de aproximación es pedir ciertas restricciones a la función objetivo para garantizar que un número determinado de iteraciones es suficiente para obtener un ϵ-aproximante. Se dará ahora la vuelta a la situación. Imagínese que se quiere garantizar la existencia de una función con ciertas restricciones tal que para cualquier método de aproximación del tipo black-box se precisa de un determinado número de iteraciones mínimo para alcanzar un ϵ-aproximante. Por ende este número constituye una cota inferior para el número de iteraciones justificando así el título de la sección.

[4] dedica su sección 3.5 a ofrecer una serie de resultados en este sentido. Se hará simplemente esta referencia rápida en este texto por el hecho de no reproducir exactamente lo que ya se ha escrito en [4] alargando innecesariamente el trabajo. Primero se prueban cotas para funciones convexas, o α-fuertemente convexas, y L-lipschitzianas. Posteriormente para funciones también convexas, o fuertemente convexas, se incluye la condición de β-suavidad y se ve como en los primeros casos las cotas superiores proporcionadas anteriormente coinciden con las cotas inferiores que se dan ahora, mientras que para funciones β-suaves existe un margen entre ambas, lo cual incita a plantearse la posibilidad de desarrollar métodos de tipo black-box, más eficientes, que sean capaces de cerrar este hueco. En futuras secciones se verá que efectivamente existen métodos con estas características.

En las secciones siguientes se habla de dos métodos de tipo black-box (descenso geométrico y descenso por gradiente acelerado). Ambos surgen del afán por cerrar este margen que existe entre las cotas inferiores de esta sección y las cotas superiores dadas en la anterior cuando se habla de funciones β-suaves y α-fuertemente convexas simultáneamente, dando lugar a algoritmos con una complejidad óptima dentro de los métodos de tipo black-box con unas determinadas exigencias de regularidad sobre la función objetivo. Ambos algoritmos conseguirán alcanzar una

complejidad del orden de $O(\sqrt{k}\frac{1}{ln(\epsilon)})$, donde $k = \frac{\alpha}{\beta}$ es el número de condición de la función f. Esta es la cota inferior proporcionada para algoritmos black-box con las condiciones de suavidad y de convexidad. En la sección anterior se veía como el descenso por gradiente y su versión proyectada ofrecían, sin embargo, un orden de convergencia un tanto más lento, $O(k\frac{1}{ln(\epsilon)})$ (se observa que el número de condición ha de ser ¿1 para que esta afirmación tenga sentido). La gran diferencia entre el descenso geométrico y el descenso por gradiente acelerado o de Nesterov radica, como se discutirá en las secciones siguientes, en su aplicabilidad.

2.4. Descenso geométrico

El descenso geométrico corresponde a uno de los dos métodos que se expondrán en este texto para optimizar el orden de convergencia de un método de tipo black box en el caso de funciones α-fuertemente convexas y β-suaves. Este método, que se utiliza para funciones definidas en todo el espacio $\mathbb{R}^n$, nos permitirá mejorar el orden de convergencia del descenso por gradiente en un factor $\sqrt{k}$, de $O\left(k \ln\left(\frac{1}{\epsilon}\right)\right)$ a $O\left(\sqrt{k} \ln\left(\frac{1}{\epsilon}\right)\right)$, donde $k = \frac{\alpha}{\beta}$ es de nuevo el número de condición para funciones β-suaves y α-fuertemente convexas. Lógicamente esto resultará interesante para valores $k > 1$ elevados.

2.4.1. Interpretación geométrica

El apodo "geométrico" viene motivado por la forma en que el algoritmo actúa. Partiendo de la base de que de alguna manera se ha conseguido encerrar el mínimo en una bola de radio R se van a obtener a partir de intersecciones con otras bolas, en las que resulta que también se encuentra localizado el mínimo, conjuntos cada vez más pequeños en los que tenemos garantías de que el mínimo no se queda fuera. El proceso es posible gracias a las propiedades de α-fuerte convexidad y β-suavidad de f y a una serie de resultados sobre intersección de bolas. En esta subsección $B(c, R)$ denotará la bola de centro c y radio R ($B(c, R) = \{x \in \mathbb{R}^n : ||x - c|| \leq R\}$)

Sea $x \in \mathbb{R}^n$ definimos

$$x^+ := x - \frac{1}{\beta}\nabla f(x) \quad \text{, y} \quad x^{++} := x - \frac{1}{\alpha}\nabla f(x)$$

Como f es α-fuertemente convexa se deduce fácilmente

$$-\frac{f(x) - f(y)}{\alpha} \geq \frac{\nabla f(x)^T}{\alpha}(y - x) + \frac{1}{2}||x - y||^2$$

Ahora de la igualdad

$$\frac{\nabla f(x)^T}{\alpha}(y - x) = \frac{||y - x + \frac{1}{\alpha}\nabla f(x)||^2}{2} - \frac{||\nabla f(x)||^2}{2\alpha^2} - \frac{||y - x||^2}{2}$$

se deduce que $\forall x, y \in \mathbb{R}^n$ se cumple

$$\frac{\alpha}{2}||y - x + \frac{1}{\alpha}\nabla f(x)||^2 \leq \frac{||\nabla f(x)||^2}{2\alpha} - (f(x) - f(y))$$

Si se elige $y = x^*$ se ha hallado, para cada punto x, una bola de centro x^{++} en la que se encuentra encerrado el mínimo.

$$x^* \in B\left(x^{++}, \sqrt{\frac{||\nabla f(x)||^2}{\alpha^2} - \frac{2}{\alpha}(f(x) - f(x^*))}\right)$$

Además por ser f β-suave, la desigualdad $f(x^+) - f(x) \leq -\frac{1}{2\beta}||\nabla f(x)||$ permitirá reducir el radio de la bola

$$x^* \in B\left(x^{++}, \sqrt{\frac{||\nabla f(x)||^2}{\alpha^2}\left(1 - \frac{1}{k}\right) - \frac{2}{\alpha}(f(x^+) - f(x^*))}\right) \tag{2.4.1}$$

Basta llevar a cabo las siguientes comprobaciones

$$\begin{aligned} ||x^* - x^{++}|| &\leq \frac{||\nabla f(x)||^2}{\alpha^2} - \frac{2}{\alpha}(f(x) - f(x^+) + f(x^+) - f(x^*)) \\ &= \frac{||\nabla f(x)||^2}{\alpha^2} - \frac{2}{\alpha}\left(\frac{||\nabla f(x)||^2}{2\beta} + f(x^+) - f(x^*)\right) \\ &= \frac{||\nabla f(x)||^2}{\alpha^2}\left(1 - \frac{1}{k}\right) - \frac{2}{\alpha}(f(x^+) - f(x^*)) \end{aligned}$$

El siguiente lema dará pie a la construcción del método geométrico.

Lema 2.32 Para todo $g \in \mathbb{R}^n$ y $\epsilon \in (0,1)$ existe $x \in \mathbb{R}^n$ tal que

$$B(0,1) \cap B\left(g, ||g||\sqrt{1-\epsilon}\right) \subset B\left(x, \sqrt{1-\epsilon}\right)$$

Dem: Supongamos $||g|| \geq 1$ sino el resultado sería trivial pues

$$B(0,1) \cap B\left(g, ||g||\sqrt{1-\epsilon}\right) \subset B\left(g, \sqrt{1-\epsilon}\right)$$

Obviamente si la intersección es vacía, el resultado también es trivial. Sea $y \in B(0,1) \cap B\left(g, ||g||\sqrt{1-\epsilon}\right)$. Se define $x = g - \left((1 - ||g||)\sqrt{1-\epsilon}\right)u$ siendo $u \in \mathbb{R}^n$ un vector unitario. Entonces

$$||y - x|| = ||y - g + g - x|| \leq ||y - g|| + ||g - x|| \leq ||g||\sqrt{1-\epsilon} + (1 - ||g||)\sqrt{1-\epsilon} = \sqrt{1-\epsilon}$$

□

Por lo tanto si llamamos $A = B(x_0, R)$ a la bola de partida que contiene al mínimo y $B = B\left(x_0^{++}, \frac{||\nabla f(x_0)||}{\alpha}\sqrt{1 - \frac{1}{k}}\right)$ la estrategia a seguir es clara.

Tras una traslación por $-x_1$ y una homotecia por $\frac{1}{R}$ las bolas

$$\bar{A} = \frac{A - x_0}{R} = B(0,1) \text{ y } \bar{B} = \frac{B - x_0}{R} = B\left(\frac{\nabla f(x_0)}{R\alpha}, \frac{||\nabla f(x_0)||}{R^2\alpha}\sqrt{1 - \frac{1}{k}}\right)$$

encajan en las hipótesis del lema previo. Este afirma la existencia de un punto $\bar{x_1}$ tal que $\bar{A} \cap \bar{B} \subset B\left(\bar{x_1}, \sqrt{1 - \frac{1}{k}}\right)$. Así pues, tras invertir la traslación y la homotecia se puede afirmar que

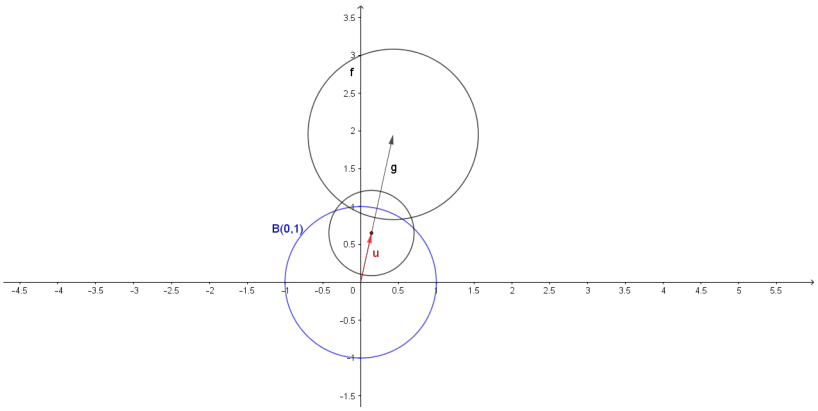


Figura 2.6: Lema 2.32

$x^* \in A \cap B \subset B\left(x_1, R\sqrt{1-\frac{1}{k}}\right)$, donde $x_1 = R(\bar{x_1}+x_0)$ y es posible proceder de la misma manera para cada iteración tomando como A la bola recién obtenida y como B la bola proporcionada por (2.4.1) para el iterante x_t. Así, reduciendo el radio de la bola $1-\frac{1}{k}$ en cada iteración, estamos ante una convergencia par a la del descenso por gradiente.

Nótese que se ha obviado una parte de (2.4.1) en cada iteración, $-\frac{2}{\alpha}(f(x^+) - f(y))$, y que de $f(x^+) - f(x) \leq -\frac{1}{2\beta}||\nabla f(x)||^2$ (2.1.10) se deduce

$$\begin{aligned} -\frac{2}{\alpha}(f(x^+) - f(x^*)) &\geq \frac{1}{\alpha\beta}||\nabla f(x)||^2 - \frac{2}{\alpha}(f(x^*) - f(x)) \\ &\geq \frac{1}{\alpha^2 k}||\nabla f(x)||^2 \end{aligned}$$

Así pues, resulta que el radio de cada nueva bola puede ser reducido aún más, obteniendo $B\left(x_t, \sqrt{R_t^2 - \frac{||\nabla f(x)||^2}{\alpha^2 k}}\right)$ para luego aplicar la intersección con (2.4.1). De esta forma se consigue acelerar la convergencia en $1-\frac{1}{\sqrt{k}}$ como se garantiza en el siguiente lema.

Lema 2.33 Para todo $g \in \mathbb{R}^n$ y $\epsilon \in (0,1)$, existe $x \in \mathbb{R}^n$ tal que

$$B\left(0, \sqrt{1-\epsilon||g||^2}\right) \cap B\left(g, ||g||\sqrt{(1-\epsilon)}\right) \subset B\left(x, \sqrt{1-\sqrt{\epsilon}}\right)$$

No obstante, al realizar la reducción $R_t^2 - \frac{||\nabla f(x)||^2}{\alpha^2 k}$ se ha de tener en cuenta que esta cantidad puede ser negativa. Para lidiar con esto sin dar lugar a problemas tenemos el siguiente resultado que refuerza aún más los límites dados en el lema anterior y que se usará en la demostración del teorema 2.35 una vez descrito el método de descenso geométrico con exactitud.

Lema 2.34 Para todo $a \in \mathbb{R}^n$, $\epsilon \in (0,1)$ y $g \in \mathbb{R}_+$. De forma que $||a|| \geq g^2$, existe $c \in \mathbb{R}^n$ tal que para todo $\delta > 0$

$$B\left(0, \sqrt{1-\epsilon g^2 - \delta}\right) \cap B\left(a, g\sqrt{(1-\epsilon)-\delta}\right) \subset B\left(c, \sqrt{1-\sqrt{\epsilon}-\delta}\right) \tag{2.4.2}$$

2.4.2. El método

Sean $x_0 \in \mathbb{R}^n$, $c_0 = x_0^{++}$ y $R_0^2 = \left(1 - \frac{1}{k}\right) \frac{||\nabla f(x_0)||^2}{\alpha^2}$. Para $t \geq 0$ se define

$$x_{t+1} = argmin_{\{x \in (1-\lambda)c_t + \lambda x_t^+ : \lambda \in \mathbb{R}\}} f(x) \tag{2.4.3}$$

y c_{t+1} y R_{t+1} son respectivamente el centro y el radio de la bola dada por el lema para la intersección

$$B\left(c_t, \sqrt{R_t^2 - \frac{||\nabla f(x_{t+1})||^2}{\alpha^2 k}}\right) \cap B\left(x_t^{++}, \frac{||\nabla f(x_{t+1})||}{\alpha}\sqrt{1 - \frac{1}{k}}\right)$$

Se tiene el siguiente resultado de convergencia para el método de descenso geométrico.

Teorema 2.35 El descenso geométrico para funciones α-fuertemente convexas y β-suaves satisface

$$x^* \in B\left(c_t, R_t\right) \forall t \geq 0 \text{ y } R_{t+1} \leq \sqrt{1 - \frac{1}{\sqrt{k}}} R_t$$

En particular

$$||x^* - c_t||^2 \leq \left(1 - \frac{1}{\sqrt{k}}\right)^t R_0^2$$

Dem: Se probará por inducción

$$x^* \in B\left(c_t, \sqrt{R_t^2 - \frac{2}{\alpha}(f(x_t^+) - f(x^*))}\right)$$

Se ve claramente que probar esto implicaría probar $x^* \in B(c_t, R_t)$. Además $R_{t+1} \leq \sqrt{1 - \frac{1}{\sqrt{k}}} R_t$ como se verá a posteriori.

