USC
UNIVERSIDADE
DE SANTIAGO
DE COMPOSTELA

FACULTADE DE MATEMÁTICAS

Traballo Fin de Grao

Optimización en modelos de regresión

Fabián Miranda Mouzo

2020/2021

UNIVERSIDADE DE SANTIAGO DE COMPOSTELA

GRAO DE MATEMÁTICAS

Traballo Fin de Grao

Optimización en modelos de regresión

Fabián Miranda Mouzo

Febreiro, 2021

UNIVERSIDADE DE SANTIAGO DE COMPOSTELA

Traballo proposto

Área de Coñecemento: Estatística e Investigación Operativa
Título: Optimización en modelos de regresión
Breve descrición do contido
Os modelos de regresión buscan determinar a relación dunha variable dependente con respecto a outras variables explicativas ou independentes. Unha vez establecido un modelo para dita relación, é necesaria a estimación dos seus parámetros. Esta pode levarse acabo por diferentes procedementos en función do criterio de axuste, sendo o criterio de mínimos cadrados o máis habitual. No caso do modelo de regresión linear, este criterio da lugar a un problema de optimización convexa para o cal se pode obter a forma explícita da súa solución. En modelos máis complexos, como a regresión con regularización ou os modelos de regresión non linear, os problemas de optimización resultantes non teñen unha solución explícita. O obxectivo deste traballo é que o alumno faga unha revisión dos métodos de optimización empregados neste contexto.
Recomendacións
Outras observacións

Índice xeral

Resumo

O obxectivo deste traballo é a introdución de diversos métodos de optimización para o cálculo de parámetros en distintos modelos de regresión, centrándonos especialmente nos que teñen a propiedade de ser convexos. Para isto, comezamos expoñendo os modelos de regresión lineal, tanto simple como múltiple, plantexándoos como problemas de optimización e dando unha solución explícita do cálculo dos seus parámetros obtida por mínimos cadrados. No seguinte capítulo, falamos brevemente da regresión Ridge e LASSO, facendo unha comparación entre ambos modelos. No terceiro capítulo, introducimos o concepto de problema de optimización convexa e por que esta propiedade é importante. Ademáis, explicamos os distintos métodos que utilizaremos para resolver ditos problemas. Por último, aplicaremos estes métodos a exemplos prácticos para ver como se comportan.

Abstract

The purpose of this work is to introduce various optimization techniques to calculate parameters in different regression models, focusing especially on those which have the property of being convex. To do this, we start exposing the linear regression models, both simple and multiple, as optimization problems and giving an explicit expression of their parameters obtained using the least squares criterion. In the next chapter, we discuss briefly Ridge and LASSO regression, making a comparison between both models. In chapter three, we define what a convex optimization problem is and the relevance of the convexity. In addition, we explain different methods to solve these problems. Finally, we use these methods in some examples to see their perfomance.

Introdución

A búsqueda de solucións para problemas que xorden na práctica cautivou a atención dos matemáticos, ocupando un lugar importante no estudo das matemáticas. Atopar unha solución exacta dun problema pode chegar a ser imposible. Cando isto ocorre, perseguiremos dar respostas útiles que involucren a búsqueda de resultados aproximados suficientemente bos. Ista é a razón de ser dos métodos numéricos, tendo moitos deles unha larga historia.

Os métodos numéricos son técnicas matemáticas que se empregan para resolver problemas que non se poden resolver, ou que son difíciles de resolver, analíticamente. Unha solución numérica é un valor numérico aproximado da solución. Aínda que as solucións numéricas son aproximadas, con frecuencia están moi preto da solución exacta. En moitos destes métodos os cálculos execútanse de maneira iterativa ata alcanzar unha exactitude desexada.

Neste traballo centrarémonos en expoñer varios métodos iterativos para resolver certos problemas de optimización convexa tendo como base os libros [2] e [1]. Ditos problemas serán modelos de regresión para os cales facemos uso de [5] e [4] para a súa presentación. O interese de estudar estes métodos iterativos é o de mostrar distintas alternativas á hora de estimar parámetros en modelos de regresión xa que pode chegar a ser, nalgunha ocasión, moi complicado obter estas estimación por procesos máis clásicos como o criterio por mínimos cadrados do cal falaremos neste traballo.

A continuación pasamos a describir brevemente a organización do traballo. Comezamos recordando os modelos de regresión lineais e como estes se poden plantexar como problemas de optimización convexa. Veremos como podemos obter as estimacións dos parámetros empregando o criterio por mínimos cadrados e, así, chegar a unhas ecuacións que permiten calcular ditos parámetros de maneira directa. Tamén presentaremos unha alternativa a este criterio, o método de máxima verosimilitude.

No segundo capítulo introduciremos os modelos de regresión regularizada, en especial a regresión Ridge e LASSO. Veremos como estes engaden un termo de penalización aos modelos lineais para, así, minimizar a influencia dos parámetros menos importantes ademais de plantexalos como problemas de optimización convexa, ao igual que nos modelos lineais. Ao final deste capítulo, engadiremos unha sección onde formularemos de maneira diferente estes modelos para poder comparalos ademais de xustificar o porqué dos termos de penalización que empregan.

O seguinte capítulo estará dedicado á explicación do que podemos entender por un problema de optimización convexa e a métodos para resolvelos. Nunha primeira parte do capítulo, definiremos tales problemas así como o que entedemos por función convexa. De seguido, expoñeremos unha serie de definicións e resultados para xustificar a importancia da convexidade nese tipo de problemas. Continuaremos cuns exemplos de funcións convexas, demostrando a convexidade das mesmas. Tamén engadiremos un par de exemplos de problemas de optimización convexa coincidindo estes cos modelos de regresión expostos nos dous capítulos anteriores. Xa na segunda parte do capítulo dedicaremos unha sección aos métodos iterativos que empregaremos para resolver ditos problemas.

Por último, dedicaremos un capítulo a implementar e comparar os distintos métodos expostos. Nunha primeira sección, simularemos un modelo de regresión lineal simple que nos servirá para validar e comprobar o comportamento destes métodos. Será nunha segunda sección onde aplicaremos os métodos a un modelo de regresión lineal múltiple con datos reais e remataremos resolvendo un modelo LASSO con distinto número de variables.

Capítulo 1

Modelos de regresión lineal

Neste capítulo explicaremos en que consiste o modelo de regresión lineal, tanto o simple coma o múltiple, así como o enfoque de mínimos cadrados e outros problemas de optimización que veremos máis adiante (véxase [5]).

1.1. Regresión lineal simple

A regresión lineal simple é un método estadístico que permite explicar a relación lineal que existe entre dúas variables. A variable resposta identifícase por y e a variable explicativa ou independente como x. O modelo de regresión lineal simple descríbese dacordo a ecuación:

$$y = \beta_0 + \beta_1 x + \varepsilon \tag{1.1}$$

Na ecuación anterior, β_0 e β_1 son dous parámetros descoñecidos que representan o intercepto e a pendente do modelo lineal respectivamente, mentres que ε é o erro.

Na gran maioría dos casos, os valores β_0 e β_1 son descoñecidos. Por tanto, antes de poder empregar (1.1) para facer predicciónS, necesitamos a mostra $(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$, a cal representa os n pares de observacións do noso modelo para obter, así, as súas estimacións $\hat{\beta}_0$ e $\hat{\beta}_1$. Estas estimacións coñecense como coeficientes de regresión ou *least square coefficient estimates*, xa que toman aqueles valores que minimizan a suma dos residuos ao cadrado, dando lugar á recta que pasa máis cerca de tódo-los puntos. Definindo a suma de residuos ao cadrado (RSS) como

$$RSS = \varepsilon_1^2 + \varepsilon_2^2 + ... + \varepsilon_n^2$$

ou equivalentemente

$$RSS = (y_1 - \beta_0 - \beta_1 x_1)^2 + (y_2 - \beta_0 - \beta_1 x_2)^2 + ... + (y_n - \beta_0 - \beta_1 x_n)^2$$

podemos plantexar o axuste destes coeficientes como un problema de optimización convexa, o cal explicaremos en que consiste no capítulo 3, da seguinte maneira:

$$\min_{\beta_0,\beta_1} \qquad RSS = \varepsilon_1^2 + \varepsilon_2^2 + ... + \varepsilon_n^2 \tag{1.2}$$

onde RSS sería a función obxectivo. Para calcular os valores de β_0 e β_1 que minimizan dita función, podemos proceder directamente sen empregar ningún método iterativo. Na función obxectivo, podemos substituír β_0 por $y - \beta_1 x$ e derivar respecto de β_1. Feito isto, igualamos a 0 e despexamos β_1. Por último, multiplicamos por $\frac{1}{n}$ numerador e denominador para obter a seguinte expresión:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

e podemos expresar a estimación de β_0 en función da de β_1

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x},$$

onde $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$ e $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ son as medias muestrais.

Unha recta de regresión pode empregarse para distintos propósitos. No caso de querer medir a relación lineal entre dúas variables, a recta de regresión indicao de forma directa (xa que calcula a correlación). Sen embargo, en caso de querer predecir o valor dunha variable en función doutra, non só se necesita calcular a recta, senón que además hai que asegurar que o modelo sexa bo.

1.2. Regresión lineal múltiple

A regresión lineal simple é un enfoque útil para predicir unha resposta en base a unha única variable explicativa. Sen embargo, na práctica soemos ter máis de unha destas variables. A regresión lineal múltiple permite xerar un modelo no que a variable resposta, y, determínase a partir dun conxunto de variables independentes chamadas explicativas $(x_1, x_2, x_3, ...)$. Podemos facer isto dando a cada variable explicativa un coeficiente de pendente separado nun só modelo. En xeral, supoñamos que temos p variables explicativas distintas e n obsevacións. Entón o modelo de regresión lineal múltiple toma a forma

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_p x_{ip} + \varepsilon_i \qquad i = 1, ..., n \tag{1.3}$$

onde β_0 é a ordenada na orixe, o valor da variable dependente y cando todas as variables explicativas son cero; β_i é o efecto promedio que ten o incremento dunha unidade da variable explicativa x_i sobre a variable dependente y cando o resto de variables permanecen constantes; ε é o erro.

Como no caso da regresión lineal simple, os coeficientes $\beta_0, \beta_1, ..., \beta_p$ son descoñecidos e deben de ser estimados. Dados os estimadores $\hat{\beta}_0, \hat{\beta}_1, ..., \hat{\beta}_p$ podemos facer predicións usando a fórmula

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i2} + ... + \hat{\beta}_p x_{ip} \qquad i = 1, ..., n$$

Os parámetros son estimados co mesmo criterio de mínimos cadrados visto para o modelo simple. Escollemos $\beta_0, \beta_1, ..., \beta_p$ que minimicen a suma dos residuos ao cadrado. Dito doutra forma, podemos plantexar esta situación coma un problema de optimización

$$\min_{\beta_0, \beta_1, ..., \beta_p} \qquad RSS = \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_{i1} - \beta_2 x_{i2} - ... - \beta_p x_{ip})^2 \tag{1.4}$$

Para obter a expresión das estimacións dos parámetros, podemos reescribir (1.3) de forma matricial:

$$Y = X\beta + \varepsilon$$

onde

$$Y = \begin{pmatrix} y_1 \\ y_2 \\ ... \\ y_n \end{pmatrix}; \quad \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix}; \quad X = \begin{pmatrix} 1 & x_{11} & x_{12} & ... & x_{1p} \\ 1 & x_{21} & x_{22} & ... & x_{2p} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} & ... & x_{np} \end{pmatrix}; \quad \varepsilon = \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ ... \\ \varepsilon_n \end{pmatrix}$$

Así, podemos plantexar (1.4) como

$$\min_{\beta} \qquad RSS = (Y - X\beta)^T (Y - X\beta)$$

Agora ben, diferenciando respecto de β obtemos

$$\frac{\partial RSS}{\partial \beta} = -2X^T (Y - X\beta)$$

e igualando a cero temos

$$X^T (Y - X\beta) = 0$$

a cal ten solución única

$$\hat{\beta} = (X^T X)^{-1} X^T Y$$

É habitual considerar o modelo de regresión lineal normal onde os erros se comportan como variables aleatorias independentes igualmente distribuidas cunha distribución normal de media 0 e varianza σ^2. Neste caso considerando Y como un vector aleatorio e que $Y = X\beta + \varepsilon$ con $\varepsilon \sim N_n(0_n, \sigma^2 I_n)$, onde I_n é a matriz identidade de orde n, temos que súa media e matriz de covarianza obtéñense a partir das de ε e son

$$E(Y) = X\beta, \qquad Var(Y) = \sigma^2 I_n$$

1.2.1. Máxima verosimilitude

Ata agora, estivemos vendo como calcular os estimadores, $\hat{\beta}$, utilizando mínimos cadrados. Sen embargo, existen outras formas de obter ditos estimadores.