Para t = 0 es cierto gracias a (2.4.1) y a la definición de c_0 y R_0. Se supone $x^* \in B(c_t, \sqrt{R_t^2 - \frac{2}{\alpha}(f(x_t^+) - f(x^*))}$ como hipótesis de inducción y se procede a verificar que se cumple también para $t+1$. Por la β-suavidad de f (2.1.10) y por la definición de x_{t+1} se obtiene

$$f(x_{t+1}^+) \leq f(x_{t+1}) - \frac{1}{2\beta}||\nabla f(x_{t+1})||^2 \leq f(x_t^+) - \frac{1}{2\beta}||\nabla f(x_{t+1})||^2$$

Entonces

$$\begin{aligned}
\frac{2}{\alpha}(f(x_t^+) - f(x^*)) &= \frac{2}{\alpha}(f(x_t^+) - f(x_{t+1}^+) + f(x_{t+1}^+) - f(x^*)) \\
&\geq \frac{2}{\alpha}\frac{1}{2\beta}||\nabla f(x_{t+1})||^2 - \frac{2}{\alpha}(f(x_{t+1}^+) - f(x^*)) \\
&= \frac{1}{\alpha^2 k}||\nabla f(x_{t+1})||^2 - \frac{2}{\alpha}(f(x_{t+1}^+) - f(x^*))
\end{aligned}$$

$$\Rightarrow x^* \in B\left(c_t, \sqrt{R_t^2 - \frac{||\nabla f(x_{t+1})||^2}{\alpha^2 k} - \frac{2}{\alpha}(f(x_{t+1}^+) - f(x^*))}\right) (:= B_1)$$

Además de nuevo por 2.4.1

$$x^* \in B\left(x_{t+1}^{++}, \sqrt{\frac{||\nabla f(x_{t+1})||^2}{\alpha^2}\left(1-\frac{1}{k}\right) - \frac{2}{\alpha}(f(x_{t+1}^+) - f(x^*))}\right) (:= B_2)$$

Denotando $g = \frac{||\nabla f(x_{t+1})||}{\alpha R_t}$, $\epsilon = \frac{1}{k}$, $\delta = \frac{2}{\alpha}\frac{f(x_{t+1})-f(x^*)}{R_t^2}$ y $a = \frac{x_{t+1}^{++}-c_t}{R_t}$ estamos en condiciones de aplicar el lema 2.34 para $\frac{B_1-c_t}{R_t} \cap \frac{B_2-c_t}{R_t}$, basta comprobar $||a|| \geq g$ o lo que es lo mismo

$$\begin{aligned} ||x_{t+1}^{++} - c_t||^2 &= ||x_{t+1} - \frac{1}{\alpha}\nabla f(x_{t+1}) - c_t||^2 \\ &= ||x_{t+1} - c_t||^2 + \frac{1}{\alpha^2}||\nabla f(x_{t+1})||^2 - \frac{1}{\alpha}\nabla f(x_{t+1})^T(x_{t+1} - c_t) \\ &\geq \frac{||\nabla f(x_{t+1})||^2}{\alpha^2} \end{aligned}$$

La última desigualdad se deduce del hecho de que $\nabla f(x_{t+1})^T(x_{t+1} - c_t) = 0$. El lema nos garantiza la existencia de un cierto $\overline{c_{t+1}}$ tal que $B_1 \cap B_2$ está contenido en

$$B\left(\overline{c_{t+1}}, \sqrt{1 - \sqrt{\frac{1}{k}} - \frac{2}{\alpha R_t^2}(f(x_{t+1}) - f(x^*))}\right)$$

Reescalando convenientemente y denontando $c_{t+1} = R_t(\overline{c_{t+1}} + c_t)$ y $R_{t+1} = R_t\sqrt{1-\frac{1}{k}}$ se obtiene

$$B_1 \cap B_2 \subset B\left(c_{t+1}, \sqrt{R_{t+1}^2 - \frac{2}{\alpha}(f(x_{t+1}) - f(x^*))}\right)$$

completándose así la inducción y quedando claro ahora que también se demuestra $R_{t+1} \leq \sqrt{1-\frac{1}{k}}R_t$. Nótese que c_{t+1} y R_{t+1} así definidos son válidos según la descripción del método ya que

$$B(c_{t+1}, R_{t+1}) \supset B\left(c_t, \sqrt{R_t^2 - \frac{||\nabla f(x)||^2}{\alpha^2 k}}\right) \cap B\left(x_t^{++}, \frac{||\nabla f(x)||}{\alpha}\sqrt{1-\frac{1}{k}}\right)$$

□

2.4.3. Convergencia e implementación

Al inicio de la sección 2.4.1 se afirmaba que el orden de convergencia del descenso geométrico para funciones β-suaves y α-fuertemente convexas alcanzaba el orden de $O\left(\sqrt{k}\ln\left(\frac{1}{\epsilon}\right)\right)$. El corolario que se prueba a continuación servirá como justificación de tal afirmación.

Corolario 2.36 Si se realizan al menos $\frac{1}{\ln\left(1-\frac{1}{k}\right)^{-1}}\ln\left(R_0^2\frac{1}{\epsilon}\right)$ iteraciones el método de descenso geómetrico alcanzará un aproximante a distancia menor que ϵ del verdadero valor del minimizador. El ϵ-aproximante, en virtud del teorema 2.35, será el centro c_t de la última bola generada por el método.

Dem: El teorema 2.35 garantiza que tras t iteraciones se obtiene un iterante c_t tal que $||x^* - c_t||^2 \leq \left(1 - \frac{1}{\sqrt{k}}\right)^t R_0^2$. Por lo tanto dado que $\left(1 - \frac{1}{\sqrt{k}}\right) = \frac{\sqrt{k}-1}{\sqrt{k}}$ y $0 < \frac{\sqrt{k}-1}{\sqrt{k}} < 1$

$$||x^* - c_t|| \leq \epsilon \Leftrightarrow t \geq \log_{(1-\frac{1}{\sqrt{k}})} \frac{\epsilon}{R_0^2} = \frac{\ln \frac{\epsilon}{R_0^2}}{\ln\left(1 - \frac{1}{\sqrt{k}}\right)} = \frac{\ln \frac{R_0^2}{\epsilon}}{\ln\left(\left(1 - \frac{1}{\sqrt{k}}\right)^{-1}\right)} = \frac{1}{\ln\left(1 - \frac{1}{k}\right)^{-1}} \ln\left(R_0^2 \frac{1}{\epsilon}\right)$$

Y la última expresión se es cierta por hipótesis.

□

Dado que $R_0^2 = \left(1 - \frac{1}{k}\right) \frac{||\nabla f(x_0)||^2}{\alpha^2} \leq \frac{||\nabla f(x_0)||^2}{\alpha^2}$ se tiene que $\ln\left(R_0^2 \frac{1}{\epsilon}\right) = c \ln \frac{1}{\epsilon}$, con c una constante. Además, debido a que $\ln(1+x)$ y x son infinitésimos equivalentes cuando $x \to \infty$, se deduce

$$\frac{1}{\ln\left(1 - \frac{1}{k}\right)^{-1}} \ln\left(R_0^2 \frac{1}{\epsilon}\right) = \frac{-1}{\ln\left(1 - \frac{1}{k}\right)} \ln\left(R_0^2 \frac{1}{\epsilon}\right) = O\left(\sqrt{k} \ln\left(\frac{1}{\epsilon}\right)\right) \text{ cuando } k \to \infty \,, \epsilon \to 0$$

Es decir, el corolario 2.36 prueba el orden de convergencia para el descenso geométrico.

El método de descenso geométrico desentraña, sin embargo, verdaderos problemas a la hora de llevar a cabo su implementación. Para empezar (2.4.3) puede ser irresoluble si la expresión de $\nabla f(x)$ se antoja complicada. Aún suponiendo que no sea así, el hecho de que se conozca la existencia del centro de la nueva bola donde se encuentra localizado el mínimo no facilita su cálculo en ningún sentido, es decir, no se tiene ninguna orientación de cómo calcularlo a pesar de la seguridad de su existencia. Así pues, aunque teóricamente se ha encontrado un mñetodo de optimización dentro del modelo black-box con una convergencia más rápida que el descenso por gradiente, este sigue prevalece gracias a su utilidad práctica.

2.5. Descenso por gradiente acelerado de Nesterov

El segundo método que mencionábamos anteriormente recibe el nombre de "descenso por gradiente acelerado de Nesterov". Alcanza el orden de convergencia óptimo para un método de tipo black-box con funciones α-fuertemente convexas y β-suaves, en este caso, definidas de nuevo en todo el espacio $\mathbb{R}^n$. Es decir, el número de iteraciones necesario para alcanzar un ϵ-aproximante es de nuevo $O\left(\sqrt{k} \ln\left(\frac{1}{\epsilon}\right)\right)$.

Dado $x_1 \in \mathbb{R}^n$ se definenen para $t \geq 0$

$$\begin{aligned} y_{t+1} &= x_t - \frac{1}{\beta} \nabla f(x_t) \\ x_{t+1} &= \left(1 + \frac{\sqrt{k}-1}{\sqrt{k}+1}\right) y_{t+1} - \frac{\sqrt{k}-1}{\sqrt{k}+1} y_t \end{aligned}$$

donde k, de nuevo, denotará el número de condición de la función objetivo.

Teorema 2.37 El descenso por gradiente acelerado de Nesterov satisface

$$f(y_t) - f(x^*) \leq \frac{\alpha+\beta}{2}||x_1 - x^*||^2 \exp(-\frac{t-1}{\sqrt{k}})$$

Dem: En primer lugar se definen las funciones ϕ_s, cuadráticas y α-fuertemente convexas, de manera recursiva de la siguiente manera.

$$\begin{aligned}\phi_1(x) &= f(x_1) + \frac{\alpha}{2}||x - x_1||^2, \\ \phi_{s+1}(x) &= \left(1 - \frac{1}{\sqrt{k}}\right)\phi_s(x) + \frac{1}{\sqrt{k}}\left(f(x_s) + \bigtriangledown f(x_s)^T(x - x_s) + \frac{\alpha}{2}||x - x_s||^2)\right)\end{aligned}$$

Se prueba por inducción

$$\phi_{s+1} \leq f(x) + \left(1 - \frac{1}{\sqrt{k}}\right)^s (\phi_1(x) - f(x)) \tag{2.5.1}$$

Para $s = 0$ se comprueba fácilmete que $\phi_1(x) \leq f(x) + \phi_1(x) - f(x) = \phi_1(x)$. Se supone que la desigualdad es válida para un cierto s y se prueba para $s + 1$.

$$\begin{aligned}\phi_{s+1} &= \left(1 - \frac{1}{\sqrt{k}}\right)\phi_s(x) + \frac{1}{\sqrt{k}}\left(f(x_s) + \bigtriangledown f(x_s)^T(x - x_s) + \frac{\alpha}{2}||x - x_s||^2\right) \\ &\leq \left(1 - \frac{1}{\sqrt{k}}\right)(\phi_s(x) - f(x)) + f(x) \\ &\leq \left(1 - \frac{1}{\sqrt{k}}\right)^s (\phi_1(x) - f(x)) + f(x)\end{aligned}$$

La primera desigualdad viene de que f es α-fuertemente convexa y por lo tanto $f(x_s) + \bigtriangledown f(x_s)^T(x - x_s) + \frac{\alpha}{2}||x - x_s||^2 \leq f(x)$ y la segunda es consecuencia de la hipótesis de inducción. También por inducción se prueba

$$f(y_s) \leq \min_{x \in \mathbb{R}^n} \phi_s(x) \tag{2.5.2}$$

La prueba es bastante tediosa con lo que no se desarrollará aquí. Veamos sin embargo como se deduce el resultado fácilmente a partir de (2.5.1) y (2.5.2). Por ser f β-suave se cumple $f(x) - f(x^*) \leq \frac{\beta}{2}||x - x^*||^2$. Aplicamos (2.5.2), (2.5.1), hacemos uso de la definición de ϕ_1 y usamos la desigualdad anterior; en este orden, para concluir la prueba.

$$\begin{aligned}f(y_t) - f(x^*) &\leq \phi_t(x^*) - f(x^*) \leq \left(1 - \frac{1}{\sqrt{k}}\right)^{t-1} (\phi_1(x^*) - f(x^*)) \\ &= (f(x_1) + \frac{\alpha}{2}||x^* - x_1||^2 - f(x^*))\left(1 - \frac{1}{\sqrt{k}}\right)^{t-1} \\ &\leq \frac{\alpha+\beta}{2}||x^* - x_1||^2 \left(1 - \frac{1}{\sqrt{k}}\right)^{t-1}\end{aligned}$$

□

La virtud del descenso por gradiente acelerado se halla, más que en esta excelente cota de convergencia, en su fácil implementación. Esta es la gran diferencia con el descenso geométrico. Ambos alcanzan un orden de convergencia óptimo, pero en este último no es factible el cálculo de los iterantes, y no por el hecho de que calcularlos sea computacionalmente costoso sino porque no se conoce una forma general para calcularlos (quizá sea factible para algún caso específico). Por su parte, Nesterov da con un método cuya implementación es prácticamente igual de sencilla a la del descenso por gradiente clásico. De hecho, el primer paso del método coincide con el descenso por gradiente, mientras que el segundo supone una combinación convexa de dos puntos, es decir $2n$ productos y n sumas siendo n la dimensión. Cierto es que si la dimensión es verdaderamente elevada el costo computacional también lo será. A cambio de esto se gana una excelente cota superior de convergencia.

Asímismo es posible adaptar el método para funciones β-suaves que carecen de convexidad fuerte (es decir, $\alpha = 0$) mejorando la convergencia ofrecida anteriormente por el descenso por gradiente y el descenso por gradiente proyectado. Para ello se escoge una secuencia variable para cada iteración como sigue:

$$\lambda_0 = 0, \;\; \lambda_t = \frac{1 + \sqrt{1 + 4\lambda_{t-1}^2}}{2}, \;\; \gamma_t = \frac{1 - \lambda_t}{\lambda_{t-1}}$$

Obsérvese que $\lambda_t \geq 0$ y $\gamma_t \leq 0$. El funcionamiento del método ahora es idéntico al anterior. Dados $x_1 = y_1$ arbitrarios se itera para $t \geq 1$

$$\begin{aligned} y_{t+1} &= x_t - \frac{1}{\beta}\nabla f(x_t) \\ x_{t+1} &= (1 - \gamma_t)y_{t+1} - \gamma_t y_t \end{aligned}$$

Como se probará a continuación, este método alcanza un orden de convergencia $\mathrm{O}(\frac{1}{\sqrt{\epsilon}})$ igualando el orden dado en el teorema 2.32 para la cota inferior.

Teorema 2.38 El descenso por gradiente acelerado de Nesterov para funciones β-suaves satisface

$$f(y_t) - f(x^*) \leq \frac{2\beta||x_1 - x^*||^2}{t^2}$$

Dem: Nótese que como la función objetivo está definida en todo $\mathbb{R}^n$ es cierto que $y_{s+1} = x_s^+ = \Pi_{\mathbb{R}^n}(x_s + \frac{1}{\beta}\nabla f(x_s)) = x_s + \frac{1}{\beta}\nabla f(x_s)$ por lo que se puede hacer uso del lema 2.15. Haciendo uso de también de la descripción del método se llega a la siguiente expresión.

$$\begin{aligned} f(y_{s+1}) - f(y_s) &\leq \nabla f(x_s)^T(x_s - y_s) - \frac{1}{2\beta}||\nabla f(x_s)||^2 \\ &= \beta(x_s - y_{s+1})^T(x_s - y_s) - \frac{\beta}{2}||x_s - y_{s+1}||^2 \qquad (1) \end{aligned}$$

De la misma manera se obtiene

$$f(y_{s+1}) - f(x^*) \leq \beta(x_s - y_{s+1})^T(x_s - x^*) - \frac{\beta}{2}||x_s - y_{s+1}||^2 \qquad (2)$$

Se denota $\delta_s = f(y_s) - f(x^*)$ la cantidad que se desea acotar. Dado que $f(y_{s+1}) - f(y_s) = \delta_{s+1} - \delta_s$, si se multiplica (1) por $(\lambda_s - 1)$ y se le suma (2) se llega a la desigualdad siguiente

$$\begin{aligned} (\lambda_s - 1)(1) + (2) &= \lambda_s\delta_{s+1} - (\lambda_s - 1)\delta_s \\ &\leq \beta(x_s - y_{s+1})^T(\lambda_s x_s - (\lambda_s - 1)y_s - x^*) - \frac{\beta}{2}\lambda_s||x_s - y_{s+1}||^2 \end{aligned}$$

Se multiplica esta expresión por λ_s. Como se verifica que para todo a, $b \in \mathbb{R}^n$, $2a^tb - ||a||^2 = ||b||^2 - ||b-a||^2$, eligiendo $a = \lambda_s(x_s - y_{s+1})$ y $b = \lambda_s x_s - (\lambda_s - 1)y_s - x^*$ se obtiene

$$\begin{aligned} \lambda_s^2\delta_{s+1} - \lambda_{s-1}^2\delta_s &\leq \frac{\beta}{2}2\lambda_s(x_s - y_{s+1})^T(\lambda_s x_s - (\lambda_s - 1)y_s - x^*) - \frac{\beta}{2}\lambda_s^2||x_s - y_{s+1}||^2 \\ &= \frac{\beta}{2}\left(||\lambda_s x_s - (\lambda_s - 1)y_s - x^*||^2 - ||\lambda_s y_{s+1} - (\lambda_s - 1)y_s - x^*||^2\right) \end{aligned}$$

Acudiendo a la definición del método se deduce

$$\begin{aligned} x_{s+1} = y_{s+1} + \gamma_s(y_s - y_{s+1}) &\Leftrightarrow \lambda_{s+1}x_{s+1} + (1 - \lambda_s)(y_s - y_{s+1}) \\ &\Leftrightarrow \lambda_{s+1}x_{s+1} - (\lambda_{s+1} - 1)y_{s+1} = \lambda_s y_{s+1} - (\lambda_s - 1)y_s \end{aligned}$$

Se denota $u_s := \lambda_s x_s - (\lambda_s - 1)y_s - x^*$. De las expresiones anteriores se obtiene $\lambda_s^2\delta_{s+1} - \lambda_{s-1}^2\delta_s \leq \frac{\beta}{2}(||u_s||^2 - ||u_{s+1}||^2)$. Ahora tomando la suma desde 1 hasta $t-1$

$$\lambda_{t-1}^2\delta_t \leq \frac{\beta}{2}(||u_1||^2 - ||u_t||^2) \Rightarrow \delta_t \leq \frac{\beta}{2\lambda_{t-1}^2}||u_1||^2$$

$\lambda_{t-1} \geq \frac{t}{2}$ (se ve mediante una sencilla inducción), lo cual concluye la prueba.

□

Haciendo memoria, el teorema 2.14 de convergencia del descenso por gradiente para funciones β-suaves garantizaba que el iterante x_t del método cumplía

$$f(x_t) - f(x^*) \leq \frac{2\beta||x_1 - x^*||^2}{t-1}$$

Esto se traduce en un orden de convergencia de $O(\frac{1}{\epsilon})$ frente al orden $O(\frac{1}{\sqrt{\epsilon}})$ que ofrece el descenso por gradiente acelerado de Nesterov. Si se echa un ojo a la sección 2.3 sobre cotas inferiores, en concreto al teorema 2.30, se observa que además este es el orden de convergencia óptimo que puede alcanzar un algoritmo de tipo black box.

Capítulo 3

Métodos alternativos

En la práctica pocas veces se disfrutará de unas condiciones de regularidad tan favorables como las que se han tomado por hipótesis en el capítulo anterior para llegar a cotas de convergencia de orden exponencial. El mero hecho de que la función objetivo carezca de suavidad, por ejemplo, marca la diferencia entre disponer de un orden de convergencia $O(\frac{1}{\sqrt{\epsilon}})$ a solo poder garantizar $O(\frac{1}{\epsilon^2})$. No obstante, hasta ahora se han enmarcado los métodos de optimización dentro de un modelo black-box que solo permitía conocer salidas (valores de f y de $\nabla f(x)$) a partir de determinadas entradas (puntos del dominio de f), permaneciendo oculta cualquier otra información acerca de la estructura de la función. Esto supone una clara limitación. En la práctica es usual conocer la función que se desea optimizar y por lo tanto se hace posible explotar su estructura.

Con el nombre de métodos alternativos se hará referencia a algoritmos de optimización para funciones que carecen de la propiedad de β-suavidad, pero que al salirse del modelo black-box permiten analizar la estructura de la función objetivo para conseguir una optimización rápida hacia el mínimo. En este capítulo se van a exponer en concreto dos métodos útiles para llevar a cabo la siguiente optimización

$$\min_{x\in\mathbb{R}^n} f(x) + g(x)$$

donde se suponen f y g funciones definidas en $\mathbb{R}^n$ y convexas. Además f es β-suave y se conoce la expresión de g. Se supone que esta última es de alguna manera "simple". Se verá lo que esto quiere decir. Es factible llevar a cabo el método de descenso por gradiente para obtener un aproximante a mínimo pero, como ya se ha comentado, la falta de suavidad en la función suma no ofrece garantías de que el algoritmo converja al mínimo a la velocidad deseada. Así pues se van a dar dos métodos alternativos, ISTA y FISTA. Este último va a recuperar un orden de convergencia $O(\frac{1}{\sqrt{\epsilon}})$ (el mejor que se podía garantizar en un algoritmo black-box), haciendo tan útil el conocimiento de la estructura de g como hubiera sido el hecho de contar con una suavidad global en la función $f + g$.

3.1. ISTA (Iterative Shrinkage-Thresholding Algorithm)

En esta sección se introducirá primero el método ISTA haciendo hincapié en su relación con el descenso por gradiente y la aplicación proximal. Seguidamente se establecerán una serie de

consideraciones y resultados que tienen como objetivo llegar al teorema 3.6 de convergencia del método. En lo que resta de sección f representará siempre una función convexa β-suave y β, su constante de suavidad.

3.1.1. Aplicación proximal y envolvente de Moreau

La aplicación proximal, que se definirá en esta sección, da pie a una serie de algoritmos conocidos como algoritmos proximales que sirven como herramienta para resolver problemas de optimización convexa. En el caso que se pretende abordar aparece al tratar de obtener ISTA a partir del descenso por gradiente. A fin de cuentas veremos que ISTA no es más que una particularización de un algoritmo proximal más general. La aplicación proximal está estrechamente relacionada con otra aplicación conocida como envolvente de Moreau.

Sea $g : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ una función cerrada y convexa, lo que en particular implica que su epígrafe

$$epi(f) = \{(x, t) \in \mathbb{R}^n \times \mathbb{R} : g(x) \leq t\}$$

es un conjunto no vacío, cerrado y convexo. Se denotará por $dom(g)$ el conjunto de puntos propios de g

$$dom(g) = \{x \in \mathbb{R}^n : g(x) < +\infty\}$$

es decir, el conjunto de puntos en los que la función g toma valores finitos. En lo que resta de sección g representará una función descrita como hasta ahora sino se añade nada más.

Definición 3.1 Se denota por $g_\eta(v)$ y se denomina envolvente de Moreau de g con parámetro η a la función

$$g_\eta(v) = \min_{x \in \mathbb{R}^n} \{g(x) + \frac{1}{2\eta}||x - v||^2\}$$

La función a minimizar en la definición de la envolvente de Moreau es $h_\eta(v, x) = g(x) + \frac{1}{2\eta}||x - v||^2$, estrictamente convexa en x debido a que es suma de una función convexa y de una función estrictamente convexa. Esto sigfinica que, para un valor fijo de v, el mínimo de h_η, es decir el valor de $g_\eta(v)$, se alcanza en un único punto. Esta característica justifica la definición de aplicación proximal.

Definición 3.2 Se define la aplicación proximal de g con parámetro η como

$$P_\eta g(v) = \arg\min_{x \in \mathbb{R}^n} \left\{ g(x) + \frac{1}{2\eta}||x - v||^2 \right\}$$

Evaluar la aplicación proximal en $v \in \mathbb{R}^n$ equivale por lo tanto a calcular el minimizador en x de la función $h_\eta(v, x)$, cuyo valor, el mínimo, es precisamente el que toma la envolvente de Moreau en el mismo punto v.

En la Figura 3.1 se ha dado una interpretación del funcionamiento del proximal de una función g. Las curvas de nivel quedan representadas por las lineas más finas, mientras que la frontera de $dom(g)$ queda limitada por la línea gruesa.

Así, cuando se evalúa $P_\eta g$ en un punto $v \notin dom(g)$, este es enviado a la frontera y hacia el mínimo. Cuando $v \in dom(g)$ el proximal lo envía hacia el mínimo siempre sin salirse de la región

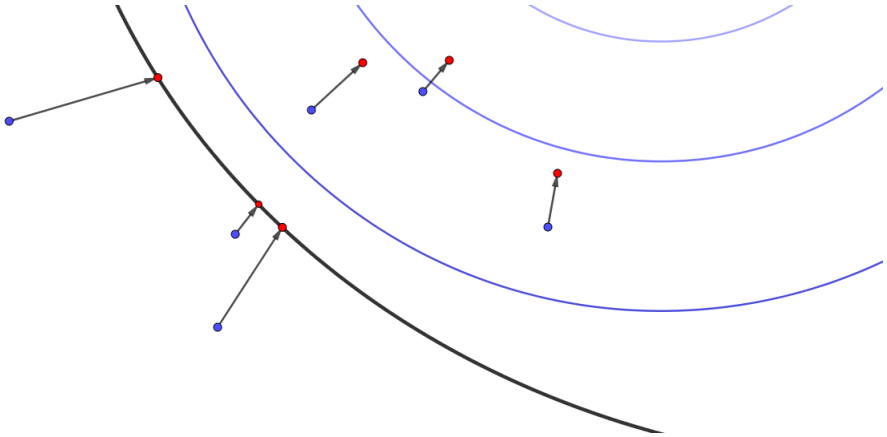


Figura 3.1: Interpretación de la aplicación proximal

de puntos propios. Se podría interpretar pues, el proximal de una funcion g evaluado en un punto v, como un punto que trata de mimizar g sin distanciarse en exceso de v. La graduación que se le da a la minimización de g o al distanciamiento respecto de v vendrá dada por el valor del parámetro η.

Cuando g es el indicador de un conjunto no vacío, cerrado y convexo $C \subset \mathbb{R}^n$

$$I_C(x) = \begin{cases} 0 & si \quad x \in C \\ +\infty & si \quad x \notin C \end{cases}$$

El proximal se reduce a la proyección ortogonal sobre C

$$\Pi_C(x) = \arg\min_{x \in C} ||x - v||$$

Luego el proximal se puede ver también como una generalización de la proyección ortogonal. Existe una abundante literatura sobre las propiedades de la aplicación proximal y la envolvente de Moreau así como de los algoritmos proximales a los que dan pie y su uso en la optimización convexa, pero esto va más allá de los objetivos del trabajo. [7] es un texto muy completo en este sentido para quien tenga mayor interés. Se mencionaran tan solo dos propiedades de utilidad para los propósitos de este texto.

Va a ser frecuente que la función g sea completemente separable.

$$g(x) = \sum_{i=1}^{n} g_i(x_i)$$

Esto tiene importante valor a la hora de optimizar, pues en lugar de enfrentarse a un problema en dimensión n arbitrariamente grande, se tendrán n problemas (puede que además idénticos) en dimensión 1 gracias a la siguiente propiedad que se da cuando g es separable.

$$(P_\eta g(v))_i = P_\eta g_i(v_i) \tag{3.1.1}$$

Lo cual viene a decir que cada componente de el proximal de una función g completamente separable en un punto $v \in \mathbb{R}^n$ equivale al proximal de la respectiva componente de g en la respectiva componente de v. La justificación es sencilla puesto que basta tener en cuenta que el

término $\frac{1}{2\eta}||x-v||^2$ es tambien completamente separable por lo que a la hora de minimizar h_η en x se procede componente a componente. Si además las funciones g_i son idénticas para todo i se tiene un único problema en dimensión 1.

La otra propiedad a la que se desea hacer referencia tiene que ver con la envolvente de Moreau y se enunciará como una proposición.

Proposición 3.3 La envolvente de Moreau de una función g es una función convexa.

Dem: Sean $v_1, v_2 \in \mathbb{R}^n$, sea $\alpha \in (0,1)$ y sean $x_1 = P_\eta g(v_1)$ y $x_2 = P_\eta g(v_2)$. Entonces

$$\begin{aligned} g_\eta(v_1) &= g(x_1) + \frac{1}{2\eta}||x_1 - v_1||^2 \\ g_\eta(v_2) &= g(x_2) + \frac{1}{2\eta}||x_2 - v_2||^2 \end{aligned}$$

Denotando $\hat{x} = \alpha x_1 + (1-\alpha)x_2$ y haciendo uso de esta observación y de la convexidad de g se prueba la convexidad de g_η:

$$\begin{aligned} \alpha g_\eta(v_1) + (1-\alpha)g_\eta(v_2) &\geq g(\alpha x_1 + (1-\alpha)x_2) + \frac{1}{2\eta}\left(\alpha||x_1 - v_1||^2 + (1-\alpha)||x_2 - v_2||^2\right) \\ &\geq g(\alpha x_1 + (1-\alpha)x_2) + \frac{1}{2\eta}||\alpha(x_1 - v_1) + (1-\alpha)(x_2 - v_2)||^2 \\ &= g(\hat{x}) + \frac{1}{2\eta}||\hat{x} - (\alpha v_1 + (1-\alpha)v_2)||^2 \\ &\geq g_\eta(\alpha v_1 + (1-\alpha)v_2) \end{aligned}$$

□

Por último se expondrá un sencillo lema que será de ayuda para probar el lema 3.5 que precede al teorema de convergencia 3.6.

Lema 3.4 $z = P_\eta g(v)$, si y solo si, $\frac{v-z}{\eta}$ es un subdiferncial de la función g.

Dem: Es inmediato probar que una función convexa g alcanza su mínimo en x^*, si y solo si, $0 \in \partial g(x^*)$. Esto llevado al caso de la aplicación proximal equivale a $z = P_\eta g(v) \Leftrightarrow 0 \in \partial g(z) + \frac{1}{\eta}(z - v)$, donde esta última es una suma de conjuntos. A raíz de esta última equivalencia se deduce directamente el resultado.

□

3.1.2. Descripción del método

Tomando como punto de partida un iterante inicial $x_1 \in \mathbb{R}^n$, el método ISTA propone la siguiente iteración para $t \leq 0$

$$x_{t+1} = \underset{x \in \mathbb{R}^n}{\arg\min} \left(g(x) + \frac{1}{2\eta}||x - (x_t - \eta \nabla f(x_t))||^2 \right) \tag{3.1.2}$$

o equivalentemente

$$x_{t+1} = P_\eta g(x_t - \eta \nabla f(x_t))$$

Si se denota $\hat{x}_t = x_t - \eta \nabla f(x_t)$ resulta que $\hat{x}_t$ es exactamente el iterante de descenso por gradiente y lo que se tiene en (3.1.2) es $P_\eta g(\hat{x}_t)$. Dado el análisis previo de la aplicación proximal está claro que lo que ISTA pretende es obtener un minimizador de la función g sin alejarse demasiado del iterante de descenso por gradiente en cada paso. El iterante de descenso por gradiente bajo una elección conveniente de η resulta ser además un buen aproximante al mínimizador de f, por lo que ISTA intenta a fin de cuentas minimizar $f(x) + g(x)$. Se insiste una vez más que el éxito de ISTA recaerá en la elección de la longitud de paso, es decir, del parámetro η.

En [7] se desarrollan una serie de algoritmos de optimización convexa basados en la aplicación proximal, entre ellos una generalización de ISTA que recibe el nombre de descenso por gradiente proximal. Para el problema $\min_{x \in \mathbb{R}^n} f(x) + g(x)$, con las condiciones expuestas al inicio de la sección, el descenso por gradiente proximal realiza la iteración

$$x_{t+1} = P_{\eta^t} g(x_t - \eta^t \nabla f(x_t))$$

Si se elige una longitud de paso constantemente igual a η se obtiene precisamente la iteración de ISTA.

3.1.3. Convergencia

Si en el lema 3.4 se toman $v = \hat{y} = y - \frac{1}{\beta} \nabla f(y)$ y $\eta = \frac{1}{\beta}$ se obtiene

$$z = P_\eta g(\hat{y}) \Leftrightarrow \beta(\hat{y} - z) \in \partial g(z)$$

Por otro lado el lema 2.13 establecía que una función diferenciable y β-suave verifica (2.1.8) $\forall x, y \in \mathbb{R}^n$

$$0 \leq f(x) - f(y) - \nabla f(y)^T (x - y) \leq \frac{\beta}{2} ||x - y||^2$$

Estos dos resultados van a servir para probar el lema 3.5, que caracteriza los óptimos de $P_{\frac{1}{\beta}} g$, donde se ha tomado $\eta = \frac{1}{\beta}$, y resulta clave para probar al fin la convergencia del método ISTA.

En el lema 3.5 se usará la función

$$Q(x, y) := f(y) + \nabla f(y)^T (x - y) + \frac{\beta}{2} ||x - y||^2 + g(x)$$

la cual, fijado $y \in \mathbb{R}^n$ alcanza su mínimo en x en el mismo punto que la función a minimizar en la envolvente de Moreau evaluada en $\hat{y}$ y con parámetro $\frac{1}{\beta}$, $h_{\frac{1}{\beta}}(\hat{y}, x)$ (basta suprimir los términos constantes en x) y por lo tanto alcanza los minimizadores que se tratan de aproximar por ISTA.

Lema 3.5 Sea $F(x) = f(x) + g(x)$ y sean $y \in \mathbb{R}^n$ y $\beta > 0$ tales que

$$F(P_{\frac{1}{\beta}} g(\hat{y})) \leq Q(P_{\frac{1}{\beta}} g(\hat{y}), y). \tag{3.1.3}$$

Entonces, para todo $x \in \mathbb{R}^n$,

$$F(x) - F(P_{\frac{1}{\beta}} g(\hat{y})) \geq \frac{\beta}{2}||P_{\frac{1}{\beta}} g(\hat{y}) - y||^2 + \beta(y - x)^T (P_{\frac{1}{\beta}} g(\hat{y}) - y)$$

Dem: De (3.1.3) se deduce fácilmente

$$F(x) - F(P_{\frac{1}{\beta}} g(\hat{y})) \geq F(x) - Q(P_{\frac{1}{\beta}} g(\hat{y}), y) \tag{3.1.4}$$

De la convexidad de f y g, de la definición de subgradiente y del lema 3.4 se deduce

$$\begin{aligned} f(x) &\geq f(y) + \nabla f(y)^T (x - y) \\ g(x) &\geq g(P_{\frac{1}{\beta}} g(\hat{y})) + \xi_y^T (x - P_{\frac{1}{\beta}} g(\hat{y})) \end{aligned}$$

con $\xi_y = \beta(\hat{y} - P_\eta g(\hat{y}))$.