O método de máxima verosimilitude é un procedemento de estimación que necesita o coñecemento da distribución de probabilidade poboacional, a cal se refire a todos os posibles resultados que poida ter unha variable aleatoria, é dicir, describe o comportamento de dita variable dentro dun intervalo de valores ou de posibles resultados. Para levar a cabo este procedemento é necesario obter a función de verosimilitude.

Sexa a nosa variable aleatoria, ρ, con probabilidade $P(\rho = a, \delta)$ no caso de ser discreta ou función de densidade $f(x; \delta)$, se é continua. En ambos casos, δ é un parámetro descoñecido.

Se se escolle unha mostra aleatoria de tamaño n, entón a función de verosimilitude represéntase por $L(X; \delta)$ e calcúlase do seguinte modo:

$$L(X; \delta) = f(x_1; \delta)...f(x_n; \delta)$$

O criterio que se adopta para obter o valor estimado do parámetro descoñecido δ, é escoller aquel estimador que maximiza a función de verosimilitude.

$$L(X; \hat{\delta}) = \max_{\delta} \qquad L(X; \delta)$$

Sen embargo, como a función de verosimilitude é multiplicativa, os cálculos pódense complicar, por esta razón, obténse o valor do parámetro que maximiza o logaritmo da función de verosimilitude. Para obter dito estimador, hai que seguir os seguintes pasos:

- Derivar o logaritmo da función de verosimilitude respecto do parámetro descoñecido.
- Igualar a cero a primeira derivada e obter o valor do estimador en función dos elementos da mostra.

- Obter a segunda derivada do logaritmo da función de verosimilitude respecto do parámetro. Para que sexa un máximo esta segunda derivada ten que ser negativa.

Se temos máis dun parámetro para estimar, o procedemento é similar ao exposto anteriormente, pero será necesario derivar o logaritmo da función de verosimilitude respecto de cada un deles.

Exemplo 1.1. Na regresión lineal múltiple, supoñendo a normalidade e independencia dos erros tense

$$\varepsilon \sim N_n(0_n, \sigma^2 I_n) \Rightarrow Y \sim N_n(X\beta, \sigma^2 I_n)$$

onde σ denota a varianza e I_n a matriz identidade de orde n.

No caso xeral $Y \sim N_p(\vec{\mu}, \Sigma)$, a función de densidade de Y ven dada por

$$f(\vec{y}) = (2\pi)^{-p/2}|\Sigma|^{-1/2} exp\{-\frac{1}{2}(\vec{y}-\vec{\mu})^T\Sigma^{-1}(\vec{y}-\vec{\mu})\}$$

Entón, a función de verosimilitude de Y no modelo de regresión lineal múltiple é

$$L(\beta, \sigma^2|Y, X) = (2\pi\sigma^2)^{-n/2} exp\{-\frac{1}{2\sigma^2}(Y-X\beta)^T(Y-X\beta)\}$$

$$\ell(\beta, \sigma^2|Y, X) = logL(\cdot) = -\frac{n}{2}log(2\pi) - \frac{n}{2}log\sigma^2 - \frac{1}{2\sigma^2}(Y-X\beta)^T(Y-X\beta)$$

Temos que maximizar $\ell(\beta, \sigma^2|\vec{y}, X)$ con respecto a β e σ^2. Primeiro derivamos para encontrar os puntos críticos

$$\frac{\partial}{\partial\beta}\ell = -\frac{1}{\sigma^2}(X^TX\beta - X^TY)$$
$$\frac{\partial}{\partial\sigma^2}\ell = -\frac{n}{2\sigma^2} + \frac{1}{2\sigma^4}(Y-X\beta)^T(Y-X\beta)$$

Entón

$$\frac{\partial}{\partial\beta}\ell = 0 \quad \Rightarrow \quad X^TX\beta = X^TY$$
$$\frac{\partial}{\partial\sigma^2}\ell = 0 \quad \Rightarrow \quad \sigma^2 = \frac{1}{n}(Y-X\beta)^T(Y-X\beta)$$

A solución para σ^2 depende de β e a solución para β é a mesma que por mínimos cadrados. Entón, se X é de rango completo, os EMV son

$$\hat{\beta} = (X^TX)^{-1}Y \qquad \text{e} \qquad \hat{\sigma^2} = \frac{1}{n}(Y-X\hat{\beta})^T(Y-X\hat{\beta})$$

Capítulo 2

Regresión lineal regularizada

Como xa vimos, os modelos de regresión lineal múltiple soen axustarse mediante regresión por mínimos cadrados, aunque existen outros procedementos como a estimación por máxima verosimilitude como vimos ao final do capítulo anterior. Debemos ter en conta dous aspectos á hora de empregar estes procedementos de axuste, o que nos levará a considerar outras opcións como veremos neste capítulo. Ditos aspectos son:

- **Erro de predicción**: se a relación existente entre os predictores e a variable resposta é lineal, as estimacións obtidas polo método de mínimos cadrados serán insesgadas. Se ademáis, o número de observacións coa que se axusta o modelo é moito maior que o número de predictores ($n >> p$), as estimacións obtidas por mínimos cadrados tenden a ter pouca varianza. Cumpríndose ambas condicións, un modelo lineal múltiple xerado mediante mínimos cadrados terá unha boa capacidade de predición. A medida que o número de observacións deixa de ser moito maior que o número de variables explicativas, a varianza aumenta, ata chegar ao punto no que si $p > n$, a varianza é infinita e, polo tanto, o método dos mínimos cadrados non debe utilizarse.

- **Interpretabilidade do modelo**: cantas máis variables explicativas se introduzan nun modelo, máis complexa se fai a súa interpretación. Por esta razón, compre limitar o modelo a aquelas variables explicativas que teñan unha influencia importante sobre a variable resposta, excluindo aqueles que son irrelevantes e que engaden complexidade innecesaria. O método de regresión por mínimos cadrados dificilmente obterá estimacións de coeficientes que sexan exactamente 0, polo que tenderá a considerar útiles todos os predictores

Así pois, cando se dispón de poucas observacións ou moitas variables explicativas, compre empregar un método de regresión que permita excluir variables irrelevantes ou, mellor dito, identificar os máis relevantes. A este proceso coñéceselle como *selección de variables*

e os 3 métodos máis empregados son: *selección de subconxunto, método de penalización ou regularización* e *reducción da dimensión*. Neste capítulo centrarémonos nos métodos de penalización ou regularización, os cales consisten en axustar o modelo incluindo todos os predictores pero empregando un método que obrigue a que as estimacións dos parámetros de regresión tendan a cero, é dicir, que tenda a minimizar a influencia das variables explicativas menos importantes. Dous dos métodos máis empregados son:

- **Regresión Ridge**: aproxima a cero os parámetros das variables explicativas sen chegar a excluir ningún.
- **Lasso**: aproxima a cero os parámetros, chegando a excluir variables.

Ambos métodos están indicados para situacións nas que hai un maior número de variables explicativas que de observacións. Vexámolos máis en profundidade, tendo como base [5] e [4].

2.1. Regresión Ridge

A regresión Ridge é moi similar á de mínimos cadrados, excepto que os parámetros se estiman minimizando unha función lixeiramente distinta. En particular, as estimacións dos parámetros da regresión de Ridge, $\hat{\beta}^R$, son valores que resolven o seguinte problema de optimización

$$\min_{\beta} \quad \sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2 + \lambda\sum_{j=1}^{p}\beta_j^2 = \min_{\beta} \quad RSS + \lambda\sum_{j=1}^{p}\beta_j^2, \tag{2.1}$$

onde $\lambda \geq 0$ é un parámetro de axuste, que se determinará por separado. No seguinte capítulo explicaremos que se trata de un problema de optimización convexo, así como diversas alternativas para resolvelo.

Ao igual que cos mínimos cadrados, a regresión Ridge busca estimacións de coeficientes que se axusten ben aos datos, facendo o RSS pequeno. Sen embargo, o segundo termo, $\lambda\sum_j \beta_j^2$, chamado *shrinkage penalty* ou termo de penalización, é pequeno cando $\beta_1, ..., \beta_p$ son próximos a cero. O parámetro de axuste λ serve para controlar o impacto relativo de estes dous termos na estimación dos coeficientes de regresión. Cando $\lambda = 0$, o termo de penalización non ten efecto, e as estimacións da regresión Ridge coinciden coas dos mínimos cadrados. Sen embargo, se $\lambda \to \infty$, o impacto do termo de penalización crece, e as estimacións dos coeficientes da regresión de Ridge acercáranse a cero. A diferencia dos mínimos cadrados, que xera só un conxunto de estimacións, a regresión Ridge producirá

un conxunto diferente de estimadores, $\hat{\beta}^R$, para cada valor de λ. Escoller un bo valor para λ é crucial, para o cal existen diversos métodos como o de validación cruzada (para máis detalle véxase [5], capítulo 5).

Nótese que no problema (2.1), o termo de penalización aplícase aos $\beta_1, ..., \beta_p$ pero non ao intercepto, β_0. Queremos reducir a asociación estimada de cada variable coa resposta; sen embargo, non queremos reducir o intercepto, que é simplemente unha medida do valor medio da resposta cando $x_{i1} = x_{i2} = ... = x_{ip} = 0$. Se asumimos que as variables se centraron para ter media cero antes de facer a regresión Ridge, entón a estimación do intercepto tomará a forma $\hat{\beta}_0 = \bar{y} = \sum_{i=1}^{n} y_i/n$.

Agora ben, podemos plantexar (2.1) de forma matricial como segue

$$\min_{\beta} \quad \{\| Y - X\beta \|_2^2 + \lambda \| \beta \|_2^2\} \tag{2.2}$$

onde $\| \beta \|_2 = (\sum |\beta_j^2|)^{1/2}$ é a norma euclídea. Equivalentemente, podemos plantexar o problema (2.2) da seguinte maneira

$$\min_{\beta} \qquad (Y - X\beta)^T (Y - X\beta) + \lambda \beta^T \beta$$

de onde se obteñen os estimadores Ridge derivando respecto β e igualando a cero, tomando a seguinte expresión

$$\hat{\beta}^R = (X^T X - \lambda I)^{-1} X^T Y$$

En moitos casos quérese predicir o comportamento da variable resposta e dado que se descoñece como se comporta, escóllese un número de variables explicativas considerablemente extenso cando, sen embargo, non se conta con demasiada información previa, é dicir, é frecuente que $p >> n$. Se isto sucede, tanto a regresión lineal como a regresión Ridge, a pesar de chegar a unha solución única, non alcanzan unha estimación suficientemente fiable. Ademáis, dado o elevado número de variables explicativas sería interesante obter un estimador que reflexe claramente cales de estas variables son influintes á hora de predicir a variable resposta, é dicir, resultaría convinte que o estimador tivera gran parte das súas compoñentes nulas, pero esto non acontece na regresión Ridge. Sí o fará, sen embargo, se en 2.2 se toma a norma l_1 á hora de limitar o tamaño de β. Isto da lugar a un novo modelo coñecido como LASSO.