Como $F(x) = f(x) + g(x)$ se cumple

$$F(x) \geq f(y) + \nabla f(y)^T (x - y) + g(P_{\frac{1}{\beta}} g(\hat{y})) + \xi_y^T (x - P_{\frac{1}{\beta}} g(\hat{y})) \tag{3.1.5}$$

y de la definición de $Q(x, y)$ se obtiene

$$Q(P_{\frac{1}{\beta}} g(\hat{y}), y) = f(y) + \nabla f(y)^T (P_{\frac{1}{\beta}} g(\hat{y}) - y) + \frac{\beta}{2}||P_{\frac{1}{\beta}} g(\hat{y}) - y||^2 + g(P_{\frac{1}{\beta}} g(\hat{y})) \tag{3.1.6}$$

Usando (3.1.5) y (3.1.6) en (3.1.4) se llega finalmente al resultado

$$\begin{aligned} F(x) - F(P_{\frac{1}{\beta}} g(\hat{y})) &\geq -\frac{\beta}{2}||P_{\frac{1}{\beta}} g(\hat{y}) - y||^2 + (\nabla f(y) + \xi_y)^T (x - P_{\frac{1}{\beta}} g(\hat{y})(y)) \\ &= -\frac{\beta}{2}||P_{\frac{1}{\beta} g} (\hat{y}) - y||^2 + \beta(y - P_{\frac{1}{\beta}} g(\hat{y}))^T (x - P_{\frac{1}{\beta}} g(\hat{y})(y)) \\ &= \frac{\beta}{2}||P_{\frac{1}{\beta}} g(\hat{y}) - y||^2 + \beta(P_{\frac{1}{\beta}} g(\hat{y}) - y)^T (y - x) \end{aligned}$$

□

Si en (3.1.2) se elige $\eta = \frac{1}{\beta}$ como longitud de paso constante resulta que a partir de un iterante inicial $x_0 \in \mathbb{R}^n$ ISTA lleva a cabo la iteración para $t \geq 1$

$$x_{t+1} = P_{\frac{1}{\beta}} g(\hat{x}_t)$$

Gracias al lema 3.5 es posible probar que, al fin, el método ISTA alcanza un orden de convergencia $O(\frac{1}{\epsilon})$ cuando se tiene $\frac{1}{\beta}$ como longitud de paso.

Teorema 3.6 El método ISTA con una longitud de paso $\eta = \frac{1}{\beta}$ satisface

$$f(x_t) + g(x_t) - (f(x^*) + g(x^*)) \leq \frac{\beta||x_1 - x^*||^2}{2t}$$

donde se denota $x^* = \arg\min_{x \in \mathbb{R}^n} f(x) + g(x)$.

Dem: De (3.1.3) se deduce trivialmente $F(x_{k+1}) \leq Q(x_{k+1}, x_k)$ con lo que se dan las condiciones para aplicar el lema 3.5 con $x = x^*, y = x_k$ obteniendo

$$\frac{2}{\beta}(F(x^*) - F(x_{k+1})) \geq ||x_{k+1} - x_k||^2 + 2(x_k - x^*)^T(x_{k+1} - x_k) = ||x^* - x_{k+1}||^2 - ||x^* - x_k||^2$$

y sumando cada lado de la desigusldad desde $k = 0$ hasta $k = t - 1$

$$\frac{2}{\beta}\left(tF(x^*) - \sum_{k=0}^{t-1} F(x_{k+1})\right) \geq ||x^* - x_t||^2 - ||x^* - x_0||^2 \tag{3.1.7}$$

En virtud de nuevo del lema 3.5 con $x = y = x_k$

$$\frac{2}{\beta}(F(x_k) - F(x_{k+1})) \geq ||x_{k+1} - x_k||^2$$

Multiplicando por k y sumando desde $k = 0$ hasta $k = t - 1$, la expresión anterior es equivalente a

$$\frac{2}{\beta}\sum_{k=0}^{t-1}(kF(x_k) - (k+1)F(x_{k+1}) + F(x_{k+1}))) \geq \sum_{k=0}^{t-1} k||x_{k+1} - x_k||^2$$

y teniendo en cuenta que los dos primeros sumandos del lado izquierdo de la desigualdad forman una serie telescópica

$$\frac{2}{\beta}\left(-tF(x_t) + \sum_{k=0}^{t-1} F(x_{k+1})\right) \geq \sum_{k=0}^{t-1} k||x_k - x_{k+1}||^2 \tag{3.1.8}$$

Ahora sumando (3.1.7) y (3.1.8)

$$\frac{2t}{\beta}\left(F(x^*) - F(x_t)\right) \geq ||x^* - x_t||^2 + \sum_{k=0}^{t-1} k||x_k - x_{k+1}||^2 - ||x^* - x_0||^2$$

De donde finalmente se deduce la cota de convergencia

$$F(x_t) - F(x^*) \leq \frac{\beta||x^* - x_0||}{2t}$$

□

Sin embargo, este resulado de convergencia es útil en la medida en que (3.1.2) sea calculable. Se trata de una minimización que por lo general no resulta sencilla. Es ahora cuando entra en juego la simplicidad de g. Supóngase, por ejemplo, que g es separable, es decir, $g(x) = \sum_{i=1}^n g_i(x(i))$. En tal caso

$$\begin{aligned}(3.1.2) \quad &\equiv \quad x_{t+1} = \arg\min_{x\in\mathbb{R}^n}\left(\sum_{i=1}^n g_i(x(i)) + \frac{1}{2\eta}||x - (x_t - \eta\nabla f(x_t))||^2\right) \\ &\equiv \quad x_{t+1}(i) = \arg\min_{x(i)\in\mathbb{R}}\left(g_i(x(i)) + \frac{1}{2\eta}\left(x(i) - (x_t(i) - \eta\frac{\partial f}{\partial x_i}(x_t))\right)^2\right), \quad i \in \{1\ldots n\}\end{aligned}$$

y el problema de optimización en $\mathbb{R}^n$ se reduce a n problemas de optimización convexa en dimensión 1. Esto es bastante más asumible aunque se mantienen algunas limitaciones. Si las n componentes de g son distintas se tienen n problemas de optimización particulares. Suponiendo además que las componentes de $\triangledown f$ no son calculables se hace necesario recurrir a algún método iterativo n veces; lo cual, si n es elevado ralentiza considerablemente el método.

En cualquier caso se trata de una ventaja que se da a raíz de las propiedades del proximal que se adelantaban previamente, concretamente a la mencionada en (3.1.1)

$$(3.1.2) \equiv P_\eta g(\hat{x}_t) = (P_\eta g_i(\hat{x}_{t_i}))_{i=1}^n$$

Es frecuente que se den, no obstante, unas condiciones mucho más favorables. Imagínese el caso de $g(x) = \lambda||x||_1$. Como las componentes de g son idénticas tan solo hace falta resolver un problema de optimización convexa en dimensión 1.

$$(3.1.2) \equiv \arg\min_{x\in\mathbb{R}} \lambda|x| + \frac{1}{2\eta}(x - x_0)^2, \text{ donde } x_0 \in \mathbb{R} \text{ y } \lambda > 0$$

Además es posible obtener una solución analítica. Sea $f(x) := \lambda|x| + \frac{1}{2\eta}(x - x_0)^2$. f es diferenciable en $\mathbb{R} \setminus \{0\}$.

$$f'(x) = \begin{cases} f_1'(x) = -\lambda + \frac{1}{\eta}(x - x_0) & si \quad x < 0 \\ \\ f_2'(x) = \lambda + \frac{1}{\eta}(x - x_0) & si \quad x > 0 \end{cases}$$

Se distinguen tres casos que se pueden agrupar utilizando el operador de reducción

$$S_\alpha(x) = (|x| - \alpha)_+ sign(x)$$

$$\begin{cases} x_0 \in [-\infty, -\eta\lambda] & \text{el mínimo se alcanza en } x_0 + \eta\lambda \\ x_0 \in [-\eta\lambda, \eta\lambda] & f_1 \text{ decrece, } f_2 \text{ crece y el mínimo se alcanza en } 0 \\ x_0 \in [\eta\lambda, \infty] & \text{el mínimo se alcanza en } x_0 - \eta\lambda \end{cases}$$

Es claro que, denotando por x^* al mínimo de f, se cumple que $x^* = S_{\eta\lambda}(x_0)$. La optimización de (3.1.2) no conlleva entonces absolutamente ningún coste computacional adicional.

Ejemplo 3.7 El problema LASSO en su forma lagrangiana es un claro ejemplo en el que ISTA y también FISTA, como se verá a posteriori, resultan ser métodos adecuados para la obtención del mínimo.

$$\min_{\beta\in\mathbb{R}^p} \left\{||y - X\beta||_2^2 + \lambda||\beta||_1\right\}$$

La clave está en que el problema puede abordarse en la forma $\min_{\beta\in\mathbb{R}^p} f(\beta) + g(\beta)$ donde:

1. $g(\beta) = \lambda||\beta||_1$ es convexa pero no β-suave.
2. $f(\beta) = ||y - X\beta||_2^2$ es convexa y suave con constante de suavidad $2||X^TX||_2$

La constante de suavidad se deduce de que

$$\frac{\partial f}{\partial \beta_k}(\beta) = -2\sum_{i=1}^{N}\left[X(i,k)\left(y_i - \sum_{j=1}^{p} X(i,j)\beta_j\right)\right] = -2X(:,k)^T(y - X\beta)$$

dentando por $X(:,k)$ la columna k-ésima de la matriz, entonces

$$\begin{aligned} \nabla f(\beta) &= -2X^T(y - X\beta) \\ ||\nabla f(\beta_1) - \nabla f(\beta_2)||_2 &\leq 2||X^TX||_2||\beta_1 - \beta_2||_2 \end{aligned}$$

ISTA, por lo tanto, consiste en la siguiente iteración para cada $t \geq 1$

$$\beta_{t+1}(k) = S_{\eta\lambda}(\beta_t(k) - \eta\frac{\partial f}{\partial \beta_k}(\beta_t)) \text{ para cada } k \in 1 \ldots p$$

que es equivalente a

$$\beta_{t+1}(k) = S_{\eta\lambda}\left(\beta_t(k) + 2\eta X(:,k)^T(y - X\beta_t)\right)$$

Esta expresión supone para cada componente del nuevo iterante Np productos y $N(p-1)$ sumas para calcular $X\beta_t$; p productos y $p-1$ sumas que provienen de $2\eta X(:,k)^TX\beta_t$; y una suma adicional al añadir el término $\beta_t(k)$. En t iteraciones el costo computacional será de $tp^2(N+1)$ productos y $tp\left((p-1)(N+1)+1\right)$ sumas, además del costo de X^Ty que se calcula una sola vez. Lo que quiere decir que el método ISTA para el problema LASSO alcanza un ϵ-aproximante a mínimo de la función llevando a cabo un número de operaciones no mayor del orden de $O(\frac{Np^2}{\epsilon})$ siempre y cuando se elija $\eta = \frac{1}{2||X^TX||}$ como establece el teorema 3.6.

3.2. FISTA (Fast ISTA)

La idea de reutilizar ISTA combinándolo con el método que hasta ahora ha proporcionado la mejor cota de convergencia, el método acelerado de Nesterov, va a dar lugar a un nuevo método conocido como FISTA y a una cota de convergencia del orden $O(\frac{1}{\sqrt{\epsilon}})$ para la optimización que se analiza en este sección.

Así pues FISTA actúa de la siguiente manera. Dados

$$\lambda_0 = 0\ ,\ \lambda_t = \frac{1 + \sqrt{1 + 4\lambda_{t-1}^2}}{2} \text{ y } \gamma_t = \frac{1 - \lambda_t}{\lambda_{t+1}}$$

Sea $x_1 = y_1$ un punto inicial arbitrario, para $t \geq 1$ se definen

$$y_{t+1} = \arg\min_{x \in \mathbb{R}^n} g(x) + \frac{1}{2\eta}||x - (x_t - \eta\nabla f(x_t))||_2^2 \quad (3.2.1)$$

$$x_{t+1} = (1 - \gamma_t)y_{t+1} + \gamma_t y_t \quad (3.2.2)$$

De nuevo la idea intuitiva es la de minimizar $f+g$ usando que f es β-suave para aplicar el método de Nesterov a este sumando. Dado que el primer paso de la iteración que usa Nesterov coincide con el descenso por gradiente el primer paso de la iteración de FISTA coincide con la de ISTA, por las razones que se han expuesto en la sección anterior, mientras que el segundo paso de la iteración de FISTA es exactamente la misma que la del propio método de Nesterov.

Usando, como antes, el proximal $P_\eta g$ se puede reescribir la iteración de FISTA como

$$y_{t+1} = P_\eta g(\hat{x}_t) \tag{3.2.3}$$
$$x_{t+1} = (1-\gamma_t)y_{t+1} + \gamma_t y_t \tag{3.2.4}$$

De nuevo va a suceder que el orden de convergencia de FISTA va a igualar a aquel obtenido por Nesterov, es decir, la convergencia de FISTA va a ser del orden $O(\frac{1}{\sqrt{\epsilon}})$, el mejor obtenido hasta el momento. Para probar esto primero se darán una serie de lemas previos.

Lema 3.8 La secuencia $\{x_k, y_k\}$ generada por FISTA con $\eta = \frac{1}{\beta}$ cumple

$$\frac{2}{\beta}\left(\lambda_k^2 v_k - \lambda_{k+1}v_{k+1}\right) \geq ||u_{k+1}||^2 - ||u_k||^2 \tag{3.2.5}$$

donde $v_k = F(y_{k+1}) - F(x^*)$ y $u_k = \lambda_k y_{k+1} - (\lambda_k - 1)y_k - x^*$

Dem: Se aplica el lema 3.5 a los puntos $x = y_{k+1}$, $y = x_{k+1}$, y asímismo, a los puntos $x = x^*$, $y = x_{k+1}$ (ambos cumplen (3.1.3) dado que $F(y_{k+1}) \leq Q(y_{k+1}, x_{k+1})$ por (2.1.8)) para obtener

$$\begin{aligned}\frac{2}{\beta}(v_k - v_{k+1}) &\geq ||y_{k+2} - x_{k+1}||^2 + 2(y_{k+2} - x_{k+1})^T(x_{k+1} - y_{k+1}) \\ -\frac{2}{\beta}v_{k+1} &\geq ||y_{k+2} - x_{k+1}||^2 + 2(y_{k+2} - x_{k+1})^T(x_{k+1} - x^*)\end{aligned}$$

donde se ha tenido en cuenta que $y_{k+1} = P_{\frac{1}{\beta}}g(\hat{x}_k)$. Para relacionar v_k y v_{k+1} se multiplica la primera desigualdad por $\lambda_{k+1} - 1$ y se añade el resultado a la segunda.

$$\frac{2}{\beta}\left((\lambda_{k+1}-1)v_k - \lambda_{k+1}v_{k+1}\right) \geq \lambda_{k+1}||y_{k+2} - x_{k+1}||^2 + 2(y_{k+2} - x_{k+1})^T(\lambda_{k+1}x_{k+1} - (\lambda_{k+1}-1)y_{k+1} - x^*)$$

Como por la descripción del método se tiene que $\lambda_k^2 = \lambda_{k+1}^2 - \lambda_{k+1}$, si se multiplica la última desigualdad por λ_{k+1} se obtiene

$$\frac{2}{\beta}(\lambda_k^2 v_k - \lambda_{k+1}^2 v_{k+1}) \geq ||\lambda_{k+1}(y_{k+2} - x_{k+1})||^2 + 2\lambda_{k+1}(y_{k+2} - x_{k+1})^T(\lambda_{k+1}x_{k+1} - (\lambda_{k+1}-1)y_{k+1} - x^*)$$

El teorema de Pitágoras afirma que dados tres vectores $a, b, c \in \mathbb{R}^n$ se cumple

$$||b-a||^2 + 2(b-a)^T(a-c) = ||b-c||^2 - ||a-c||^2$$

tomando $a = \lambda_{k+1}x_{k+1}$, $b = \lambda_{k+1}y_{k+2}$ y $c = (\lambda_{k+1}-1)y_{k+1} + x^*$ es cierto que

$$\frac{2}{\beta}(\lambda_k^2 v_k - \lambda_{k+1}^2 v_{k+1}) \geq ||\lambda_{k+1}y_{k+2} - (\lambda_{k+1}-1)y_{k+1} - x^*||^2 - ||\lambda_{k+1}x_{k+1} + (1-\lambda_{k+1})y_{k+1} - x^*||^2$$

y basta tener en cuenta la expresión de u_k para obtener que el minuendo de la segunda parte de la última desigualdad es exactamente u_{k+1}, mientras que el sustraendo es u_k (esto no se ve tan claramente, pero se puede demostrar comprobando que $\frac{a-c-u_k}{\lambda_{k+1}} = 0$ a partir de la definición de FISTA).