2.2. LASSO

Como acabamos de ver, a regresión Ridge ten unha gran desventaxa xa que inclue todas as variables explicativas no modelo. O termo de penalización $\lambda \sum \beta_j^2$ no problema (2.1)

reducirá todos os coeficientes hacia cero, pero ningún deles chegará a cero. Esto non ten por que ser un problema para a estimación dos parámetros, pero pode complicar a interpretación do modelo cando o número de variables explicativas é moi grande.

O problema de LASSO trata de penalizar, ao igual que a regresión Ridge, un tamaño grande de β no problema de mínimos cadrados para a regresión lineal. A diferencia co método anterior é que interpreta o tamaño de β en función da norma l_1. O que en principio parece unha desventaxa, pois esta restricción ven dada a partir dunha función non diferenciable nos puntos de $\mathbb{R}^p$ con algunha compoñente nula, proporciona por outro lado unha maior interpretabilidade do problema ao proporcionar para o caso $p >> n$ solucións con un alto número de variables explicativas con valor nulo, o que se coñece como *selección de variables*. Os estimadores $\hat{\beta}_{LASSO}$ resolven o seguinte problema de optimización que, como veremos no seguinte capítulo, é convexo

$$\min_{\beta} \quad \sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2 + \lambda\sum_{j=1}^{p}|\beta_j| = \min_{\beta} \quad RSS + \lambda\sum_{j=1}^{p}|\beta_j| \tag{2.3}$$

Comparando (2.1) e (2.3), vemos que a única diferencia entre eles é que o termo de penalización β_j^2 na regresión Ridge se sustitue por $|\beta_j|$ en LASSO. Noutras palabras, o método LASSO usa a norma l_1 en lugar da norma l_2. A norma l_1 dos coeficientes do vector β ven dada por $\| \beta \|_1 = \sum|\beta_j|$. Dito esto, outra forma de plantexar o método LASSO é de forma matricial da seguinte maneira

$$\min_{\beta} \qquad \{\| Y - X\beta \|_2^2 + \lambda \| \beta \|_1\} \tag{2.4}$$

Como dixemos ao comezo da sección, a norma l_1 non é diferenciable. Esto supón que non podemos calcular os estimadores, $\hat{\beta}_{LASSO}$, utilizando mínimos cadrados como estivemos facendo ata agora, pois este método require derivar respecto de β. Para solucionar isto, existen certos métodos para obter o mínimo dunha función sen necesidade de que esta sexa diferenciable, como é o caso do método de Hooke e Jeeves. Verémolos máis en profundidade no capítulo 3.

2.3. Outra formulación para a regresión Ridge e LASSO

Nesta sección veremos outro plantexameno para a regresión Ridge e LASSO, a cal nos axudará a ver gráficamente as diferenzas que existen entre os dous procesos. Tamén veremos a posibilidade de utilizar outras normas e o motivo polo cal se empregan a norma l_1 e l_2.

Dito isto, as estimacións dos parámetros no método LASSO resolven o problema

$$\min_{\beta} \quad \{\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2\} \tag{2.5}$$

$$\text{s. a} \quad \sum_{j=1}^{p}|\beta_j| \leq s$$

mentres que as estimacións obtidas na regresión Ridge resolven

$$\min_{\beta} \quad \{\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2\} \tag{2.6}$$

$$\text{s. a} \quad \sum_{j=1}^{p}\beta_j^2 \leq s$$

Noutras palabras, para cada valor de λ, hai algún s para o cal as ecuacións (2.3) e (2.5) dannos os mesmos estimadores. De maneira análoga, para cada valor λ hai un correspondente valor s tal que as ecuacións (2.1) e (2.6) nos calculan os mesmos estimadores. Cando $p = 2$, entón (2.5) indica que as estimacións dadas polo método LASSO teñen o RSS máis pequeno de todos os puntos que se encontran dentro do diamante definido por $|\beta_1| + |\beta_2| \leq s$. Do mesmo modo, as estimacións proporcionadas pola regresión Ridge teñen o RSS máis pequeno de todos os puntos que se encontran dentro do círculo dado por $\beta_1^2 + \beta_2^2 \leq s$.

Podemos pensar (2.5) da seguinte maneira. Cando usamos o método LASSO, estamos tratando de encontrar o conxunto de estimacións que fan o RSS máis pequeno, suxeito á limitación de que hai unha marxe para o tamaño de $\sum_{j=1}^{p}|\beta_j|$. Se s é extremadamente grande, entón esta marxe non é moi restrictiva e, polo tanto, as estimacións dos parámetros poden ser grandes. De feito, se s é o suficientemente grande para que a solución por mínimos cadrados caiga dentro desta marxe, entón (2.5) ten a mesma solución que por mínimos cadrados. Do mesmo modo, (2.6) indica que cando realizamos a regresión Ridge, buscamos un conxunto de estimadores tal que o RSS sexa o máis pequeno posible, suxeito a que $\sum_{j=1}^{p}\beta_j^2$ non exceda a marxe s.

Se se escolle s suficientemente pequeno, os valores β_i producidos son na súa maioria nulos, sendo solo uns poucos distintos de cero. Este fenómeno coñécese como *dispersión* e

permite simplificar a interpretación dos resultados. Podemos plantexarnos a norma l_q para $q \geq 0$, o que nos leva o seguinte problema

$$\min_{\beta} \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij})^2 + \lambda \sum_{j=1}^{p} |\beta_j|^q \tag{2.7}$$

Sen embargo, a elección de $q = 1$ é a xusta se se busca un problema disperso e á vez convexo. Se $q > 1$ pérdese a propiedade de dispersión e, se $q < 1$, a dispersión mantense pero o problema deixa de ser convexo. As ventaxas que ofrece a convexidade xustificaranse no capítulo 3. Os contornos para un valor constante de $\sum_{j=1}^{p} |\beta_j|^q$ móstranse na figura (2.1) para o caso de dúas entradas. Algunhas destas normas teñen nome propio, como pode ser a norma euclídea cando $q = 2$, norma valor absoluto cando $q = 1$, tamén coñecida como norma Manhattan ou a norma infinito cando $q = \infty$.

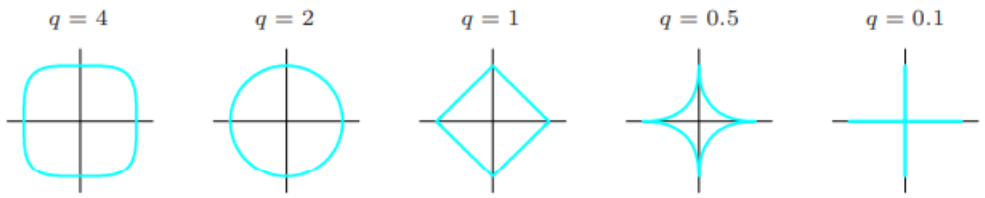

Figura 2.1: Contornos para un valor constante de $\sum_{j=1}^{p} |\beta_j|^q$ para distintos valores de q. Fonte da imaxe ([4]).

Na figura (2.2) representáronse en dúas variables ($p = 2$) os erros cadráticos de LASSO e da regresión Ridge dados por $(y_i - x_1\beta)^2 + (y_2 - x_2\beta)^2 = K$, con K constante, e as rexións $\| \beta \|_1^2$ e $\| \beta \|_2^2$ respectivas de cada un dos problemas. Apréciase como en LASSO, os contornos das cónicas intersecan primeiro con algún dos vértices do diamante, onde unha das variables explicativas é nula; mentres que na regresión Ridge, aunque se quedan cerca de facer o propio, finalmente intersecan nun punto onde ambas variables son non nulas. Se o número de variables é elevado, esto pódese xeneralizar e ocurrirá o que xa se comentou: a solución $\hat{\beta}^R$ tenderá a contar con compoñentes non nulas e esto dificultará a interpretación real do modelo. Se tiveramos tres variables ($p = 3$), a rexión $\| \beta \|_2^2$ da regresión Ridge convertiríase nunha esfera, mentres que a rexión $\| \beta \|_1^2$ de LASSO sería un poliedro. Cando $p > 3$, a rexión na regresión Ridge convírtese nunha hiperesfera e en LASSO obteríamos un politopo. En particular, LASSO selecciona variables cando $p > 2$ debido as esquinas do poliedro ou politopo.

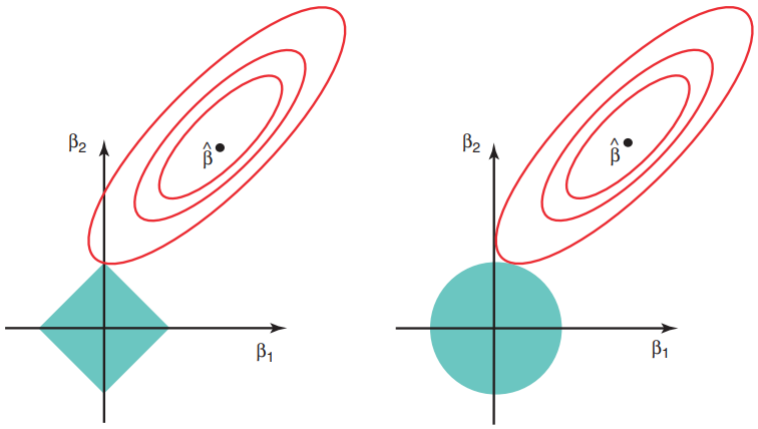

Figura 2.2: Interpretación de LASSO (esquerda) e regresión Ridge (dereita). Fonte da imaxe ([5]).

Ata agora estivemos vendo que tipos de problemas se nos presentan ao estimar parámetros en regresión. No seguinte capítulo veremos en que consisten estes problemas e porqué son convexos, así como a ventaxa desta condición. Ademáis, veremos algunhas alternativas ao método de mínimos cadrados para resolver ditos problemas.

Capítulo 3

Optimización convexa

Neste capítulo imos ver en que consisten os problemas de optimización convexa, además da importancia da condición de convexidade, pois son os casos que, habitualmente, se nos presentan ao estimar parámetros en regresión. Por outra parte, veremos métodos para resolver este tipo de problemas([2]).

Definición 3.1. Un problema de optimización convexa é da forma

$$\begin{aligned} &\text{mín} && f_0(\vec{x}) \\ &\text{s. a.} && f_i(\vec{x}) \leq b_i, \qquad i = 1, ..., m, \end{aligned}$$

onde as funcións $f_0, ..., f_m : \mathbb{R}^n \to \mathbb{R}$ son convexas, é dicir, satisfan

$$f_i(\phi\vec{x} + (1-\phi)\vec{y}) \leq \phi f_i(\vec{x}) + (1-\phi) f_i(\vec{y})$$

para todo $\vec{x}, \vec{y} \in \mathbb{R}^n$ e todo $\phi \in [0, 1]$. O problema de mínimos cadrados é un bo exemplo deste tipo de problemas, o cal empregamos na regresión lineal tal e como vimos no capítulo 1.

Agora presentaremos un par de resultados para ilustrar o bo comportamento das funcións convexas. Además, definimos un par de conceptos previos necesarios para a comprensión de ditos resultados (véxase [3]).