□

Los dos siguientes lemas se prueban trivialmente por inducción sobre k. El primero de ellos es un resultado sobre sucesiones de términos positivos.

Lema 3.9 Sean $\{a_k\}$ y $\{b_k\}$ sendas sucesiones de números reales y positivos y $c > 0$ que satisfacen

$$\begin{aligned} a_1 + b_1 &\leq c \\ a_k - a_{k+1} &\geq b_{k+1} - b_k, \quad \forall k \geq 1 \end{aligned}$$

entonces $a_k \leq c$ para todo $k \geq 1$.

Mientras que el segunda da una cota inferior para la sucesión definida en la descripción del método FISTA.

Lema 3.10 La sucesión $\{\lambda_k\}$ dada por el método FISTA satisface que

$$\lambda_k \geq \frac{k+1}{2}, \quad \forall k \geq 1$$

Usando estos dos resultados es sencillo probar el siguiente teorema que prueba la buena convergencia de FISTA.

Teorema 3.11 El método FISTA con $\eta = \frac{1}{\beta}$ satisface

$$f(y_t) + g(y_t) - (f(x^*) - g(x^*)) \leq \frac{2\beta||x_1 - x^*||^2}{t^2} \tag{3.2.6}$$

Dem: Se definen

$$a_k := \frac{2}{\beta}\lambda_k^2 v_k, \quad b_k := ||u_k||^2, \quad c := ||y_1 - x^*||^2 = ||x_1 - x^*||^2$$

donde v_k y u_k son los mismos que en el lema 3.8. Según este $a_k - a_{k+1} \geq b_{k+1} - b_k$. Luego asumiendo que $a_1 + b_1 \leq c$ el lema 3.9 garantiza que

$$\frac{2}{\beta}\lambda_k^2 v_k \leq ||x_1 - x^*||^2$$

Teniendo en cuenta que $\lambda_k \geq \frac{k+1}{2}$ (Lema 3.10)

$$v_k \leq \frac{2\beta||x_1 - x^*||^2}{(k+1)^2}$$

Solo resta probar la validez de $a_1 + b_1 \leq c$. Dado que $\lambda_1 = 1$

$$a_1 = \frac{2}{\beta}v_1, \quad b_1 = ||u_1||^2 = ||y_2 - x^*||^2$$

Se aplica el lema 3.5 a los puntos $x = x^*$ y $y = x_1$ para obtener

$$F(x^*) - F(P_\beta(x_1)) \geq \frac{\beta}{2}||P_\beta(x_1) - x_1||^2 + \beta(x_1 - x^*)^T(P_\beta(x_1) - x_1)$$

Nótese que $y_2 = P_\beta(x_1)$ y, por lo tanto

$$\begin{aligned} F(x^*) - F(y_2) &\geq \frac{\beta}{2}||y_2 - x_1||^2 + \beta(x_1 - x^*)^T(y_2 - x_1) \\ &= \frac{\beta}{2}\left(||y_2 - x^*||^2 - ||x_1 - x^*||^2\right) \end{aligned}$$

y en consecuencia

$$\frac{2}{\beta}v_1 \leq ||x_1 - x^*||^2 - ||y_2 - x^*||^2$$

o lo que es lo mismo, $a_1 \leq c - b_1$, lo cual finaliza la demostración.

□

En cuanto a la implementación de FISTA se da el mismo problema que en su predecesor. La minimización del primer paso del método está supeditada a la función g. Solo si esta ofrece una simplicidad suficiente será posible llevar a cabo el método. Análogamente a ISTA, una función separable facilita notablemente la minimización, y, en concreto $g(x) = \lambda||x||_1$, ofrece un comportamiento ejemplar. Se analizará de nuevo el problema LASSO.

Ejemplo 3.12 De nuevo se trata de resolver

$$\min_{\beta \in \mathbb{R}^p} \left\{||y - X\beta||_2^2 + \lambda||\beta||_1\right\}$$

Basta simplemente reutilizar todo lo hecho en el ejemplo 3.7. La función a optimizar en el primer paso del método es $f(x) = \lambda|x| + \frac{1}{2\eta}(x - x_0)^2$ que alcanza el mínimo en $x^* = S_{\eta\lambda(x_0)}$, exactamente igual que en el ejemplo 4.7, y por lo tanto (3.2.1) conduce a

$$\beta_{t+1}(k) = S_{\eta\lambda}\left(\beta_t(k) + \eta\frac{\partial f}{\partial \beta_k}(\beta_t)\right)$$

que es equivalente a

$$\beta_{t+1}(k) = S_{\eta\lambda}\left(\beta_t(k) + 2\eta X(:,k)^T(y - X\beta_t)\right)$$

Hasta ahora el coste computacional por iteración es idéntico al de ISTA. Falta, no obstante, calcular (3.2.2), que supone $2p$ productos y p sumas adicionales por iteración a cambio de ofrecer la mejora vista en el orden de convergencia.

Capítulo 4

Implementación práctica del problema LASSO

En este capítulo se procede a analizar uno de los problemas de mayor interés en el aprendizaje automático, el cual ya se ha utilizado como ejemplo de implementación del descenso condicional de Frank Wolfe y de los algoritmos ISTA y FISTA de la sección anterior: el problema LASSO. El objetivo final es comparar la velocidad de convergencia de dos métodos de optimización. El primero de ellos será el descenso múltiple por coordenadas, que ha sido utilizado históricamente para resolver este problema; el otro, uno de los que se han analizado en este texto de forma teórica, FISTA. Se espera que este segundo mejore la convergencia del primero dado que aprovecha la convexidad de la función objetivo y la suavidad de una de sus partes como ya se ha observado. En la sección 4.1 se introduce el problema LASSO como una regularización del problema de mínimos cuadrados para el modelo de regresión lineal normal, se justifica su importancia en el análisis de datos y se discuten formulaciones para implementarlo de la manera más sencilla posible; en la sección 4.2 se presenta el descenso múltiple por coordenadas y en la sección 4.3 se implementan ambos algoritmos para resolver problemas idénticos analizando la velocidad de convergencia de cada uno de ellos.

4.1. Regresión lineal regularizada

A lo largo de este texto se ha descrito el funcionamiento de varios de los métodos expuestos para un problema concreto: el problema LASSO. Sin embargo en ningún momento se han dado las razones por las que este problema merece tanta atención. El LASSO deriva de uno de los grandes conocidos en estadística y análisis numérico, la regresión lineal para el problema de mínimos cuadrados.

Se supone que el valor de una variable respuesta y depende de los p valores que toman otras tantas variables predictoras $x^1, \dots, x^p$ y que se han observado N valores y_i en respuesta a las correspondientes p-uplas $x_i = (x_{i1} \dots x_{ip})$, $i = 1 \dots N$. El objetivo es predecir nuevos valores de la variable respuesta basándose en la experiencia previa. Para ello el conocido como modelo de regresión lineal asume una correspondencia lineal entre y y $x^1, \dots, x^p$ en función de los valores a priori desconocidos $\beta_0, \beta_1 \dots \beta_p$ y añade un error aleatorio e que puede deberse, por ejemplo, a

fallos de medición.

$$y = \beta_0 + \sum_{j=1}^{p} \beta_j x^j + e$$

En tales circunstancias los datos observados verifican

$$y_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ij} + e_i$$

Una forma clásica para abordar este modelo es mediante mínimos cuadrados. La idea consiste en estimar los valores de $\beta_0, \beta_1 \dots \beta_p$ haciendo mínima la norma del vector error $e = (e_1, \dots, e_N)$ formado por los errores de las observaciones previas.

$$\min_{\beta_0,\beta} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2$$

Se denotará por $X \in \mathcal{M}_{N\times p}$ la matriz de filas cada una de las realizaciones x_i y por y, tanto el vector de componentes los valores y_i como la variable respuesta. x^j hará referencia indistintamente también tanto a la columna j-ésima de X como a la variable predictora j-ésima. Concedánse estos abusos de notación. Con ánimo de simplificar la notación se prescindirá, sin pérdida de generalidad, de β_0 ya que tan solo aporta una traslación a la dependencia lineal de y respecto de $x^1, \dots, x^p$. Tras tales consideraciones se llega finalmente a la forma matricial simplificada para el problema de mínimos cuadrados

$$\min_{\beta} ||y - X\beta||_2^2 \tag{4.1.1}$$

Limitándose en primer lugar a la situación en la que $p \leq N$, se supone que $x^1, \dots, x^p$ son linealmente independientes. En este caso la solución de 4.1.1 existe y es única. Precisamente se trata de la proyeción de y sobre el subespacio lineal $L_X = \text{span}\ (x^1, \dots, x^p)$. Sea $\hat{y} = X\hat{\beta}$ esta proyección, $y - \hat{y}$ se caracteriza por ser ortogonal a L_X, es decir

$$\langle y - X\hat{\beta}, x_i \rangle = 0 \quad i = 1 \dots N$$

equivalentemente se tiene

$$\langle y, x_i \rangle = \langle X\hat{\beta}, x_i \rangle \quad i = 1 \dots N$$

o lo que es lo mismo,

$$X^T X\hat{\beta} = X^T y$$

Así pues la solución se obtiene en un solo paso siendo $\hat{\beta} = (X^T X)^{-1} X^T y$ ya que $(X^T X)^{-1}$ existe pues se ha supuesto que X tiene columnas linealmente independientes. Para obtener $\hat{\beta}$ en realidad se resuelven las ecuaciones $X^T X\hat{\beta} = X^T y$, conocidas como ecuaciones normales, evitando así el cálculo de una matriz inversa.

Si las columas de X no fueran linealmente independientes ocurriría que una de las variables

predictoras depende linearmente de las demás, y sería redundante considerarla al no influir directamente en la respuesta y. Por lo tanto, se evitará esta situación suponiendo que $x^1, \ldots, x^p$ son linealmente independientes.

La estimación de $\hat{\beta}$ a partir de mínimos cuadrados queda vinculada a los errores e_i en la variable respuesta y. Es habitual considerar el modelo de regresión lineal normal donde los errores se comportan como variables aleatorias independientes igualmente distribuidas con una distribución normal de media 0 y varianza σ^2. En ese caso considerando y como un vector aleatorio y dado que $y = X\beta + e$ con $e \sim \mathrm{N}(0, \sigma^2\mathcal{I}_N)$, su media y matriz de covarianza se obtienen a partir de las de e y son

$$\mathrm{E}(y) = X\beta \quad , \quad \mathrm{Var}(y) = \sigma^2\mathcal{I}_N$$

Y como $\hat{\beta} = (X^TX)^{-1}X^Ty$ se llega a que $\hat{\beta}$ es un estimador insesgado y con matriz de covarianza $(X^TX)^{-1}\sigma^2$

$$\begin{aligned} \mathrm{E}(\hat{\beta}) &= \mathrm{E}((X^TX)^{-1}X^Ty) = (X^TX)^{-1}X^TX\beta = \beta \\ \mathrm{Var}(\hat{\beta}) &= \mathrm{Var}((X^TX)^{-1}X^Ty)) = (X^TX)^{-1}X^T\sigma^2((X^TX)^{-1}X^T)^T \\ &= (X^TX)^{-1}(X^TX)(\sigma^2)^T((X^TX)^{-1})^T = (X^TX)^{-1}\sigma^2 \end{aligned}$$

La varianza refleja que ante una matriz X^TX cercana a ser singular, el valor estimado es muy sensible a los errores e de variable de respuesta. Evidentemente esto es algo que se desearía evitar y una de las formas de hacerlo es reguralizar el problema de mínimos cuadrados para la regresión lineal estimando β mediante una minimización cuadrática del error penalizando a su vez normas de β excesivamente elevadas. Si se entiende por norma, la norma usual l_2 se habla de regresión ridge, mientras que si por se trabaja con la norma l_1 se llega al problema LASSO. Se abordará el primero en 4.1.1 y el segundo, que será el que finalmente se implemente, en 4.1.2.

4.1.1. La regresión ridge

Con la misma notación que el modelo de regresión lineal, el conocido como modelo de regresión ridge estima la variable de coeficientes β con una restricción sobre la norma de esta en función de $t > 0$.

$$\min_{\beta} \left\{ ||y - X\beta||^2 \right\} \quad \text{s.a.} \quad ||\beta||_2^2 \leq t \tag{4.1.2}$$

Existe una reformulación de (4.1.2) en términos de una variable de control $\lambda > 0$ que da lugar a la conocida como forma lagrangiana del problema

$$\min_{\beta} \left\{ ||y - X\beta||^2 + \lambda||\beta||_2^2 \right\} \tag{4.1.3}$$

Ambas son formas equivalentes de escribir el mismo problema, como se probará en la proposición 4.1. Esta hace uso del concepto de lagrangiano de una función y de las condiciones de Karush-Kuhn-Tucker para el problema general de optimización. Sea D un conjunto abierto de $\mathbb{R}^n$ y sean $f : D \to \mathbb{R}$ la función objetivo a minimizar; $g_i : D \to \mathbb{R}$, $i = 1, \ldots s$ las restricciones de desigualdad y $h_j(x) : D \to \mathbb{R}$, $j = 1, \ldots t$ las restricciones de igualdad. Todas ellas continuamente diferenciables en D. Se considera el problema (P)

$$\begin{aligned} \min f(x) \quad & \text{s.a.} \\ & g_i(x) \leq 0 \quad i = 1, \ldots, s \\ & h_j(x) = 0 \quad j = 1, \ldots, t \end{aligned}$$

Sea x^* un punto que cumple las restricciones de igualdad y desigualdad ($g_i(x^*) \leq 0, \ \forall i \in \{1,\ldots,s\}$, $h_j(x^*) = 0, \ \forall j \in \{1,\ldots,t\}$). x^* es un punto de mínimo global para el problema de optimización general, si y solo si, existen constantes $u_i \geq 0$ $(i = 1,\ldots,s)$ y v_j $(j = 1,\ldots,t)$ tales que:

$$\nabla f(x^*) + \sum_{i=1}^{s} \nabla g_i(x^*) + \sum_{j=1}^{t} \nabla h_j(x^*) = 0$$
$$u_i g_i(x^*) = 0 \quad i = 1,\ldots s$$

Estas últimas se conocen como condiciones de Karush-Kuhn-Tucker (KKT) y la función $L(x, u_1,\ldots,u_s, v_1,\ldots,v_t)$ se denomina *lagrangiano* asociado al problema P.

El problema descrito al detalle y la demostración del resultado se puede ver en [1], concretamente se corresponde con el teorema 3.4 de esta referencia

Proposición 4.1 Los problemas (4.1.2) y (4.1.3) son equivalentes. Además, si $\hat{\beta}_{\text{ridge}}$ es solución de (4.1.3) para cierto λ, también lo será de (4.1.2) para $t = ||\hat{\beta}_{\text{ridge}}||^2$.

Dem: Se aplicarán las condiciones de Karush-Kuhn-Tucker (KKT) a cada uno de los problemas para obtener las condiciones necesarias y suficientes que ha de cumplir β para ser solución óptima del problema, y se verá que las soluciones de (4.1.2) son soluciones de (4.1.3) y viceversa.

Las condiciones KKT aplicadas a (4.1.2) establecen que β es solución del problema, si y solo si, existen $\mu_1, \mu_2 \in \mathbb{R}$ constantes no negativas tales que se cumplen las siguientes condiciones

$$\begin{aligned} (1) \quad & \mu_2\beta - \mu_1 X^T(y - X\beta) = 0 \\ & \mu_1 + \mu_2 > 0 \quad \text{para cada} \\ & \mu_2(||\beta||_2^2 - t) = 0 \end{aligned}$$

Las condiciones KKT aplicadas a (4.1.3) son equivalentes a las condiciones necesarias de mínimo para una función diferenciable, luego establecen que β es solución del problema, si y solo si, lo es también de

$$(2) \qquad 2\lambda\beta - X^T(y - X\beta) = 0$$

Tanto en (1) como en (2) ha sido necesario derivar la función $||Y - X\beta||_2^2$ respecto a β. Teniendo en cuenta que denotando $f(\beta) = ||Y - X\beta||^2$, $\frac{\partial f}{\beta_k}(\beta) = -2\sum_{i=1}^{N}\left[X(i,k)(y(i) - \sum_{j=1}^{p} X_{i,j}\beta(j))\right]$ y se obtiene fácilmente que el gradiente se puede escribir como $\nabla f(\beta) = -2X^T(y - X\beta)$.