Definición 3.2 (Mínimo local)**.** Dado $\vec{x_0} \in V$, dise que é mínimo local da función $f : V \subset \mathbb{R}^n \to \mathbb{R}$ se existe un entorno de $\vec{x_0}$, $U(\vec{x_0})$, tal que

$$\forall \vec{x} \in U(\vec{x_0}) \cap V \qquad f(\vec{x_0}) \leq f(\vec{x}).$$

Definición 3.3 (Mínimo global)**.** Dado $\vec{x_0} \in V$, dise que é un mínimo global da función $f : V \subset \mathbb{R}^n \to \mathbb{R}$ se

$$f(\vec{x_0}) \leq f(\vec{x}) \qquad \forall \vec{x} \in V.$$

Proposición 3.4. *Sexa $f : V \subset \mathbb{R}^n \to \mathbb{R}$ convexa. Se $\vec{x}$ é un mínimo local de f entón $\vec{x}$ é tamén un mínimo global de f.*

Demostración. Sexa $\vec{x}$ un mínimo local de f, sexa $\vec{y} \in V$ e sexa $\phi > 0$ suficientemente pequeno para que $(1-\phi)\vec{x} + \phi\vec{y}$ pertenza ao entorno de $\vec{x}$ onde este minimiza a función. Entón

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

xa que na primeira desigualdade empregamos que $\vec{x}$ é mínimo local, e a segunda tense por ser f convexa. De ditas desigualdades, podemos deducir que $f(\vec{x}) - (1-\phi)f(\vec{x}) \leq \phi f(\vec{y})$ e, polo tanto, $f(\vec{x}) \leq f(\vec{y})$ verificando a definición de mínimo global. □

Proposición 3.5 (Caracterización do mínimo dunha función convexa)**.** *Dados $V \subset \mathbb{R}^n$ cerrado e convexo, $U \subset \mathbb{R}^n$ aberto e $f : V \to \mathbb{R}$ convexa e diferenciable en $U \supset V$, entón tense que*

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

Demostración. Para a condición suficiente probarase primeiro que

$$f(\vec{x}) - f(\vec{y}) \leq \nabla f(\vec{x})^T(\vec{x} - \vec{y}), \qquad \forall \vec{x}, \vec{y} \in V. \tag{3.1}$$

Sexa $\phi \in [0,1]$ e sexa $\vec{h} = \phi(\vec{y} - \vec{x})$. Para todo $\vec{x}, \vec{y} \in V$ basta ver que se verifica, grazas á convexidade de f

$$\begin{aligned} f(\vec{y}) \leq f(\vec{x}) + \frac{f((1-\phi)\vec{x} + \phi\vec{y}) - f(\vec{x})}{\phi} = f(\vec{x}) + \frac{f(\vec{x} - \phi\vec{x} + \phi\vec{y}) - f(\vec{x})}{\phi} = \\ = f(\vec{x}) + \frac{f(\vec{x} - \phi(\vec{x} + \vec{y})) - f(\vec{x})}{\phi} = f(\vec{x}) + \frac{f(\vec{x} + \vec{h}) - f(\vec{x})}{\vec{h}}(\vec{y} - \vec{x}) \to \\ \to f(\vec{x}) + \nabla f(\vec{x})^T(\vec{y} - \vec{x}) \qquad \text{cando} \qquad \phi \to 0 (\vec{h} \to 0). \end{aligned}$$

Se existe $\vec{x^*} \in V$ tal que para todo $\vec{y} \in \mathbb{R}^n$, $\nabla f(\vec{x^*})^T(\vec{x^*} - \vec{y}) \leq 0$ aplicando o anterior chégase a que $f(\vec{x^*}) \leq f(\vec{y})$, polo tanto $\vec{x^*} \in argmin_{\vec{x} \in X} f(\vec{x})$.

Para demostrar a condición necesaria considérase para cada $\vec{y} \in V$ a función real de variable real $h(t) := f(\vec{x^*} + t(\vec{y} - \vec{x^*}))$ cuxa derivada é $h'(t) = \nabla f(\vec{x^*} + t(\vec{y} - \vec{x^*}))^T(\vec{y} - \vec{x^*})$. Dado que $h(0) = f(\vec{x^*})$, 0 é un punto mínimo da función h. En particular, $h'(0) = \nabla f(\vec{x^*})^T(\vec{y} - \vec{x^*}) \geq 0$, de onde se obtén o resultado. □

Antes de ver algún exemplo de problema de optimización convexa, a seguinte sección resultaranos de gran utilidade xa que nos amosará información sobre as funcións que se nos presentarán neste tipo de problemas.

3.1. Funcións convexas

Nesta sección veremos algúns exemplos de funcións convexas que nos serán de utilidade para comprender que os problemas de optimización expostos máis adiante son, efectivamente, convexos.

A aplicación da definición de convexidade a unha función pode resultar complicado, polo que se recorre ás caracterizacións, é dicir, a certas condicións que poden verificar as funcións e que nos permiten ver se son convexas. Para poder comprender un resultado que daremos a continuación, necesitamos un par de definicións previas.

Definición 3.6. Sexa $f : \mathbb{R}^n \to \mathbb{R}$ unha función dúas veces continuamente diferenciable. A matriz hessiana da función f é a matriz cadrada $n \times n$ formada polas segundas derivadas parciales de f. Ademáis, o teorema de Schwarz aseguranos que non importa a orde de derivación, polo que a matriz hessiana é simétrica. Denotaremos a matriz hessiana de f nun punto x por $H_f(x)$. Se queremos máis detalles sobre o teorema de Schwarz, véxase ([9]).

Definición 3.7. Dada unha matriz simétrica $A \in M_n(\mathbb{R})$, tense que:

$$A \text{ é semidefinida positiva} \Leftrightarrow x^T Ax \geq 0, \qquad \forall x \in \mathbb{R}^n.$$

En particular,

$$A \text{ é definida positiva} \Leftrightarrow x^T Ax > 0, \qquad \forall x \in \mathbb{R}^n.$$

Sabendo isto, podemos enunciar a seguinte proposición:

Proposición 3.8. *Dada unha función $f : \mathbb{R}^n \to \mathbb{R}$ dúas veces diferenciable, cúmprese que:*

$$f \text{ é convexa se, e só se, } H_f(x) \text{ é semidefinida positiva}$$

onde H é a matriz hessiana de f.

Demostración. Supoñamos que f é convexa en todo punto $\vec{z} \in \mathbb{R}^n$. Temos que probar que $\vec{x}^T H(\vec{z})\vec{x} \geq 0$ para todo $\vec{x} \in \mathbb{R}^n$. Para calquera $\vec{x} \in \mathbb{R}^n$, $\vec{z} + \lambda\vec{x} \in \mathbb{R}^n$ para $|\lambda| \neq 0$. Agora, por (3.1) e que f sexa dúas veces diferenciable, obtemos as seguintes expresións:

$$f(\vec{z} + \lambda\vec{x}) \geq f(\vec{z}) + \lambda \nabla f(\vec{z})^T \vec{x}, \tag{3.2}$$

$$f(\vec{z}+\lambda\vec{x}) = f(\vec{z}) + \lambda\nabla f(\vec{z})^T\vec{x} + \frac{1}{2}\lambda^2\vec{x}^T H(\vec{z})\vec{x} + \lambda^2 \parallel \vec{x} \parallel^2 O(\lambda^2). \tag{3.3}$$

Substituíndo (3.3) en (3.2), obtemos

$$\frac{1}{2}\lambda^2\vec{x}^T H(\vec{z})\vec{x} + \lambda^2 \parallel \vec{x} \parallel^2 O(\lambda^2) \geq 0.$$

Dividindo entre λ^2 e facendo que λ tenda a 0, séguese que $\vec{x}^T H(\vec{z})\vec{x} \geq 0$.

Para a outra implicación, supoñamos que H é semidefinida positiva. Consideramos $\vec{x}$ e $\vec{z}$ puntos de $\mathbb{R}^n$. Entón, polo teorema do valor medio (véxase [8], p. 174), temos

$$f(\vec{x}) = f(\vec{z}) + \nabla f(\vec{z})^T(\vec{x}-\vec{z}) + \frac{1}{2}(\vec{x}-\vec{z})^T H(\hat{x})(\vec{x}-\vec{z}) \tag{3.4}$$

onde $\hat{x} = \lambda\vec{z} + (1-\lambda)\vec{x}$ para algún $\lambda \in (0,1)$. Como asumimos que H é semidefinida positiva temos que $(\vec{x}-\vec{z})^T H(\hat{x})(\vec{x}-\vec{z}) \geq 0$ e, xunto con (3.4), concluimos que

$$f(\vec{x}) \geq f(\vec{z}) + \nabla f(\vec{z})^T(\vec{x}-\vec{z}).$$

Posto que a desigualdade anterior é certa para cada $\vec{x}, \vec{y} \in \mathbb{R}^n$, f é convexa en virtude de (3.1). □

Dito isto, podemos comprobar que as seguintes funcións son convexas (ben pola definición ou ben, facendo uso da proposición anterior).

- $f(x) = x^2$ é convexa, pois a segunda derivada correspóndese con $f''(x) = 2 > 0$.
- As funcións afíns $f(\vec{x}) = A\vec{x} + \vec{b}$ son convexas. Comprobémolo facendo uso da definición:

$$\begin{aligned} &f(\phi\vec{x} + (1-\phi)\vec{y}) = A(\phi\vec{x} + (1-\phi)\vec{y}) + \vec{b} = A\phi\vec{x} + A\vec{y} - A\phi\vec{y} + \vec{b} = \\ &= \phi A\vec{x} + \phi\vec{b} + A\vec{y} - \phi A\vec{y} + \vec{b} - \phi\vec{y} = \phi f(\vec{x}) + (1-\phi)f(\vec{y}) \end{aligned}$$

- As funcións norma $f(\vec{x}) = \parallel \vec{x} \parallel_q = (\sum_{i=1}^n |x_i|^q)^{1/q}$ para $q \geq 1$ son funcións convexas. A condición $q \geq 1$ é necesaria pois para demostrar que estas normas son convexas, usaremos a desigualdade triangular que se verifica para estes valores de q.
  Entón, pola desigualdade triangular tense que

$$\left(\sum_{i=1}^n |x_i + y_i|^q\right)^{1/q} \leq \left(\sum_{i=1}^n |x_i|^q\right)^{1/q} + \left(\sum_{i=1}^n |y_i|^q\right)^{1/q}$$

Polo tanto,

$$f(\phi\vec{x}+(1-\phi)\vec{y}) = \| \phi\vec{x}+(1-\phi)\vec{y} \|_q = \left(\sum_{i=1}^{n} |\phi x_i + (1-\phi)y_i|^q\right)^{1/q} \leq$$
$$\leq \left(\sum_{i=1}^{n} |\phi x_i|^q\right)^{1/q} + \left(\sum_{i=1}^{n} |(1-\phi)y_i|^q\right)^{1/q} =$$
$$= \phi\left(\sum_{i=1}^{n} |x_i|^q\right)^{1/q} + (1-\phi)\left(\sum_{i=1}^{n} |y_i|^q\right)^{1/q} = \phi f(\vec{x}) + (1-\phi)f(\vec{y})$$

- A combinación lineal con coeficientes positivos (ou combinación cónica) de funcións convexas é convexa. Sexan $f : \mathbb{R}^n \to \mathbb{R}$ e $g : \mathbb{R}^n \to \mathbb{R}$ dúas funcións convexas e $a, b \in \mathbb{R}^+$. Entón $af + bg$ é convexa xa que

$$af(\phi\vec{x}+(1-\phi)\vec{y})+bg(\phi\vec{x}+(1-\phi)\vec{y}) \leq a[\phi f(\vec{x})+(1-\phi)f(\vec{y})]+b[\phi g(\vec{x})+(1-\phi)g(\vec{y})]$$

Agora que vimos algunhas funcións convexas, na seguinte sección veranse problemas que tratan de minimizar algunha destas funcións.

3.2. Exemplos

A continuación mostramos un par de exemplos de problemas de optimización convexa que teñen especial interese pois son os que se utilizan na regresión lineal para a estimación de parámetros tal e como se pode ver en ([2]).

3.2.1. Problema dos mínimos cadrados

Trátase dun problema de optimización sen restriccións no que, dado un conxunto de pares ordenados intentamos encontrar a función continua que mellor se aproxime aos datos. Ao elevar ó cadrado a función obxectivo que queremos minimizar, obtemos o problema de aproximación por mínimos cadrados,

$$\text{mín} \quad \| A\vec{x} - \vec{b} \|^2$$

onde a función obxectivo é a suma dos residuos ao cadrado que, no caso da regresión lineal, coñecemos por RSS, tal e como vimos no capítulo 1. Este problema pode resolverse analíticamente expresando a función de forma cuadrática convexa

$$f(\vec{x}) = \vec{x}^T A^T A\vec{x} - 2\vec{b}^T A\vec{x} + \vec{b}^T\vec{b}.$$

O punto $\vec{x}$ minimiza f se, e só se

$$\nabla f(\vec{x}) = \vec{x}^T A\vec{x} - 2A^T\vec{b} = 0$$

é dicir, se, e só se, $\vec{x}$ satisfai as chamadas ecuacións normais

$$A^T A\vec{x} = A^T\vec{b},$$

que sempre teñen solución. Además, se asumimos que as columnas de A son independentes, dito problema ten solución única $\vec{x} = (A^T A)^{-1}A^T\vec{b}$.

O problema de mínimos cadrados é a base para a estimación de parámetros, regresión e diversos métodos de axuste de datos. O cal se utiliza en moitas técnicas como, por exemplo, na regularización, que veremos a continuación.

3.2.2. Regularización

Un problema común de regularización, especialmente cando se utiliza a norma euclídea, é reducir o mínimo da suma ponderada das normas ao cadrado, é dicir,

$$\text{mín} \quad \| A\vec{x} - \vec{b} \|^2 + \delta \| \vec{x} \|^2,$$

onde $\delta > 0$ é un parámetro do problema.

Estes problemas de aproximación regularizada resolven esta situación facendo $\| A\vec{x} - \vec{b} \|$ e $\| \vec{x} \|$ pequenos, engadindo un termo de penalización asociado á norma de $\vec{x}$.

Os exemplos anteriores son claramente problemas de optimización convexa pois as correspondentes funcións obxectivo son convexas como vimos na sección 3.1. Para resolver ditos problemas existen diversas formas de conseguilo como, por exemplo, analíticamente. Sen embargo, existen alternativas para resolvelos e que plantexamos na seguinte sección.

3.3. Métodos iterativos

Cómpre destacar que non sempre existe solución analítica, pois se traballamos coa norma l_1 de $\vec{x}$ como ocorre co método LASSO visto na sección 2.2 do capítulo anterior, esta non é diferenciable e necesitamos facer uso doutras ferramentas para resolver o problema. A continuación, explicaremos unha serie de métodos iterativos para resolver os problemas de optimización mencionados previamente, pero antes, daremos algúns criterios teóricos

que nos permitan comparar o desempeño dos algoritmos que se empregan nestes métodos.

Dado un algoritmo, o que facemos con él é xerar unha sucesión $\vec{x}_k$ que tenda ó óptimo do problema. O n-ésimo termo da sucesión xerámolo a partir de minimizar unha función auxiliar dunha soa variable ao longo dunha recta que pasa polo punto $\vec{x}_{n-1}$. A converxencia do algoritmo vai depender de como converxa a sucesión que xera.

Diremos que un algoritmo ten converexencia global se a sucesión que xera converxe independentemente do punto inicial x_0 que se escolla. Se, pola contra, a súa converxencia depende do punto inicial, dise que ten converxencia local. Para escoller que método utilizar, necesitamos un criterio para decidir, entre as sucesións que xeran, cal converxe máis rápido.

Definición 3.9. Diremos que unha sucesión converxe linealmente a un punto $\vec{s}$ se existe $k \in (0,1)$ tal que

$$\| \vec{x}_{n+1} - \vec{s} \| \leq k \| \vec{x}_n - \vec{s} \| .$$

De igual modo, diremos que unha sucesión converxe cuadráticamente se

$$\| \vec{x}_{n+1} - \vec{s} \| \leq k \| \vec{x}_n - \vec{s} \|^2 .$$

Diremos que un algoritmo converxe de certa forma se a sucesión que xera converxe desa forma. Un algoritmo con converxencia cuadrática converxe máis rápido que un con converxencia lineal e dados dous algoritmos lineais, converxe máis rápido aquel que teña un menor k.

Por outro lado, esto só nos dá un aspecto do desempeño dun algoritmo. Tamén hai que ter en conta o número de evaluacións da función, do seu gradiente e da matriz hessiana que se requiren en cada iteración. Dito isto, podemos proceder a explicación dos distintos métodos, entre os que se encontran o método de Hooke e Jeeves, o descenso por gradiente, o método de Newton, o método do gradiente conxugado e o dos subgradientes (Véxase [1]).

3.3.1. Método de Hooke e Jeeves

Un dos primeiros métodos que veremos é o de Hooke e Jeeves, pois non necesitamos calcular nin o gradiente nin a matriz hessiana da función que queremos minimizar durante o proceso que segue e, polo tanto, dita función non necesita ser diferenciable. O algoritmo comeza dende un punto inicial, a continuación determinamos mediante unha regra unha dirección de movemento, e seguimos nesa dirección ata chegar a un mínimo da función

obxectivo sobre esa recta. O proceso de búsqueda do mínimo sobre a recta chámase búsqueda lineal. Outros métodos, en ausencia de diferenciabilidade, poden situarse nun punto non óptimo. Para evitar esta situación podemos introducir unha búsqueda na dirección $x_{k+1} - x_k$ en cada paso do algoritmo, para evitar problemas con puntos onde a función obxectivo non é diferenciable. Polo tanto, o método de Hooke e Jeeves emprega dous tipos de búsqueda.

O proceso é o seguinte: dado un punto inicial x_1, defínese $z_1 = x_1$. A búsqueda lineal ao longo das direccións coordenadas produce un novo punto x_2. Despois, a búsqueda de patróns ao longo da dirección $x_2 - x_1$ conduce ao punto z_2. Outra búsqueda exploratoria empezando en z_2 danos o punto x_3. A seguinte búsqueda de patróns realízase ao longo da dirección $x_3 - x_2$, dándonos o novo punto z_3. O proceso vaise repetindo ata a converxencia.

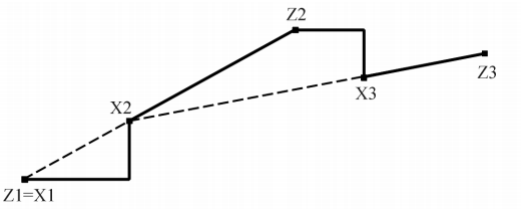

Figura 3.1: Ilustración do método de Hooke e Jeeves. Imaxe tomada de ([1]).

Un resumo do método é o seguinte:
Paso de inicialización: Sexan $\vec{d_1}, ..., \vec{d_n}$ as direccións coordenadas. Sexa $\delta > 0$, un escalar que determinará cando parar o algoritmo. Además, elíxese un tamaño de paso $\lambda > \delta$ e un factor de aceleración $\gamma > 0$. Elíxese un punto inicial x_1 e fíxase $z_1 = x_1$, $k = j = 1$. Agora, imos a iteración xeral.
Iteración k:

1. Se $f(z_j + \lambda\vec{d_j}) < f(z_j)$, entón sexa $z_{j+1} = z_j + \lambda\vec{d_j}$, e imos ao paso 2. Se, pola contra, $f(z_j + \lambda\vec{d_j}) \geq f(z_j)$, poden pasar dúas cousas:
   - Se $f(z_j - \lambda\vec{d_j}) < f(z_j)$, sexa $z_{j+1} = z_j - \lambda\vec{d_j}$ e imos ao paso 2.
   - Se $f(z_j - \lambda\vec{d_j}) \geq f(z_j)$, sexa $z_{j+1} = z_j$ e vamos ao paso 2.
2. Se $j < n$, reemplazamos j por $j + 1$ e repetimos o paso 1. En caso contrario:
   - Imos ao paso 3 se $f(z_{n+1}) < f(x_k)$.

- Imos ao paso 4 se $f(z_{n+1}) \geq f(x_k)$.

3. Sexa $x_{k+1} = z_{n+1}$, e sexa $z_1 = x_{k+1} + \gamma(x_{k+1} - x_k)$. Reemplazamos k por $k+1$, faise $j = 1$ e volvemos ao paso 1.

4. Se $\lambda \leq \delta$, paramos; x_k é a solución. Noutro caso, cambiamos λ por $\lambda/2$. Sexa $z_1 = x_k$, $x_{k+1} = x_k$, sustituimos k por $k+1$, faise $j = 1$ e volvemos ao paso 1.

Os seguintes métodos que veremos precisarán que a función que se queira minimizar sexa ata dúas veces continuamente diferenciable pois faran uso do gradiente e da matriz hessiana definida na sección 3.1.

3.3.2. Descenso por gradiente

O método do descenso por gradiente ([3]) é un dos métodos máis coñecidos para encontrar o mínimo dunha función $f : \mathbb{R}^n \to \mathbb{R}$ diferenciable. Este método aproveita a información ofrecida polo gradiente dunha función, é dicir, cara onde crece a función, para moverse en sentido oposto, pois o obxectivo é ir descendendo ata encontrar o mínimo. Partindo dun punto $\vec{x}_1 \in \mathbb{R}^n$ realízase a seguinte iteración para cada $k \geq 1$

$$\vec{x}_{k+1} = \vec{x}_k - \eta \nabla f(\vec{x}_k) \tag{3.5}$$

onde $\eta > 0$ é un parámetro fixo. A razón da expresión anterior é a de moverse na dirección de máximo descenso, a cal se corresponde con $-\nabla f(\vec{x}_k)$. Se en cada paso, o desprazamento realízase na dirección $\vec{u} \in \mathbb{R}^n$ unitario, é dicir, de $\vec{x}_k$ a $\vec{x_{k+1}} = \vec{x_k} + \eta\vec{u}$, deséxase que $f(\vec{x}_{k+1}) - f(\vec{x}_k)$ sexa o menor posible pois esto indica que o descenso nese paso é máximo. Se se define $g : \mathbb{R} \to \mathbb{R}$ por $g(\eta) = f(\vec{x}_k + \eta\vec{u})$ pódese ver, utilizando o polinomio de Taylor de orde 1 de g que

$$f(\vec{x}_{k+1}) - f(\vec{x}_k) = g(\eta) - g(0) = g'(0)\eta + O(\eta^2).$$

Como pola regla da cadea $g'(\eta) = \nabla f(\vec{x}_k + \eta\vec{u})^T\vec{u}$ obtense

$$f(\vec{x}_{k+1}) - f(\vec{x}_k) = \eta \nabla f(\vec{x}_k)^T\vec{u} + O(\eta^2)$$

e o minimizador desta última expresión en $\vec{u}$ unitario é $\dfrac{-\nabla f(\vec{x}_k)}{\| \nabla f(\vec{x}_k) \|}$.

Agora ben, non só é importante a dirección de búsqueda. Unha lonxitude de paso, η, imprecisa pode xerar unha demora considerable do proceso ou incluso que nunca se chegue a alcanzar un mínimo con tolerancia menor que un certo $\varepsilon > 0$. Hai varias fórmulas para

calcular a lonxitude de paso, as cales se coñecen como regras de búsqueda lineal. Entre elas destacan a regra de Wolfe, a regra de Goldstein-Price ou a regra de Armijo, a cal explicamos brevemente a continuación (capítulo 8 de [1]).

Supoñamos que estamos minimizando algunha función diferenciable $f : \mathbb{R}^n \to \mathbb{R}$ no punto $\bar{x} \in \mathbb{R}^n$ na dirección $\vec{d} \in \mathbb{R}^n$ tal que $\nabla f(\bar{x})^T \vec{d} < 0$. Por tanto, $\vec{d}$ é unha dirección de descenso. Definimos a función de búsqueda lineal, $\theta : \mathbb{R} \to \mathbb{R}$, como $\theta(\lambda) = f(\bar{x} + \lambda\vec{d})$ para $\lambda \geq 0$. Entón, a aproximación de primeira orde de θ en $\lambda = 0$ ven dada por $\theta(0) + \lambda\theta'(0)$. Agora, definimos

$$\hat{\theta}(\lambda) = \theta(0) + \lambda\gamma\theta'(0) \quad \text{para } \lambda \geq 0, \gamma \in (0, 1).$$

A regra de Armijo determina un intervalo onde se encontran os valores de λ aceptables. Para conseguilo, aplica dous criterios. O primeiro consiste en determinar un $\hat{\lambda}$ que satisfaga que $\theta(\hat{\lambda})$ esté por debaixo da recta $\theta(0) + \gamma\theta'(0)$, é dicir

$$\theta(\hat{\lambda}) \leq \theta(0) + \gamma\hat{\lambda}\theta'(0). \tag{3.6}$$

Para evitar que o tamaño de paso, λ, sexa moi pequeno e, así, avancemos moi lento, Armijo suxire comprobar se para múltiplos de $\hat{\lambda}$ se segue verificando (3.6). O procedemento que se segue é o seguinte: determínase o mínimo $j \in \mathbb{N}$ para o cal

$$\theta(2^j\hat{\lambda}) > \theta(0) + \gamma 2^j\hat{\lambda}\theta'(0).$$

Entón o intervalo onde λ é aceptable é $(\hat{\lambda}, 2^j\hat{\lambda})$ e adoitase escoller $\lambda = 2^{j-1}\hat{\lambda}$.