Por otra parte nótese que en (1) se tiene que $\mu_1 > 0$, pues si no fuera así, μ_2 sería estrictamente positiva por la segunda de las condiciones, y por lo tanto $\hat{\beta} = \vec{0}$; pero entonces no se cumpliría $\mu_2(||\beta||_2^2 - t) = 0$.

Así pues si $(\hat{\beta}, \hat{\mu}_1, \hat{\mu}_2)$ es solución de (1) basta tomar $\beta = \hat{\beta}$ y $\lambda = \frac{\hat{\mu}_2}{2\hat{\mu}_1}$ para obtener una solución de (2).

Y si $\hat{\beta}$ es la solución de (2) para un cierto λ basta tomar en (1) $\beta = \hat{\beta}$, $\mu_1 = 1$, $\mu_2 = \lambda$ y $t = ||\hat{\beta}||_2^2$ para obtener una solución de (1).

□

Partiendo del problema en la forma dada por (4.1.3) es posible obtener una solución analítica fácilmente. De la expresión de (2) en la demostración de la proposición 4.1 se deduce directamente que $\hat{\beta}_{\text{ridge}} = (X^T X + 2\lambda \mathcal{I}_p)^{-1} X^T y$.

Si se analiza la expresión de $\hat{\beta}_{\text{ridge}}$ con más detenimiento se llega a dos conclusiones acerca de la regresión ridge. La primera es que $x^T(X^T X + 2\lambda \mathcal{I}_p)x = x^T X^T X x + \lambda ||x||^2 > 0$ si $x \neq \vec{0}$, es decir, $X^T X + 2\lambda \mathcal{I}_p$ es una matriz definida positiva y por lo tanto regular, evitando así el problema de lidiar con una matriz que puede ser singular. Y la segunda es que la regresión ridge no es sensible a errores de medición siempre y cuando se elija convenientemente λ.

Si se considera el modelo estadístico anterior con $e \sim \text{N}(0, \sigma^2 \mathcal{I}_N)$ se tiene como entonces $y = X\beta + e$, $\text{E}(y) = X\beta$, $\text{Var}(y) = \sigma^2 \mathcal{I}_N$. Sin embargo, ahora el estimador es ligeramente distinto, $\hat{\beta}_{\text{ridge}} = (X^T X + 2\lambda \mathcal{I}_p)^{-1} X^T y$.

$$\begin{aligned} \text{E}(\hat{\beta}_{\text{ridge}}) &= (X^T X + 2\lambda I_p))^{-1} X^T X \beta \\ \text{Var}(\hat{\beta}_{\text{ridge}}) &= (X^T X + 2\lambda I_p))^{-1} X^T \sigma^2 X ((X^T X + 2\lambda I_p))^{-1})^T \end{aligned}$$

y la expresión de la varianza para el estimador, aunque se antoja complicada, se puede ajustar con la variable de control λ evitando que errores pequeños produzcan grandes cambios en la estimación.

Hasta ahora no se ha analizado, sin embargo una situación muy habitual a la hora de afrontar un problema real. En muchos casos se quiere predecir el comportamiento de la variable de salida y dado el desconocimiento acerca de su comportamiento se elige un número de variables predictoras considerablemente extenso cuando, sin embargo no se cuenta con demasiada información previa, es decir, es frecuente que $p >> N$. Si esto sucede, tanto la regresión lineal (4.1.1) como la regresión ridge en cualquiera de sus formas (4.1.2) o (4.1.3), a pesar de llegar a una solución única, no alcanzan una estimación suficientemente fiable. Además, dado el elevado número de variables predictoras sería interesante obtener un estimador que reflejara claramente cuáles de estas variables son verdaderamente influyentes a la hora de predecir la variable de salida, es decir, resultaría conveniente que el estimador tuviera gran parte de sus componentes nulas, pero esto no es lo habitual en la regresión ridge. Sí lo será sin embargo si en (4.1.2) se elige la norma l_1 a la hora de limitar el tamaño de β. Esto da lugar a un nuevo modelo conocido como LASSO.

4.1.2. LASSO

El método LASSO (Least Absolute Shrinkage and Selection Operator) también conocido como método de *regresión generalizada* l_1 trata de penalizar, al igual que la regresión ridge, un tamaño grande de β en el problema de mínimos cuadrados para la regresión lineal. La diferencia con el método anterior es que interpreta el tamaño de β en función de la norma l_1. Lo que en principio parece una desventaja, pues esta restricción viene dada a partir de una función no diferenciable en los puntos de $\mathbb{R}^p$ con alguna componente nula, proporciona por otro lado una mayor interpretabilidad del problema al proporcionar para el caso $p >> N$ soluciones con un alto

número de variables predictoras con valor nulo, lo que se conoce como 'selección de variables'. La estimación $\hat{\beta}_{LASSO}$ viene dada por

$$\min_{\beta} \left\{ ||y - X\beta||^2 \right\} \quad \text{s.a.} \quad ||\beta||_1^2 \leq t \tag{4.1.4}$$

Al igual que la regresión ridge, LASSO admite una reformulación en la forma lagrangiana en términos de una varible de control $\lambda > 0$.

$$\min_{\beta \in \mathbb{R}^p} \left\{ ||y - X\beta||_2^2 + \lambda ||\beta||_1 \right\} \tag{4.1.5}$$

Proposición 4.1 Los problemas (4.1.4) y (4.1.5) son equivalentes. Además, si $\hat{\beta}_{\text{LASSO}}$ es solución de (4.1.5) para cierto λ, también lo será de (4.1.4) para $t = ||\hat{\beta}_{\text{LASSO}}||^2$.

Dem: Para $\beta \in \mathbb{R}^p$ se definen β^+ y β^-, también vectores de $\mathbb{R}^p$, cuyas compentes son respectivamente $\beta_i^+ = \text{máx}\{\beta_i, 0\}$ y $\beta_i^- = \text{máx}\{-\beta_i, 0\}$, es decir, la parte positiva y la negativa de las componentes de β. Así pues se cumple que $\beta_i = \beta_i^+ - \beta_i^-$ y $|\beta_i| = \beta_i^+ + \beta_i^-$ y por ende también $\beta = \beta^+ - \beta^-$. Por otra parte $\vec{1}$ denotará el vector unidad en $(1 \dots 1) \in \mathbb{R}^p$.

El procedimiento de la demostración consiste en obtener las condiciones de Karush-Kuhn-Tucker (KKT) de cada una de las dos formas del problema LASSO reformuladas a partir de las partes positivas y negativas. Nótese que la razón de esto reside en la falta de diferenciabilidad de las funciones objetivo expresadas en función de β. Una vez calculadas estas se probará que las soluciones de un problema constituyen soluciones para el otro y viceversa.

Se parte de la forma lagrangiana (4.1.5) que es equivalente a (1)

$$\begin{array}{ll} \min_{\beta^+, \beta^-} & \left\{ \sum_{i=1}^{N} \left(y_i - x_i\beta^+ + x_i\beta^- \right)^2 + \lambda \sum_{j=1}^{p} (\beta_j^+ + \beta_j^-) \right\} \\ \text{s.a} & -\beta^+ \leq 0 \\ & -\beta^- \leq 0 \end{array}$$

El langrangiano es en este caso

$$\mathcal{L}(\beta^+, \beta^-, u^+, u^-) = \sum_{i=1}^{N} \left(y_i - x_i^T\beta^+ + x_i^T\beta^- \right)^2 + \lambda \sum_{j=1}^{p} (\beta_j^+ + \beta_j^-) - u^+\beta^+ - u^-\beta^-$$

y las condiciones de KKT de (1)

$$\begin{array}{ll} 2\sum_{i=1}^{N} & \left(y_i - x_i^T\beta^+ + x_i^T\beta^- \right)(-x_i) + \lambda\vec{1} - u^+ = 0 \\ 2\sum_{i=1}^{N} & \left(y_i - x_i^T\beta^+ + x_i^T\beta^- \right) x_i + \lambda\vec{1} - u^- = 0 \\ u^+\beta^+ & = 0 \\ u^-\beta^- & = 0 \\ u^+, u^- & \in \mathbb{R} \text{ tales que } u^+, u^- \geq 0 \end{array}$$

De manera análoga, la forma clásica (4.1.4) es equivalente a (2)

$$\begin{aligned} \text{mín}_{\beta^+,\beta^-} \quad & \left\{ \sum_{i=1}^{N} \left(y_i - x_i\beta^+ + x_i\beta^- \right)^2 \right\} \\ \text{s.a} \quad & \sum_{j=1}^{p} (\beta_j^+ + \beta_j^-) - t \leq 0 \\ & -\beta^+ \leq 0 \\ & -\beta^- \leq 0 \end{aligned}$$

El langrangiano es ahora

$$\mathcal{L}(\beta^+, \beta^-, v, v^+, v^-) = \sum_{i=1}^{N} \left(y_i - x_i^T\beta^+ + x_i^T\beta^- \right)^2 + v \sum_{j=1}^{p} (\beta_j^+ + \beta_j^- - t) - v^+\beta^+ - v^-\beta^-$$

y las condiciones de KKT de (2)

$$\begin{array}{ll} 2\sum_{i=1}^{N} & \left(y_i - x_i^T\beta^+ + x_i^T\beta^- \right)(-x_i) + v\vec{1} - v^+ = 0 \\ 2\sum_{i=1}^{N} & \left(y_i - x_i^T\beta^+ + x_i^T\beta^- \right) x_i + v\vec{1} - v^- = 0 \\ v(\sum_{j=1}^{p} & (\beta_j^+ + \beta_j^-) - t) = 0 \\ v^+\beta^+ & = 0 \\ v^-\beta^- & = 0 \\ v^+, v^- & \in \mathbb{R} \text{ tales que } v^+, v^- \geq 0 \\ v & \in \mathbb{R} \text{ y } v \geq 0 \end{array}$$

Sea $(\beta_\lambda^+, \beta_\lambda^-, u_\lambda^+, u_\lambda^-)$ solución de las condiciones KKT de (1). Basta elegir $v = \lambda$, $u^+ = v_\lambda^+$, $u^- = v_\lambda^-$ y $t = \sum_{j=1}^{p}(\beta_{j\lambda}^+ + \beta_{j\lambda}^-)$ para obtener una solución para las condiciones (2). Esto quiere decir que el problema LASSO en forma lagrangiana para un cierto λ es equivalente al problema en forma clásica con $t = \sum_{j=1}^{p}(\beta_{j\lambda}^+ + \beta_{j\lambda}^-)$ siendo β_λ la solución del primero.

Y sea $(\beta_t^+, \beta_t^-, v_t, v_t^+, v_t^-)$ solución de las condiciones KKT de (2). Si se eligen $\lambda = v_t$, $u^+ = v_t^+$ y $u^- = v_t^-$ se obtiene una solución para las condiciones KKT de (1).

□

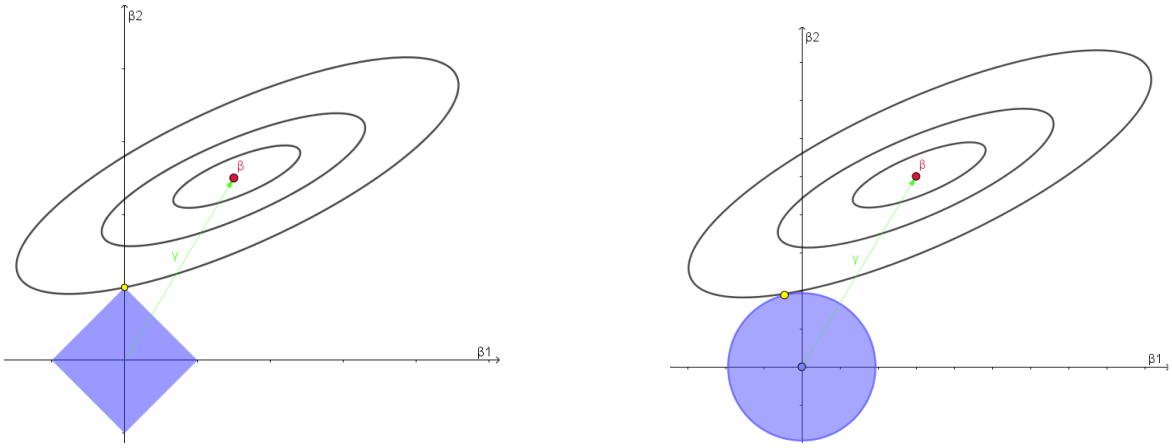
El motivo de añadir esta restricción y, en concreto, de expresarla en términos de la norma l_1 es por supuesto razonable y se debe a la búsqueda de simplicidad en las variables predictoras. Actualmente en muchos campos de la ciencia y de la industria se analizan conjuntos de datos que dependen de un número elevado de varibles predictoras. Este número puede incluso superar claramente al número de observaciones que se manejan, $p >> N$. Una optimización por mínimos cuadrados tanto para la regresión lineal como para la regresión ridge producirá generalmente una estimación del parámetro β tal que β_i es distinto de cero en la mayoría de los casos y esto dificulta notablemente la interpretación del comportamiento del sistema en cuestión. Resulta así complicado determinar cuáles son las variables predictoras verdaderamente influyentes.

Es aquí donde la restricción del LASSO juega un papel fundamental. Resulta que si se elige t suficientemente pequeño los valores β_i producidos son en su mayoría nulos, siendo solo unos pocos

distintos de cero. Este fenómeno se conoce como *dispersión* y permite simplificar la interpretación de los resultados. Es lógico plantearse la elección de otra norma l_q, sin embargo, ocurre que la elección de $q = 1$ es la justa si lo que se busca es un problema disperso y a la vez convexo. Si $q > 1$ se pierde la propiedad de dispersión y, si $q < 1$, la dispersión se mantiene pero el problema deja de ser convexo. Las ventajas que ofrece la convexidad se han justificado ya lo suficiente como para plantearse perder esta propiedad.

En la Figura 4.1 se han representado en dos variables ($p = 2$) los errores cuadráticos de LASSO (4.1.4) y la regresión ridge (4.1.2) dados por $(y_1 - x_1\beta)^2 + (y_2 - x_2\beta)^2 = K$, con K constante, y las regiones admisibles $||\beta||_1^2$ y $||\beta||_2^2$ respectivas de cada uno de los problemas. Se aprecia claramente como en LASSO los contornos de las cónicas intersecan primero con alguno de los vértices de la región admisible, donde una de las variables predictoras es nula; mientras que en la regresión ridge, aunque se quedan cerca de hacer lo propio, finalmente intersecan en un punto donde ambas variables predictoras son no nulas. Si el número de variables predictoras es elevado esto se puede generalizar y ocurrirá lo que ya se ha comentado: la solución $\hat{\beta}_{\text{ridge}}$ tenderá a contar con componentes no nulas y esto dificultará la interpretación real del modelo.