Dito isto, na maioría dos problemas de optimización que se tratan, f está definida en $V \subsetneq \mathbb{R}^n$ polo que non se ten a certeza de que $\vec{x}_{t+1} \in V$ na ecuación (3.5). A forma máis obvia de solventar dita situación é tomar o punto máis próximo que sí cumpre esta propiedade. Sen embargo, non temos garantías de que este punto sexa único. Non obstante, os problemas de optimización que se nos presentan son sen restriccións polo que non profundizaremos nese sentido. Por último, engadir que este tipo de métodos son independentes da dimensión, polo que son bastante atractivos para optimizar en grandes dimensións.

3.3.3. Método de Newton

O método do descenso polo gradiente converxe moi lentamente debido a que súa orde de converxencia é lineal. A continuación estudiarase o método de Newton (ou método de Newton-Raphson) que ten converxencia cuadrática cerca do mínimo. Para entrar máis en detalle sobre os distintos tipos de converxencia véxase ([1], p. 257).

O método de Newton é un método diseñado para converxer nunha soa iteración cando a función a minimizar é cuadrática. O método consiste en minimizar en cada iteración k unha función cuadrática $G_k(\vec{x})$ que se obtén ao extender en serie de Taylor a función que queremos minimizar, f, ao redor de x_k ata o segundo termo. É dicir, $G_k(\vec{x})$ é igual a

$$G_k(\vec{x}) = f(\vec{x}_k) + (\vec{x} - \vec{x}_k)^T \nabla f(\vec{x}_k) + \frac{1}{2}(\vec{x} - \vec{x}_k)^T H_f(\vec{x}_k)(\vec{x} - \vec{x}_k).$$

Para obter o mínimo de $G_k(\vec{x})$, calculamos o seu gradiente e igualámolo a cero

$$\nabla G_k(\vec{x}) = \nabla f(\vec{x}_k) + H_f(\vec{x}_k)(\vec{x} - \vec{x}_k) = 0,$$

entón, o mínimo alcánzase en

$$\vec{x} = \vec{x}_k - H_f(\vec{x}_k)^{-1} \nabla f(\vec{x}_k)$$

sempre que a matriz hessiana de f en $\vec{x}_k$ sexa definida positiva. O método de Newton escolle en cada paso como o punto $\vec{x}_{k+1}$ ao mínimo da función $G_k(\vec{x})$.

O algoritmo de Newton é o seguinte: Dada unha función f dúas veces continuamente diferenciable nunha veciñanza V do mínimo x^*, $\vec{x}_0$ en $V(\vec{x}^*)$ e $\delta > 0$ como tolerancia:

1. Determinar $\vec{d_k}$ resolvendo o sistema

$$H_f(\vec{x}_k)\vec{d_k} = -\nabla f(\vec{x}_k). \tag{3.7}$$

2. Calcular $\vec{x}_{k+1}$ por medio da expresión

$$\vec{x}_{k+1} = \vec{x}_k + \vec{d_k}. \tag{3.8}$$

3. Se $\| \nabla f(\vec{x}_{k+1}) \| \leq \delta$ e $\dfrac{\| \vec{x}_{k+1} - \vec{x}_k \|}{\| \vec{x}_{k+1} \|} \leq \delta$ entón tómase $\vec{x}^* \approx \vec{x}_{k+1}$.

4. Se non se cumpre o anterior, regrésase ao paso 1 e calcúlase $\vec{x}_{k+2}$.

Observemos que a expresión (3.7) indícanos que en cada iteración se escolle como dirección $\vec{d_k} = -H_f(\vec{x}_k)^{-1} \nabla f(\vec{x}_k)$, polo que Newton é un método de descenso xa que

$$\nabla f(\vec{x}_k)^T \vec{d_k} = -\nabla f(\vec{x}_k)^T H_f(\vec{x}_k)^{-1} \nabla f(\vec{x}_k) \leq 0$$

e esto cúmprese sempre que $H_f(\vec{x}_k)$ sexa unha matriz semidefinida positiva para cada iteración k. Esta última condición deber restrinxirse a que $H_f(\vec{x}_k)$ sexa estrictamente definida positiva para garantir que o sistema de ecuacións a resolver admita unha única solución.

Polo tanto, Newton converxerá sempre que a veciñanza do mínimo que se escolla sexa suficientemente pequena como para poder garantir que a matriz hessiana, evaluada en calquer punto de esa veciñanza, sexa definida positiva.

O cálculo da dirección require coñecer a matriz hessiana, polo que se clasifica o método de Newton como un método de tipo Hessiano en contraste co descenso polo gradiente, que só necesita esta información para calcular a dirección.

Por outro lado, o método de Newton pódese modificar para controlar o paso en cada iteración. Neste caso, o paso 2 cámbiase por

$$\vec{x}_{k+1} = \vec{x}_k + \lambda \vec{d}_k$$

onde λ se pode obter usando a regra de Armijo explicada no anterior método.

Debido ás dificultades que presenta Newton, cómpre considerar outros métodos que teñan converxencia cuadrática.

3.3.4. O gradiente conxugado

O método do gradiente conxugado ten converxencia cuadrática. Antes de explicar en que consiste, vexamos a seguinte definción.

Definición 3.10. Dise que $\{\vec{d}_k\}_{k=1}^n$ son vectores mutuamente conxugados respecto a unha matriz A simétrica e definida positiva se

$$\vec{d}_k^T A \vec{d}_j = 0 \quad \forall j \neq k. \tag{3.9}$$

A idea do método do gradiente conxugado é tomar un método de descenso no que as direccións sexan conxugadas respecto da matriz hessiana da función que se dexesa minimizar. Dada unha función $f : \mathbb{R}^n \to \mathbb{R}$ dúas veces continuamente diferenciable, un punto inicial $\vec{x}_0$, $\vec{d}_0 = -\nabla f(\vec{x}_0)$ e $\delta > 0$ para a tolerancia, o algoritmo é o seguinte:

1. Definimos $k = 0, 1, ...$

$$\vec{x}_{k+1} = \vec{x}_k + r_k \vec{d}_k,$$

con

$$r_k = -\frac{\nabla f(\vec{x}_k)^T \vec{d}_k}{\vec{d}_k^T H_f \vec{d}_k}.$$

2. A dirección $\vec{d}_{k+1}$ calcúlase como

$$\vec{d}_{k+1} = -\nabla f(\vec{x}_{k+1}) + c_k \vec{d}_k,$$

con

$$c_k = \frac{\nabla f(\vec{x}_{k+1})^T H_f \vec{d}_k}{\vec{d}_k^T H_f \vec{d}_k}.$$

3. Se $\parallel \nabla f(\vec{x}_{k+1}) \parallel \leq \delta$ e $\dfrac{\parallel \vec{x}_{k+1} - \vec{x}_k \parallel}{\parallel \vec{x}_k \parallel} \leq \delta$ entón tomamos $\vec{x}^* \approx \vec{x}_{k+1}$ onde $\vec{x}^*$ é o mínimo da función f.

4. Se non se cumpre o anterior, regresamos ao paso 2 e calculamos $\vec{x}_{k+2}$.

Agora ben, estes métodos (a excepción de Hooke e Jeeves) necesitan que a función que se queira minimizar sexa ata dúas veces continuamente diferenciable. Por este motivo, veremos o seguinte método que non necesita desta condición para poder empregalo.

3.3.5. Método do subgradiente

Este método, ao igual que Hooke e Jeeves, non require que a función que queremos minimizar sexa diferenciable e, polo tanto, podemos utilizalo para a estimación de parámetros no modelo LASSO. Sen embargo, antes de explicar este método iterativo necesitamos a seguinte definición:

Definición 3.11. Sexa $f : \mathbb{R}^n \to \mathbb{R}$ unha función convexa. Entón ξ é subgradiente de f nun punto x se

$$f(y) \geq f(x) + \langle \xi, y - x \rangle \quad \forall y \in \mathbb{R}^n.$$

O conxunto de tódolos subgradientes en x é coñecido como o subdiferencial de x, é dicir, o subdiferencial de f en x está definido polo conxunto

$$\partial f(x) = \{\xi : f(y) \geq f(x) + \langle \xi, y - x \rangle, \forall y \in \mathbb{R}^n\}.$$

Cabe destacar que no caso de que f sexa unha función diferenciable, o subgradiente é único e coincide co gradiente (véxase [7]).

Dito isto, consideramos o problema

$$\text{mín} \qquad \{f(q) : q \in V\} \tag{3.10}$$

onde $f : \mathbb{R}^n \to \mathbb{R}$ é unha función convexa non necesariamente diferenciable e asumimos que existe unha solución óptima.

Agora, describimos un algoritmo de optimización que utiliza subgradientes e que pode ser visto como unha xeneralización do método de descenso por gradiente no que a dirección do gradiente negativo se substitúe por unha dirección basada en subgradientes negativos. Sen embargo, esta última dirección non ten por que ser necesariamente de descenso, ainda que dá lugar a que a nova iteración estea máis preto dunha solución óptima para un tamaño de paso suficientemente pequeno. Por esta razón, non realizamos unha búsqueda lineal ao longo da dirección do subgradiente negativo, senón que prescribimos un tamaño de paso en cada iteración. Isto garantiza que a secuencia xerada acabará converxendo a unha solución óptima.

Dado o iterante $q_k \in V$ e tomando o tamaño de paso λ_k ao longo da dirección $\vec{d_k} = -\xi_k / \parallel \xi_k \parallel$, onde ξ_k pertence á subdiferencial $\partial f(q_k)$ de f en q_k, o punto resultante $\bar{q}_{k+1} = q_k + \lambda_k \vec{d_k}$ non ten por que pertencer a V. En consecuencia, o novo iterante q_{k+1} obtense proxectando $\bar{q}_{k+1}$ en V, é dicir, encontrar o punto máis cercano en V a $\bar{q}_{k+1}$. Denotamos esta operación como $q_{k+1} = P_V(\bar{q}_{k+1})$, onde

$$P_V(\bar{q}) \equiv \text{argmin}\{\parallel q - \bar{q} \parallel : q \in V\}$$

A continuación, expoñemos en que consiste o algoritmo do subgradiente de maneira esquematizada:

- **Paso inicial**: Escollemos unha solución inicial $q_1 \in V$, e sexa $UB_1 = f(q_1)$ o límite superior do valor obxectivo, e denotamos a solución por $q^* = q_1$. Escribimos $k = 1$ e imos ao paso principal.

- **Paso principal**: Dado q_k, encontramos un subgradiente $\xi_k \in \partial f(q_k)$ de f en q_k. Se $\xi_k = 0$, entón paramos; q_k resolve o problema 3.10. Noutro caso, sexa $\vec{d_k} = \dfrac{-\xi_k}{\parallel \xi_k \parallel}$, escollemos o tamaño de paso $\lambda_k > 0$, e calculamos $q_{k+1} = P_V(\bar{q}_{k+1})$, onde $\bar{q}_{k+1} = q_k + \lambda_k \vec{d_k}$. Se $f(q_{k+1}) < UB_k$; poñemos $UB_{k+1} = f(q_{k+1})$ e $q^* = q_{k+1}$. Noutro caso, $UB_{k+1} \equiv UB_k$ e repetimos este paso.

Estes métodos iterativos son dos máis coñecidos aunque existen moitos máis (véxase [1]). O motivo polo cal nos centramos nestes é que temos métodos que non requiren de que a función obxectivo sexa diferenciable (necesarios para o caso LASSO), outros requiren do cálculo do gradiente e outros da matriz hessiana. No seguinte capítulo, aplicaremos ditos métodos a casos prácticos para ver o seu comportamento.

Capítulo 4

Implementación e comparativa entre os distintos métodos iterativos

Neste capítulo utilizaremos o software libre R ([6]) para implementar os métodos explicados anteriormente para os distintos modelos de regresión vistos nos dous primeiros capítulos deste traballo. Obteremos as estimacións dos distintos modelos facendo uso de diversas funcións de R e que explicaremos no seu momento para así, validar os diversos métodos comparando os seus resultados. Por último, compararemos os métodos entre eles para observar cal converxe en menos iteracións e ver a que resultados chegan.

4.1. Modelo de regresión lineal simple

Neste primeiro caso, de regresión lineal simple, simularemos unha serie de datos e faremos regresión sobre eles coa función `lm()` de R, a cal calcula os parámetros por mínimos cadrados. A ventaxa de simular os datos é que xa sabemos de antemán os valores de β_0 e β_1 pois os escollemos nós e polo tanto, podemos comprobar como de ben estiman os diversos métodos. Para este primeiro exemplo, tomaremos $\beta_0 = 2$ e $\beta_1 = 3$ cun erro que segue unha distribución normal de media 0 e desviación típica 0.2 (véxase o Anexo I). A razón de escoller este modelo sinxelo é poder ver gráficamente como cada método vai converxendo ao valor dos parámetros.

Simulados os datos, estamos en condición de facer regresión con eles. Ao comezo do primeiro capítulo obtivemos, de forma analítica, a expresión para ditos estimadores e, reproducindo ditas fórmulas conseguimos os valores $\hat{\beta}_0 = 2{,}037072$ e $\hat{\beta}_1 = 2{,}863475$ que, son practicamente iguais aos obtidos por mínimos cadrados calculados coa función `lm()`, como vemos a continuación

```
> lm(y~x)

Call:
lm(formula = y ~ x)

Coefficients:
(Intercept)            x
      2.037        2.863
```

onde $\hat{\beta}_0 = 2{,}037$ e $\hat{\beta}_1 = 2{,}863$. Sabendo estos resultados, vexamos como de rápido converxen os diversos métodos dende o punto (1,4) para este mesmo problema con un valor de tolerancia fixado, é dicir, un valor fixado para que o método pare. Os resultados obtidos móstranse na táboa 4.1.

	D. por gradiente	G. conxugado	Hooke e Jeeves	M. Newton
Nº iteracións	44	82	37	2
$\hat{\beta}_0$	2,037044	2,037037	2,029210	2,037072
$\hat{\beta}_1$	2,863523	2,863545	2,875757	2,863475

Táboa 4.1: Número de iteracións e solucións obtidas dos distintos métodos dende o punto inicial (1,4).

Como xa comentamos, a vantaxa de simular os datos é que xa sabemos a que valores se deben acercar as estimacións e, polo tanto, podemos escoller un punto de inicio para os métodos que esté cerca da solución. Podemos ver que ocorre no caso de escoller un punto inicial que esté máis alonxado da solución como, por exemplo, o (10,15) co que obtemos os seguintes datos:

	D. por gradiente	G. conxugado	Hooke e Jeeves	M. Newton
Nº iteracións	64	94	242	2
$\hat{\beta}_0$	2,037110	2,037039	2,0230182	2,037072
$\hat{\beta}_1$	2,863406	2,863541	2,874238	2,863475

Táboa 4.2: Número de iteracións e solucións obtidas dos distintos métodos dende o punto inicial (10,15).

Como era de esperar, o número de iteracións crece en todos os métodos excepto no de Newton que segue sendo de dúas iteracións. Cabe destacar o caso de Hooke e Jeeves, o cal pasa de converxer en 37 iteracións a facelo en 242. Isto débese a que é o método máis lento, en xeral, dos expostos no capítulo 3.

Por último, na figura 4.1 representamos graficamente a converxencia dos distintos métodos.

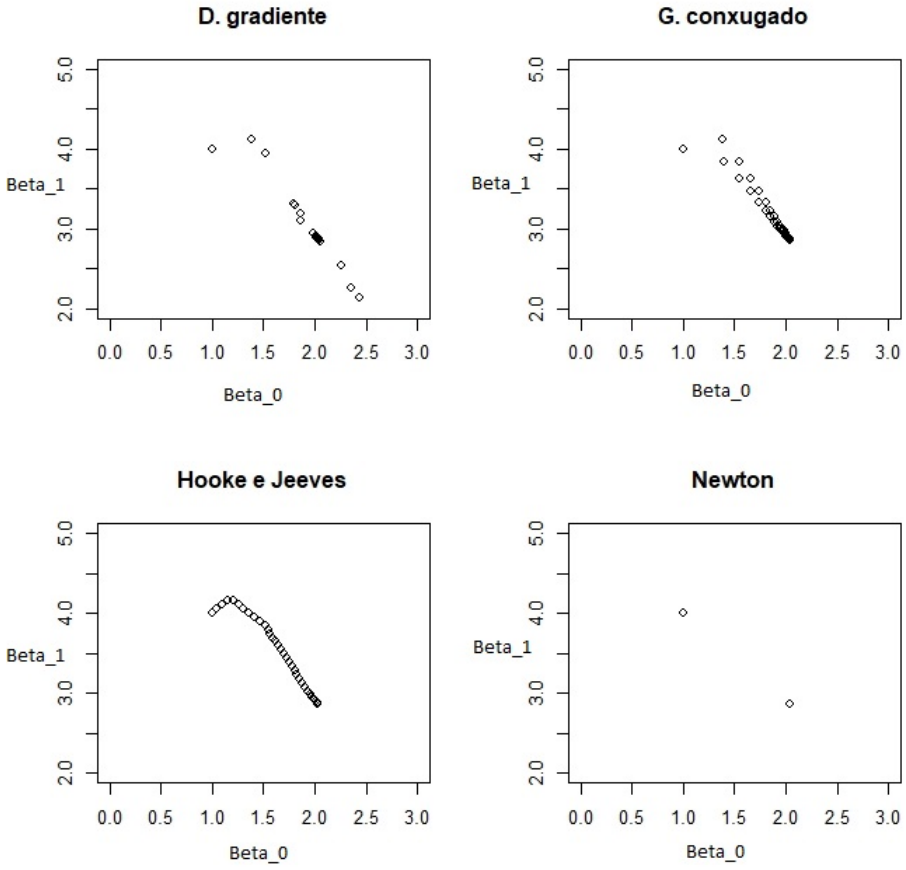

Figura 4.1: Representación gráfica da converxencia dos métodos dende o punto inicial (1,4).

4.2. Regresión lineal múltiple

Agora que temos validados os diversos métodos podemos aplicalos a un modelo de regresión lineal múltiple con datos reais. Para isto, o tema escollido será o cambio climático,

máis concretamente como afectan diversos factores á temperatura media global dende 1971 ata 2012. Para este propósito contamos con 42 observacións e as distintas variables espóñense a continuación:

- Variable resposta:
  - Temperaturas: Temperatura media global (en °C)
- Variables explicativas:
  - CO2: emisións de gases de efecto invernadoiro totais (kt equivalente de CO_2)
  - Combustibles.fósiles: uso de enerxía (kt equivalente de petróleo per capita)
  - Coches: producción mundial de vehículos de motor por cada 100000 habitantes.
  - PIB.Mundial: PIB (US$a precios actuais)

Dito isto, procedemos a facer regresión con estes datos coa mesma función `lm()` empregada no caso simple, obtendo as seguintes estimacións:

```
lm(Temperaturas~CO2+Combustibles.fosiles+Coches+PIB.Mundial)

Call:
lm(formula = Temperaturas ~ CO2 + Combustibles.fosiles + Coches +
PIB.Mundial)

Coefficients:
(Intercept)          CO2    Combustibles.fosiles       Coches    PIB.Mundial
  1.278e+01    1.214e-06                1.692e+00    3.761e-01     -1.014e-06
```

onde $\hat{\beta}_0 = 12{,}78$, $\hat{\beta}_1 = 1{,}214 \cdot 10^{-6}$, $\hat{\beta}_2 = 1{,}692$, $\hat{\beta}_3 = 0{,}3761$ e $\hat{\beta}_4 = -1{,}014 \cdot 10^{-6}$. Eses valores tradúcense en que tanto o CO_2, como a producción de coches e o PIB mundial non afectan prácticamente nada á temperatura global xa que están moi preto do 0. Sen embargo, podemos observar que un kilotón(kt) de uso de enerxía de combustibles fósiles aumenta 1,692°C a temperatura global.

Dito isto, aplicamos os distintos métodos iterativos expostos no capítulo 3 dende o punto inicial (11,1,3,2,-1) cun valor de tolerancia fixado e observamos que valores obtemos, recollidos na táboa 4.3.

	D. por gradiente	G. conxugado	Hooke e Jeeves	M. Newton
Nº iteracións	71	96	62	2
$\hat{\beta}_0$	12,78145	12,78044	12,77615	12,78271
$\hat{\beta}_1$	$1,217 \cdot 10^{-6}$	$1,215 \cdot 10^{-6}$	$1,293 \cdot 10^{-6}$	$1,223 \cdot 10^{-6}$
$\hat{\beta}_2$	1,69264	1,69241	1,70211	1,69295
$\hat{\beta}_3$	0,37614	0,37610	0,37988	0,37622
$\hat{\beta}_4$	$-1{,}016 \cdot 10^{-6}$	$-1{,}015 \cdot 10^{-6}$	$-1{,}019 \cdot 10^{-6}$	$-1{,}017 \cdot 10^{-6}$

Táboa 4.3: Número de iteracións e solucións obtidas dos distintos métodos dende o punto inicial (11,1,3,2,-1).

Podemos observar que os métodos se acercan con bastante exactitude aos valores obtidos coa función `lm()` a cal, como xa comentamos, procede co criterio de mínimos cadrados. Vemos que os que teñen mellor aproximación son o descenso por gradiente e o gradiente conxugado cos que coinciden ata en 3 decimais cos obtidos por mínimos cadrados. Cabe destacar que o método de Newton segue converxendo en tan só 2 iteracións mentres que o seguinte en converxer máis rápido é o método de Hooke e Jeeves. Isto débese ao punto inicial que escollemos pois está preto do óptimo da función obxectivo o que orixina esta situación, aunque sexa o máis lento dos catro métodos.

4.3. Regresión lineal regularizada

Nesta sección centrarémonos en resolver o problema LASSO pois para a regresión Ridge, a función obxectivo que se quere minimizar é diferenciable e, polo tanto, poderiamos proceder cos mesmos métodos vistos nas dúas seccións anteriores.

Por outra parte, a librería *glmnet* de R permítenos traballar cos modelos Ridge e LASSO. A función a utilizar é `glmnet` para a cal é necesario unha matriz de variables explicativas, X, e un vector resposta Y. Ademáis, existe un parámetro α para indicar con que modelo desexamos traballar. No caso de querer utilizar Ridge, introduciríamos $\alpha = 0$ mentres que para LASSO sería $\alpha = 1$. Podemos atopar máis información sobre esta librería na páxina web CRAN[1] para programación en R.