Aclarada la relevancia de LASSO se procede a comentar ciertas consideraciones que van a facilitar



(a) LASSO (b) Regresión ridge

Figura 4.1: Interpretación de LASSO y regresión ridge

su resolución. En primer lugar es razonable dar por supuesta la posibilidad de elegir el valor de las variables de predicción. De esta manera si se toman como hipótesis

$$\sum_{i=1}^{N} x_{ij} = 0 \text{ y } \sum_{i=1}^{N} x_{ij}^2 = 1 \ \forall j \in \{1 \dots p\}$$

se habla de problema estandarizado. Y se habla de problema LASSO estandarizado con condiciones centradas cuando además

$$\sum_{i=1}^{N} y_i = 0$$

En este último caso es posible omitir β_0 en (4.1.1) ya que, como se justifica a continuación su

valor sería 0.

$$\begin{aligned}\sum_{i=1}^{N}\left(y_i-\beta_0-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2 &= \sum_{i=1}^{N}\left[\left(y_i-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2+\beta_0^2-2\beta_0\left(y_i-\sum_{j=1}^{p}\beta_j x_{ij}\right)\right]\\ &= \sum_{i=1}^{N}\left(y_i-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2+N\beta_0^2-2\beta_0\sum_{i=1}^{N}y_i\\ &\quad +2\beta_0\sum_{j=1}^{p}\beta_j\sum_{i=1}^{N}x_{ij}\\ &= \sum_{i=1}^{N}\left(y_i-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2+N\beta_0^2\end{aligned}$$

El mínimo de esta última expresión se alcanza claramente tomando $\beta_0 = 0$. Así pues, para el problema estandarizado con condiciones centradas

$$\min_{\beta_0,\beta}\left\{\sum_{i=1}^{N}\left(y_i-\beta_0-\sum_{j=1}^{p}x_{ij}\beta_j\right)^2\right\}$$

es equivalente a

$$\min_{\beta}\left\{\sum_{i=1}^{N}\left(y_i-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2\right\}$$

La hipótesis $\sum_{i=1}^{N} y_i = 0$ no es a priori razonable puesto que estaríamos asumiendo una información acerca de las respuestas que no se obtiene de manera natural. Sin embargo, si se ha resulto el problema estandarizado con condiciones centradas obteniendo como solución una determinada p-upla para $\hat{\beta}$, es fácil calcular ahora la solución para el problema estandarizado sin condiciones centradas (4.1.1) que, como se ha visto anteriormente, es equivalente a

$$\min_{\beta,\beta_0}\left\{\sum_{i=1}^{N}\left(y_i-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2+N\beta_0^2-2\beta_0\sum_{i=1}^{N}y_i+2\beta_0\sum_{j=1}^{p}\beta_j\sum_{i=1}^{N}x_{ij}\right\}$$

que a su vez es equivalente a

$$\min_{\beta}\sum_{i=1}^{N}\left(y_i-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2+\min_{\beta_0}\left\{N\beta_0^2-2\beta_0\sum_{i=1}^{N}y_i\right\}$$

Y por lo tanto, si β y β_0 denotan los minimizadores para el problema estandarizado sin condiciones centradas, se tiene que $\beta = \hat{\beta}$ y $\beta_0 = \frac{1}{N}\sum_{i=1}^{N} y_i$.

4.2. Descenso por coordenadas

Se supone en primer lugar que solo existe una variable predictora, es decir, $p = 1$. En este caso la matriz X será un vector $1 \times N$ traspuesto que denotaremos por z. De ahora en adelante se trabajará con la versión Lagrangiana de LASSO estandarizada con condiciones centradas si no se dice otra cosa. Se trata de resolver entonces

$$\min_{\beta} \left\{ ||y - \beta z||^2 + \lambda|\beta| \right\}$$

El problema de optimización es sobre una variable, con lo cual si denotamos $f(\beta) = ||y - \beta z||^2 + \lambda|\beta|$ resulta sencillo obtener una solución analítica igualando la derivada de f a 0 teniendo presente que en $\beta = 0$ esta no existe.

$$f'(\beta) = \begin{cases} -2(y - \beta z)^T z - \lambda = -2\langle y, z\rangle + 2\beta - \lambda & \text{si} \quad \beta < 0 \\ -2(y - \beta z)^T z + \lambda = -2\langle y, z\rangle + 2\beta + \lambda & \text{si} \quad \beta > 0 \end{cases}$$

De aquí se deduce

$$\hat{\beta} = \begin{cases} \langle y, z\rangle + \frac{\lambda}{2} & \text{si} \quad \langle y, z\rangle < -\frac{\lambda}{2} \\ 0 & \text{si} \quad -\frac{\lambda}{2} \leq \langle y, z\rangle \leq \frac{\lambda}{2} \\ \langle y, z\rangle - \frac{\lambda}{2} & \text{si} \quad \langle y, z\rangle > \frac{\lambda}{2} \end{cases} \tag{4.2.1}$$

Basta tener en cuenta que si $\langle y, z\rangle < \frac{\lambda}{2}$ entonces $f(\langle y, z\rangle + \frac{\lambda}{2}) > f(0)$ y que si $\langle y, z\rangle > \frac{\lambda}{2}$ entonces $f(\langle y, z\rangle - \frac{\lambda}{2}) > f(0)$. Se probará la primera afirmación, la segunda es análoga. Nótese que dado que el problema está estandarizada se cumple que $||z||^2 = 1$

$$\begin{aligned} f\left(\langle y, z\rangle + \frac{\lambda}{2}\right) &= ||y - \left(\langle y, z\rangle + \frac{\lambda}{2}\right) z||^2 + \lambda \left(\langle y, z\rangle + \frac{\lambda}{2}\right) \\ &= ||y||^2 + ||\langle y, z\rangle + \frac{\lambda}{2}||^2 - 2\left(\langle y, z\rangle + \frac{\lambda}{2}\right)\langle y, z\rangle + \lambda\left(\langle y, z\rangle + \frac{\lambda}{2}\right) \\ &= f(0) + (\langle y, z\rangle + \frac{\lambda}{2})\left(\left(\langle y, z\rangle + \frac{\lambda}{2}\right) - 2\left(\langle y, z\rangle - \frac{\lambda}{2}\right)\right) \\ &= f(0) + \left(\langle y, z\rangle + \frac{\lambda}{2}\right)^2 > f(0) \end{aligned}$$

Se define el operador

$$S_\lambda(x) = sign(x)(|x| - \lambda)_+$$

S_λ actúa sobre x reduciendo su valor en λ cuando este es mayor que el propio λ en valor absoluto y positvo, aumentándolo cuando es mayor que λ en valor absoluto y negativo estableciéndolo en cero cuando es más pequeño en valor absoluto. Haciendo uso de este operador es posible expresar $\hat{\beta}$ de manera más concisa como $\hat{\beta} = S_{\frac{\lambda}{2}}(\langle y, z\rangle)$.

La idea ahora es trasladar lo que se ha hecho en una variable para el caso en el que el número de varibles predictoras es mayor que uno, en el cual, un análisis analítico se antojaría como mínimo complicado. El procedimiento consistirá en elegir un orden de las variables predictoras (sea $1, 2 \ldots p$ sin pérdida de generalidad) Y para cada índice k, variando su valor según el orden establecido, minimizar la función objetivo en β_k manteniendo fijas el resto de variables.

Consideramos la función objetivo como una función dependiente solo de β_k.

$$\begin{aligned} f(\beta) &= f(\beta_k) = \sum_{i=1}^{N} \left(y_i - \sum_{\substack{j=1 \\ j\neq k}}^{p} X(i,j)\beta_j - X(i,k)\beta_k \right) + \lambda \sum_{\substack{j=1 \\ j\neq k}}^{p} |\beta_j| + \lambda|\beta_k| \\ &= \sum_{i=1}^{N} \left(r_i^k - X(i,k)\beta_k \right) + \lambda \sum_{\substack{j=1 \\ j\neq k}}^{p} |\beta_j| + \lambda|\beta_k| \end{aligned}$$

donde se ha denotado por r_i^k y se denomina residuo a la cantidad $r_i^k = y_i - \sum_{\substack{j=1 \\ j\neq k}}^{p} X(i,j)\beta_j$. Minimizar la expresión anterior equivale a minizar la misma función que en el caso en el que tan solo existía una única variable predictora con $y_i = r_i^k$ y $z_i = X(i,k)$. Luego se obtiene fácilmente que el mínimo se alcanza en $\hat{\beta}_k = S_{\frac{\lambda}{2}}(\langle r^k, X(:,k)\rangle)$. El algoritmo se iteraría un número n de veces. En cada una de ellas k habría de recorrer cada una de las variables predictoras para actualizar la componente β_k obteniendo finalmente un β actualizado que se iría acercando al verdadero valor del minimizador $\hat{\beta}$. El hecho de que este algoritmo actúe en cada iteración componente por componente sugiere que la convergencia va ser más lenta que la de métodos que actualizan β de una sola vez, como es el caso de FISTA. No obstante se comprobará con el ejemplo práctico en la siguiente sección.

4.3. Resolución del problema LASSO en R

Se implementarán en esta sección los dos métodos mencionados, FISTA y descenso por coordenadas, para resolver el problema LASSO estandarizado con condiciones normales partiendo de la versión lagrangiana en forma matricial (4.1.5)

$$\min_{\beta\in\mathbb{R}^p} \left\{ ||y - X\beta||_2^2 + \lambda||\beta||_1 \right\}$$

En la subsección 4.3.1 se explicará cómo se va a definir la matriz de datos X y el vector respuesta y que caracterizan LASSO, mientras que en las subsecciones 4.3.2 y 4.3.3 se implementarán FISTA y descenso por coordenadas respectivamente. Ambos se ejecutarán para un idéntico problema y realizarán un número de iteraciones que dependerá de una tolerancia, fijada previamente, en función de la diferencia entre valores de la función f para iterantes consecutivos. Esta convención es necesaria ya que no se conoce el mínimo real de (4.1.5). Por último en la subsección 4.3.4 se compararán los resultados obtenidos.

4.3.1. Definición del problema estandarizado con condiciones centradas

Se dirá que un vector está estandarizado si la suma de sus componentes es 0 y la suma de sus componentes elevadas al cuadrado, 1. Asímismo se entenderá por matriz estandarizada aquella

cuyas columnas están estandarizadas.

La matriz X se creará de manera aleatoria a partir de valores reales entre 0 y 1 y seguidamente se estandarizará restando a cada columnna su media y dividiendo la columna resultante entre el valor de la suma al cuadrado de sus componentes (nótese que esto la varianza de la columna multiplicada bien por el factor $(N-1)$ o bien por N según se defina esta). El vector respuesta será exactamente $y = X\hat{\beta}_{LS} + e$, es decir, se definirá a partir de $\hat{\beta}_{LS} \in \mathbb{R}^p$ y de un error aleatorio. Se tomarán determinadas componentes de $\hat{\beta}_{LS}$ claramente mayores en valor absoluto que las demás, entendiendo que estas corresponden a las variables predictoras verdaderamente influyentes en el fenómeno en cuestión y esperando que al incluir la regularización l_1 que se propone en LASSO se seleccionen rápidamente gran parte de ellas como variables influyentes del modelo, es decir, que se obtenga un estimador $\hat{\beta}_{\text{LASSO}}$ con gran parte de dichas componentes distintas de cero y el resto nulas. No debiera resultar preocupante el hecho de no tener unas condiciones centradas $\sum_{i=1}^{N} y_i = 0$ pues, como ya se ha analizado al término de la sección 4.1 el valor del estimador de β va a ser el mismo tanto en un caso como en otro. La diferencia entre ambos es que sin condiciones centradas aparecerá el término β_0 estimado como $\hat{\beta}_0 = \frac{1}{N}\sum_{i=1}^{N} y_i$. Sin embargo, la estandarización de la matriz va a dar lugar a un vector respuesta centrado ya que, tal y como se ha definido y,

$$\sum_{i=1}^{N} y_i = \beta_{LS}(1)\sum_{i=1}^{N} X(i,1) + \ldots \beta_{LS}(p)\sum_{i=1}^{N} X(i,p) = 0$$

Así pues, definidos N y p se utilizará la función **replicate** para crear una matriz con p columnas de vectores aleatorios de longitud N obtenidos mediante la finción **runif**. En definitiva se le pedirá a R que cree una matriz aleatoria $N \times p$. A continuación se le va a restar a cada elemento de esta matriz la media de la columna a la que pertenece y se le va a dividir entre la raíz cuadrada de la suma de las componentes de dicha columna elevados al cuadrado. Es trivial comprobar que la matriz así obtenida está estandarizada.

```
    Genera_matrizdatos <- function(N,p){
        X <- replicate(p,runif(N))
        X<-X-matrix(rep(apply(X,2,sum)/N,N),nrow=N,byrow= TRUE)
        X<-X/sqrt(matrix(rep(apply(X,2,function(x) sum(x^2)),N),nrow=N,byrow= TRUE))
        return(X)
}
```

Para definir el vector y se crea $\hat{\beta}_{LS}$ suponiendo que alrededor del 20 % de las variables predictoras son influyentes en el fenómeno en cuestión. Así se tomarán $\left[\frac{1}{10}\right]$ de sus componentes entre -1 y -0.5; $\left[\frac{1}{10}\right]$, entre 0.5 y 1; y el resto, directamente nulas. Todo esto en un orden arbitrario. Por último $y = X\hat{\beta}_{LS} + e$, con $e \sim N(0,1)$.

$[x]$ denota la parte entera de x. En el código de R esto equivale a escribir **floor(x)**.

```
    Genera_vectorrespuesta <-function(X){
        betaLS <- c(runif(floor(dim(x)[2]/10), min =-1, max=-0.5),
                    runif(floor(dim(x)[2]/10), min =.5, max=1),
```

```
                    rep(0,dim(x)[2] - 2*floor(dim(x)[2]/10)))
        orden<-sample(1:dim(x)[2],dim(x)[2],replace = FALSE)
        betaLS <- betaLS[orden]
        y<-X%*%betaLS + rnorm(N)
    return(y)
}
```

El siguiente paso será aplicar los métodos de optimización para resolver el problema LASSO.

4.3.2. Resolución mediante FISTA

Sean

$$\lambda_0 = 0\ ,\ \lambda_t = \frac{1+\sqrt{1+4\lambda_{t-1}^2}}{2}\ \text{y}\ \gamma_t = \frac{1-\lambda_t}{\lambda_{t+1}}\ ;$$

la longitud de paso de la hipótesis teorema 3.8 (óptima en referencia al orden de convergencia del metodo)

$$\eta = \frac{1}{\beta} = \frac{1}{2||X^TX||_2} = \frac{1}{2\sqrt{\text{máx}\{\lambda \in \Lambda((X^TX)^2)\}}}\ ;$$

donde $\Lambda(X)$ denota el conjunto de autovalores de la matriz X; y un iterante inicial $\beta_x^0 = \beta_y^0$.

El método FISTA realiza la siguiente iteración para resolver el problema LASSO

$$\beta_y^{t+1}(k) = S_{\eta\lambda}\left(\beta_x^t(k) + 2\eta X(:,k)^T(y - X\beta_x^t)\right) \tag{4.3.1}$$

$$\beta_x^{t+1}(k) = (1-\gamma_t)\beta_y^{t+1} + \gamma_t\beta_y^t \tag{4.3.2}$$

Para implementar el algoritmo en R se definen primero dos funciones. La primera no es otra cosa que el operador de reducción $S_\lambda(x) = sign(x)(|x| - \lambda)_+$ utilizado también en el descenso por coordenadas.

```
S <- function(lambda,x){
       valor<-0*(abs(x)<lambda)+(x - lambda)*(x>lambda)+
              (x + lambda)*(x< -lambda)
       return(valor)
      }
```

La segunda es la función de actualización que dado un itarante β_t calcula β_{t+1} correspondiente a la aplicación de FISTA con longitud de paso η para la resolución de LASSO con constante de reguralización igual a λ.

$$\beta_{t+1}(k) = S_{\eta\lambda}\left(\beta_t(k) + 2\eta X(:,k)^T(y - X\beta_t)\right)$$

```
updateFISTA<-function(X,y,betai,eta,lambda) {
    x0<-betai+2*eta*t(X)%*%(y-X%*%betai)
    betai<-sapply(1:p, function(i) S(eta*lambda,x0[i]))
```

```
        betaf<-betai
        return(betaf)
}
```

Se representará β_x^t por 'betaix'; β_y^t, por 'betaiy'; β_x^{t+1}, por 'betafx'; y β_y^{t+1}, por 'betafy'. De tal manera que una vez definidas la matriz de datos y el vector respuesta del problema; la constente de regularización de LASSO; y un iterante inicial del que partir, β_0, es posible ejecutar el algoritmo: se obtiene la longitud de paso óptima para FISTA, se inicializan las varibles adecuadas y se le da un valor a 'dif', que representa la diferencia entre iterantes consecutivos, para tener certeza de que el algoritmo entra en el bucle. Basta iterar, este hasta que la diferencia entre valores de la función objetivo para iterantes consecutivos revase inferiormente la tolerancia fijada. Nótese primero que se fuerza a la ejecución de al menos dos iteraciones, ya que es fácil comprobar que $\gamma_1 = 1$, y por lo tanto $\beta_x^1 = \beta_y^0 = \beta_x^0$. Entonces 'dif' se anula provocando la salida del bucle. Dentro del bucle simplemente se procede a realizar los cálculos descritos en (4.3.1) y (4.3.2) calculando'dif' en cada paso.

```
FISTA<-function(X,y,lambda,beta0,TOL){
    #Cálculo de 1/beta para obtener la longitud de paso óptima
    autovalores<-eigen(t(t(X)%*%X)%*%(t(X)%*%X))[[1]]
    eta <-1/(2*sqrt(max(autovalores)))
    betaix<-beta0
    betaiy<-beta0
    lambda_t <- 0
    iter<-0
    dif<-TOL +1 #para entrar en el bucle
    time<-proc.time()
    while((dif > TOL)||(iter<2)){
        gamma_t<- 1 -lambda_t
        lambda_t<-(1 + sqrt(1 + 4*lambda_t^2))/2
        gamma_t<- gamma_t/lambda_t
        betafy<-updateFISTA(X,y,betaix,eta,lambda)
        betafx<-(1 - gamma_t)*betafy + gamma_t*betaiy
        dif<-abs(valor(X,y,betafx,lambda)-valor(X,ybetaix,lambda))
        betaix<-betafx
        betaiy<-betafy
        iter<-iter+1
    }
    time<-proc.time()-time
    return(list(betafx,iter,time[[3]]))
}
```

La tolerancia, como ya se ha mencionado previamente, se calcula en función de la diferencia entre el valor de la función objetivo entre dos iterantes consecutivos. Para ello se emplea la función **valor**.

```
valor<-function(X,y,beta,lambda){
    sum((y-X%*%beta)^2)+lambda*sum(abs(beta))
}
```

4.3.3. Resolución mediante descenso múltiple por coordenadas

Se define la función de reducción exactamente igual que para FISTA, así como una función de actualización que para cada componente k-ésima de un iterante, β_k^t, calcula la componente k-ésima del iterante siguiente, β_k^{t+1}.

```
updateDMC <- function(X,y,betai,k){
            r <-y - X[,-k]%*%betai[-k]
            x0 <- sum(X[,k]*r)
            betaf<-betai
            betaf[k]<-S(lambda/2,x0)
            return(betaf)
          }
```

Se usará asímismo la función **valor** para calcular la diferencia entre los valores de la función objetivo en iterantes consecutivos.

Se parte del problema definido por la matriz de datos X y el vector respuesta y. Se fija un valor para la constante de regularización l_1 del problema LASSO, un iterante inicial y la tolerancia. Es necesario en este caso utilizar una variable que se ha denotado como 'betaprev' para guardar la información del iterante anterior y así calcular 'dif'. A partir de aquí basta actualizar beta coordenada a coordenada en cada paso y una vez recorridas las p coordenadas calcular 'dif' para comprobar si se ha superado la tolerancia o no. Hay que tener en cuenta que por cada iteración se están realizando p actualizaciones. Esto hace pensar que para valores elevados de la segunda dimensión de la matriz de datos este algoritmo podría resultar muy lento, pero esto se precisará en la subsección siguiente.

```
    DMC<-function(X,y,lambda,beta0,TOL){
        beta<-rep(0,dim(X)[2])
        beta_prev<-beta0
        dif<-abs(valor(X,y,beta,lambda)-valor(X,y,beta_prev,lambda))
        t<-0
        time<-proc.time()
        while (dif>TOL){
            for (j in 1:dim(X)[2]){
                beta<-updateDMC(X,y,beta,j,lambda)
            }
            dif<-abs(valor(beta,lambda)-valor(beta_prev,lambda))
            beta_prev<-beta
            t<-t+1
            time<-proc.time()-time
        }
  return(list(beta,dim(X)[2]*t,time[[3]]))
}
```

4.3.4. Comparación de los métodos

A partir de las funciones **Genera_matrizdatos** y **Genera_vectorrespuesta** vistas en la sección 4.3.1 se generarán las matrices que definen el problema de regresión lineal con un

número de filas fijo para la matriz de datos X, y por ende un tamaño fijo para el vector respuesta y, y un número de columnas para X, es decir, de variables predictoras, que se irá incrementando. El valor de λ y el de la tolerancia serán constantemente iguales a 1 y a 0.00001 respectivamente y se tomará un primer iterante con todas las componentes idénticamente iguales a 1 del tamaño p correspondiente al modelo en cuestión. Así, para cada problema se ejecutarán los algoritmos de FISTA Y DMC de las subsecciones previas guardando en un dataframe el número de iteraciones y el tiempo invertidos por ambos algoritmos. Obviamente se espera que estas características crezcan con p, pues el número de operaciones que se realizan será mayor, pero se observará una ralentización mucho más acentuada para DMC debido a que en cada iteración se actualizan cada una de las p componentes de β, siendo este precisamente el parámetro que se está incrementando. Con la información obtenida y almacenada se realizarán gráficos que ilustren esta situación.

Para estos propósitos se crean las funciones **simulacionFISTA** y **simulacionDMC**. Ambas tienen como entrada p, el número de columnas de X, y fijado el valor de λ, un número de filas para la matriz y una tolerancia generan la matriz de datos y el vector respuesta del problema y ejecutan **FISTA** y **DMC** respectivamente con un vector inicial con componentes idénticamente iguales a 1 de tamaño p guardando el número de iteraciones y el tiempo de ejecución. Estas funciones se ejecutan para una secuencia de valores de p que va desde 100 hasta 2000 aumentando de 100 en 100 y finalmente se genera un dataframe con la esta información.

```
lambda<-1
TOL<-0.00001
N<-20

simulacionFISTA<-function(p){
    X<-Genera_matrizdatos(N,p)
    y<-Genera_vectorrespuesta(X)
    lista<-FISTA(X,y,lambda,rep(1,p),TOL)
    tiempo<-lista[[3]]
    niter<-lista[[2]]
    return(c(tiempo,niter))
}

simulacionDMC<-function(p){
    X<-Genera_matrizdatos(N,p)
    y<-Genera_vectorrespuesta(X)
    lista<-DMC(X,y,lambda,rep(1,p),TOL)
    tiempo<-lista[[3]]
    niter<-lista[[2]]
    return(c(tiempo,niter))
}

fista<-sapply(seq(100,1000,by=100),simulacionFISTA)
dmc<-sapply(seq(100,1000,by=100),simulacionDMC)
```

```
simulacion<-data.frame(seq(100,1000,by=100),fista[1,],dmc[1,],fista[2,]
                       ,dmc[2,])
```

El resultado de la simulación se aprecia en el cuadro 4.1. El tiempo de ejecución de FISTA ha oscilado entre 0.17 segundos para $p = 200$ y 12.78 segundos para $p = 2000$ y se aprecia un aumento gradual aunque con pequeños descensos entre algunos casos.Estos valores son en todo caso asumibles desde un punto de vista práctico. DMC ha sido muy rápido para $p = 100$ (0.25 segundos), incluso para $p = 200$ y $p = 300$ (2.05 y 2.87 segundos respectivamente). A partir de ahí los tiempos comienzan a dispararse y alcanzan tiempos superiores a los 5 minutos a partir de $p = 1600$. Si se centra la atención en el número de iteraciones ocurre exactamente lo mismo. DMC realmente está realizando menos actualizaciones completas del estimador β_{LASSO} que FISTA, pero el problema es que para cada una de ellas está actualizando coordenada a coordenada.

p	tiempo FISTA	tiempo DMC	nºiter FISTA	nº iter DMC
100	0.28	0.25	45	1600
200	0.17	2.05	75	6700
300	0.38	2.87	91	3500
400	0.67	4.7	115	4400
500	1.03	16.56	174	11500
600	1.02	13.8	143	8900
700	1.65	70.01	201	15800
800	3.61	52.51	366	20100
900	2.47	24.09	268	6900
1000	3.19	38	330	8600
1100	3.09	100.19	202	18000
1200	6.14	118.36	364	18200
1300	4.62	140.19	293	19200
1400	3.65	139.25	185	17500
1500	7.33	252.05	331	28800
1600	5.84	310.83	271	32200
1700	6	417.61	299	36300
1800	11.44	444.29	454	37100
1900	8.95	618.1	324	42300
2000	12.78	656.72	547	47400

Cuadro 4.1: Tiempo (en segundos) y nº de iteraciones para FISTA y DMC en función del número de columnas (p) de X

En la Figura 4.2 se ha obtenido una gráfica, en el caso en que $p = 500$, del número de iteraciones realizado por cada método frente al valor de la función objetivo en el iterante correspondiente a cada iteración. Así se aprecia claramente que a partir de las 40 iteraciones FISTA alcanza aproximantes cuyo valor es realmente cercano al verdadero valor del mínimo (este es 12.85), mientras que DMC desciende en pequeños pasos y en las 160 primeras iteraciones recogidas en la gráfica alcanzando finalmente un iterante cuya imagen a través de la función objetivo queda aún

realmente alejada del verdadero valor del mínimo. Nótese que se está contando como iteración cada una de las actualizaciones que realiza DMC por coordenadas, luego, dado que $p = 500$, para las 160 primeras iteraciones ni siquiera ha llegado a actualizar por completo el estimador β.

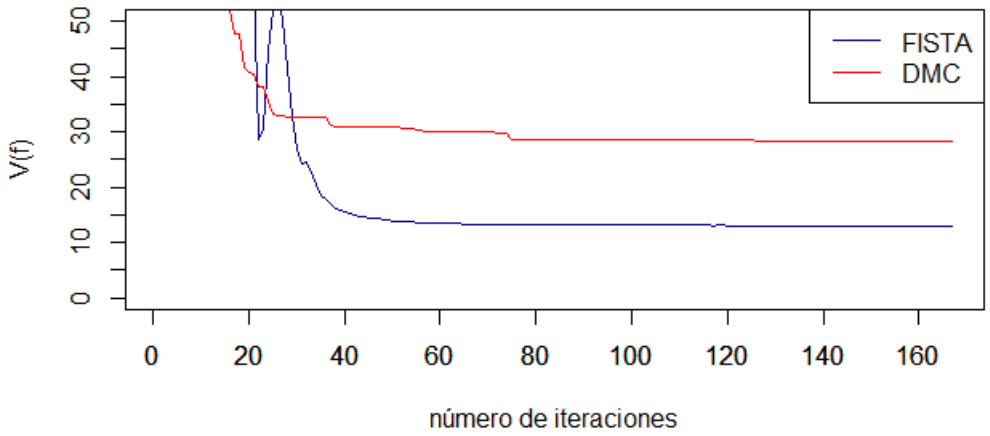


Figura 4.2: Valor de la función objetivo de LASSO en los distintos iterantes de FISTA y DMC

En cualquier caso, la comparación más fiable se da en términos del tiempo de demora de cada uno de los algoritmos a la hora de alcanzar un ϵ-aproximante. Es obvio que si se cuentan las iteraciones realizadas por DMC coordenada a coordenada el número obtenido va a ser mucho mayor que el número de iteraciones que realiza FISTA, pero muy probablemente el verdadero costo computacional que conlleva evaluar las iterciones de DMC sea considerablemente menor. Aún así cuando la dimensión se incrementa, es decir, cuando se parte de matrices de datos con valores de p elevados, no queda lugar a dudas de que el funcionamiento de FISTA es mas eficiente, consiguiendo obtener un aproximante suficientemente cercano al mínimo de la función objetivo de LASSO en un tiempo asumible a nivel práctico.

Conclusión

El éxito a la hora de resolver problemas derivados del aprendizaje automático está determinado por la capacidad de dar con una solución en el menor tiempo posible; y con esto se entiende la capacidad de ofrecer una respuesta a un problema práctico con unos datos concretos sin la menor demora, no basta con desarrollar algoritmos válidos teóricamente pero imposibles de implementar. En la sección 2.4. se ha expuesto, por ejemplo, el método de descenso geométrico, el cual alcanza una cota sobre la complejidad óptima para un método black-box con las hipótesis que se toman sobre la función objetivo. Incluso se podría decir que es un algoritmo de una subjetiva belleza dada su originalidad. Sin embargo, las eminentes dificultades prácticas que entraña, hacen de él un procedimiento imposible de aplicar a un caso real. En resumen, este éxito dependerá de la escalabilidad y de la velocidad de los algoritmos empleados. Es por ello que se ha insistido a lo largo del presente trabajo en dos ideas fundamentales: la convexidad y la (casi) independencia respecto a la dimensión.

A día de hoy casi cualquier problema de optimización resoluble que se plantee va a ser un problema de optimización convexa, por lo tanto se hace necesario manejar desde los conceptos de convexidad básicos hasta las propiedades y resultados que de ellos se deducen. Por otra parte, algoritmos con un orden de convergencia demasiado sensible a la dimension quedan obsoletos dada la masificación de datos que se manejan diariamente. Estos argumentos son los que han dado pie a estudiar, desde un principio, los métodos derivados del descenso por gradiente, pues, a excepción del ya mencionado descenso geométrico, se trata de métodos de solución numérica sencillos de implementar e independientes de la dimensión en su orden de convergencia. Se tienen además una serie de resultados que ofrecen cotas sobre este orden bajo ciertas hipótesis sobre la función objetivo de tal manera que se garantiza una velocidad elevada a la hora de dar con una solución suficientemente "buena" (próxima al mínimo en este caso). No obstante, las hipótesis que se toman conducen a funciones un tanto artificiosas o ideales que no se van a tener en la realidad y por ello surge la necesidad de adaptarlos a problemas más concretos.

Tal es el caso de LASSO, que se ha tomado como ejemplo. Proviene de una penalización sobre la optimización por mínimos cuadrados y es un problema habitual en el aprendizaje automático. De manera tradicional se ha resuelto a partir de un método conocido como "descenso por coordenadas", previsiblemente lento al actualizar coordenada a coordenada el estimador. La función que se propone minimizar en LASSO carece de la propiedad de suavidad suave, quizá la que permitía acelerar los métodos de descenso por gradiente en mayor grado, como se vio en el ejemplo 2.11. Aun así esta función objetivo puede expresarse como suma de dos funciones: la primera de ellas convexa y suave, la segunda tan solo convexa. Esto es suficiente para adaptar el descenso por gradiente y el método de Nesterov (derivado del anterior) para dar origen a los

algoritmos ISTA y FISTA respectivamente, que resuelven el problema LASSO a una velocidad deseable, de hecho de orden cuadrática en FISTA, e independiente de la dimensión del espacio ambiente del problema. Se lo deben al aprovechamiento de otra propiedad del segundo sumando en que se descompone la función objetivo de LASSO, y es que esta es completamente separable (ya se ha visto lo que esto quiere decir). No solo se ha logrado dar con un algoritmo presumiblemente más rápido que el convencional descenso por coordenadas a nivel teórico, sino que en vistas a los reultados obtenidos a raíz de una simulación con datos concretos y aleatorios para un número suficiente de casos se ha comprobado que con el aumento de la dimensión (cuantos más datos se proporcionan) FISTA resuelve el problema con un rapidez mas o menos estable y asumible, mientras que el descenso por coordenadas sufre una ralentización considerable. Ya para una matriz con 2000 columnas se está hablando de varios minutos (se está hablando siempre de una implementación en R donde la multiplicación de matrices es mucho más lenta que en C) para dar con una solución cercana al mínimo a una distancia mínima exigida y esto no es nada en comparación con el número de columnas del que dispondría una matriz de datos en un problema real. Este no es, ni mucho menos el único ejemplo que se puede trata (veáse en [4] la seccion 5.2 donde se adaptan los métodos de descenso por gradiente a la resolución de un problema de optimización convexa con una función $f(x) = min_{1 \leq i \leq s} f_i(x)$, donde las f_i son todas convexas y suaves, pero la función objetivo f, que será también convexa, pierde la propiedad de suavidad) pero se espera que sirva para consolidar la idea de que la optimización convexa es una de las claves para alcanzar algoritmos verdaderamente escalables y rápidos aplicables al "machine learning" y por ende un campo en el que seguir investigando dada la importancia que se está concediendo a esta rama de la inteligencia artificial.

Bibliografía

[1] Avriel, M.(2003), *Nonlinear Programming: Analysis and Methods*. Dover Publishing.

[2] Beck, A. y Teboulle, M. (2009), *A Fast Iterative Shrinkage-Theresholding Algorithm for Linear Inverse Problems*. SIAM Journal on Imaging Sciences, 2(1):183–202.

[3] Boyd, S. y Vandenberghe, L. (2004), *Convex Optimization*. Cambridge University Press.

[4] Bubeck, S.(2015), *Convex Optimization: Algorithms and Complexity*. Theory Group, Microsoft Research.

[5] Hastie, T., Tibshirani, R. y Wainwright, M.(2015), *Statistical learning with sparsity. The Lasso and Generalizations*. Stanford University.

[6] Nesterov, Y. (2004), *Introductory Lectures on Convex Optimization. A Basic Course*. Springer

[7] Parkish, N. y Boyd, S.(2013), *Proximal algorithms*. Foundations and Trends in Optimization, vol 1, pp: 123-231.

[8] Ritoré Cortés, M.M^{a}.(2004), *Geometría de convexos*. Notas de la Universidad de Granada. Material no publicado. Recuperado de `https://www.ugr.es/~jperez/docencia/GeomConvexos/geometria_convexos-v2.pdf`

[9] Rockafellar, R.T.(1972), *Convex analysis*. Second edition. Princeton University Press.

[10] Vaphik, V. (2000), *The Nature of Statistical Learning Theory*. Springer.