Agora ben, para resolver o problema LASSO visto no capítulo 2 implementaremos o método de Hooke e Jeeves e o descenso por subgradiente vistos no capítulo 3, pois

[1] https://cran.r-project.org/web/packages/glmnet/glmnet.pdf

ambos métodos non necesitan que a función obxectivo sexa diferenciable, como é o caso. Centrarémonos só no número de iteracións que tarda cada método en chegar ao óptimo pois o obxectivo desta sección e ver como se comportan os métodos según avanza o número de variables (neste caso explicativas). Para isto, creamos unha matriz X de maneira aleatoria a partir de valores reais entre 0 e 1 e seguidamente estandarizámola (por comodidade) restando a cada columna sua media e dividindo a columna resultante entre o valor da suma ao cadrado das súas compoñentes. Entenderemos por matriz estandarizada aquela cuxas columnas están estandarizadas, é dicir, a suma das súas compoñentes é 0 e a suma das súas compoñentes ao cadrado é 1. O vector resposta será exactamente $Y = X\hat{\beta}_{LASSO} + \varepsilon$, é dicir, definirémolo a partir de $\hat{\beta}_{LASSO} \in \mathbb{R}^p$ sendo p o número de variables explicativas e un erro aleatorio. Entón, dado un número de observacións n e un número de variables p utilizaremos a función `replicate` para crear unha matriz con p columnas de vectores aleatorios de lonxitude n obtidos mediante a función `runif`. A continuación estandarizamos dita matriz como segue:

```
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 o vector Y procedemos como segue:

```
vectorY<-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)
}
```

O seguinte paso é aplicar os métodos de optimización para resolver o problema LASSO. A partir das funcións `matrizdatos` e `vectorY` xeraremos as matrices que definen o problema de regresión lineal con un número de filas fixo para a matriz de datos X, e por ende un tamaño fixo para o vector resposta Y, e un número de columnas para X, é dicir, de variables explicativas, que se irá incrementando. Para esta situación, o valor de λ (valor

de penalización no problema LASSO) e de tolerancia serán constantemente iguales a 1 e a 0,00001 respectivamente. Os métodos executáronse para unha secuencia de valores de p que vai dende 100 ata 800 aumentando de 100 en 100. Para ambos métodos, iniciando dende o mesmo punto, obtivemos os seguintes resultados:

p	Nº iter. Hooke e Jeeves	Nº iter. D. subgradiente
100	543	151
200	2376	249
300	6438	497
400	8997	622
500	10778	952
600	13765	1387
700	17890	1564
800	21800	1673

Táboa 4.4: Número de iteracións según número de variables explicativas

Como era de esperar, o método por subgradiente é moito máis rápido que o de Hooke e Jeeves, debido ao algoritmo deste último. O método do subgradiente é fácil de implementar, sen embargo, determinar o subdiferencial de calquera función non é simple, polo que o método do subgradiente, aínda que é fácil de implementar, presenta complicacións ao momento de encontrar o subgradiente da función en cada iteración.

Anexo I: Código en R empregado

```
#Simulamos datos para unha regresión lineal simple
a<-2
b<-3
n<-100
eps<-rnorm(n,0,0.2)
x<-runif(n)
y<-a+b*x+eps
```

```
# Método de Hooke e Jeeves
# Modelo simple con punto inicial (1,4)
iteraciones=1000
tol=0.001
x0=1;y0=4
p0=c(x0,y0)

f=function(beta0,beta1){
return(sum((y-beta0-beta1*x)^2))
}

p1=x0
p2=y0
objetivo=f(x0,y0)
d1=NA
d2=NA
paso=NA

for (i in 1:iteraciones){
minim1=function(z){
return(f(x0+z,y0))
}

ls1=(optimize(minim1,c(-0.05,0.05),tol=tol,maximum = FALSE))

minim2=function(z){
return(f(x0+ls1$minimum,y0+z))
}

ls2=(optimize(minim2,c(-0.05,0.05),tol=tol,maximum = FALSE))

p0=c(x0+ls1$minimum,y0+ls2$minimum)
direccion_descenso=c(ls1$minimum,ls2$minimum)

p1[i+1]=x0+ls1$minimum
p2[i+1]=y0+ls2$minimum
```

```
objetivo[i+1]=f(x0+ls1$minimum,y0+ls2$minimum)
d1[i]=(ls1$minimum)
d2[i]=(ls2$minimum)
paso[i]=sqrt(direccion_descenso%*%direccion_descenso)

if ((sqrt(sum(c(ls1$minimum,ls2$minimum)^2))/sqrt(sum(c(x0,y0)^2)))<tol) break

minim3=function(z){
return(f(x0+ls1$minimum+z*ls1$minimum,y0+ls2$minimum+z*ls2$minimum))
}

ls3=(optimize(minim3,c(-0.05,0.05),tol=tol,maximum = FALSE))

x0=x0+ls1$minimum+ls3$minimum*ls1$minimum; y0=y0+ls2$minimum+ls3$minimum*ls2$minimum

}

a=cbind(d1,d2)
p=cbind(p1,p2)
total_iteracions=i;i
p0=c(p1[total_iteracions],p2[total_iteracions]);p0

plot(p1,p2,type="p",xlim=c(0,3),ylim=c(2,5))
```

```
# Descenso por gradiente
# Modelo simple con punto inicial (1,4)

iteraciones=1000
tol=0.001
x0=1;y0=4
p0=c(x0,y0)

f=function(beta0,beta1){
return(sum((y-beta0-beta1*x)^2))
}

grad1f=function(beta0,beta1){
return(sum(-2*(y-beta0-beta1*x)))
}

grad2f=function(beta0,beta1){
return(sum(-2*x*(y-beta0-beta1*x)))
}

p1=x0
p2=y0
objetivo=f(x0,y0)
d1=NA
d2=NA
paso=NA

for (i in 1:iteraciones){

if ((sqrt(sum(c(grad1f(x0,y0),grad2f(x0,y0))^2)))<tol) break

direccion_descenso=c(-grad1f(x0,y0),-grad2f(x0,y0))

minim=function(z){
return(f(x0-z*grad1f(x0,y0),y0-z*grad2f(x0,y0)))
}
```

```
ls=(optimize(minim,c(-0.05,0.05),tol=tol,maximum = FALSE))

p0=c(x0-ls$minimum*grad1f(x0,y0),y0-ls$minimum*grad2f(x0,y0))

p1[i+1]=x0-ls$minimum*grad1f(x0,y0)
p2[i+1]=y0-ls$minimum*grad2f(x0,y0)
objetivo[i+1]=f(x0-ls$minimum*grad1f(x0,y0),y0-ls$minimum*grad2f(x0,y0))
d1[i]=(-ls$minimum*grad1f(x0,y0))
d2[i]=(-ls$minimum*grad2f(x0,y0))
paso[i]=sqrt(direccion_descenso%*%direccion_descenso)

x0=x0-ls$minimum*grad1f(x0,y0); y0=y0-ls$minimum*grad2f(x0,y0)
}

a=cbind(d1,d2)
p=cbind(p1,p2)
total_iteracions=i;i
p0=c(p1[total_iteracions],p2[total_iteracions]);p0

plot(p1,p2,type="p",xlim=c(0,3),ylim=c(2,5))
```

```
# Método de Newton
# Modelo simple con punto inicial (1,4)

iteracions=1000
tol=0.001
x0=1;y0=4
p0=c(x0,y0)

f=function(beta0,beta1){
return(sum((y-beta0-beta1*x)^2))
}

grad1f=function(beta0,beta1){
return(sum(-2*(y-beta0-beta1*x)))
}

grad2f=function(beta0,beta1){
return(sum(-2*x*(y-beta0-beta1*x)))
}

hessiana=function(beta0,beta1){
return(matrix(c(2*n,sum(2*x),sum(2*x),sum(2*x^2)),ncol=2,nrow=2))
}

p1=x0
p2=y0
objetivo=f(x0,y0)
d1=NA
d2=NA
paso=NA

for(i in 1:iteracions){

if ((sqrt(sum(c(grad1f(x0,y0),grad2f(x0,y0))^2)))<tol) break

direccion_descenso=(-(solve(hessiana(x0,y0))))%*%c(grad1f(x0,y0),grad2f(x0,y0))
```

```
p1[i+1]=x0+direccion_descenso[1]
p2[i+1]=y0+direccion_descenso[2]
objetivo[i+1]=f(x0+direccion_descenso[1],y0+direccion_descenso[2])
d1[i]=(direccion_descenso[1])
d2[i]=(direccion_descenso[2])
paso[i]=sqrt(sum(direccion_descenso^2))

x0=x0+direccion_descenso[1]
y0=y0+direccion_descenso[2]
}

a=cbind(d1,d2)
p=cbind(p1,p2)
total_iteracions=i;i
sol=c(p1[total_iteracions],p2[total_iteracions]);sol

plot(p1,p2,type="p",xlim=c(0,3),ylim=c(2,5))
```

```
# Gradiente Conxugado
# Modelo simple con punto inicial (1,4)

iteracions=1000
tol=0.001
x0=1;y0=4
p0=c(x0,y0)

f=function(beta0,beta1){
return(sum((y-beta0-beta1*x)^2))
}
grad1f=function(beta0,beta1){
return(sum(-2*(y-beta0-beta1*x)))
}
grad2f=function(beta0,beta1){
return(sum(-2*x*(y-beta0-beta1*x)))
}
p1=x0
p2=y0
objetivo=f(x0,y0)
d1=NA
d2=NA
paso=NA

if ((sqrt(sum(c(grad1f(x0,y0),grad2f(x0,y0))^2)))<tol) break
alpha=0
direccion_descenso=c(-grad1f(x0,y0),-grad2f(x0,y0))

minim=function(z){
return(f(x0-z*grad1f(x0,y0),y0-z*grad2f(x0,y0)))
}
ls=(optimize(minim,c(-0.05,0.05),tol=tol,maximum = FALSE))
p0=c(x0-ls$minimum*grad1f(x0,y0),y0-ls$minimum*grad2f(x0,y0))
x0new=x0-ls$minimum*grad1f(x0,y0)
y0new=y0-ls$minimum*grad2f(x0,y0)
```

```
p1[2]=x0-ls$minimum*grad1f(x0,y0)
p2[2]=y0-ls$minimum*grad2f(x0,y0)
objetivo[2]=f(x0-ls$minimum*grad1f(x0,y0),y0-ls$minimum*grad2f(x0,y0))
d1[1]=(-ls$minimum*grad1f(x0,y0))
d2[1]=(-ls$minimum*grad2f(x0,y0))
paso[1]=sqrt(direccion_descenso%*%direccion_descenso)

for (i in 2:iteraciones){

if ((sqrt(sum(c(grad1f(x0new,y0new),grad2f(x0new,y0new))^2)))<tol) break
alpha=(c(grad1f(x0new,y0new),grad2f(x0new,y0new))*(c(grad1f(x0new,y0new),
grad2f(x0new,y0new))-c(grad1f(x0,y0),grad2f(x0,y0))))/sum(c(grad1f(x0,y0),
grad2f(x0,y0))^2)
direccion_descenso=c(-grad1f(x0new,y0new),-grad2f(x0new,y0new))+alpha*direccion_descenso

minim=function(z){
return(f(x0new+z*direccion_descenso[1],y0new+z*direccion_descenso[2]))
}
ls=(optimize(minim,c(-0.05,0.05),tol=tol,maximum = FALSE))
x0=x0new
y0=y0new
x0new=x0new+ls$minimum*direccion_descenso[1]
y0new=y0new+ls$minimum*direccion_descenso[2]
p1[i+1]=x0new
p2[i+1]=y0new
objetivo[i+1]=f(x0new,y0new)
d1[i]=(direccion_descenso[1])
d2[i]=(direccion_descenso[2])
paso[i]=sqrt(direccion_descenso%*%direccion_descenso)
}
a=cbind(d1,d2)
p=cbind(p1,p2)
total_iteracions=i;i
p0=c(p1[total_iteracions],p2[total_iteracions]);p0

plot(p1,p2,type="p",xlim=c(0,3),ylim=c(2,5))
```

Bibliografía

[1] Bazaraa, M., Sherali, H., Shetty, C., *Nonlinear programing: Theory and algorithms*, 2nd ed., New York, 1993.

[2] Boyd, S., Vandenberghe, L., *Convex Optimization*, Cambridge University Press, 2004.

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

[4] Hastie, T., Tibshirani, R. and Friedman, J., *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed., Springer Series in Statistics, Springer, 2009.

[5] James, G., Witten, D., Hastie, T. and Tibshirani, R., *An Introduction to Statistical Learning: with Applications in R*, 1st ed., Springer Texts in Statistics, Springer Science+Business Media, New York, 2013.

[6] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, 2017.

[7] Shor, N., *Minimization Methods for Non-differenciable function*, NaukovaDumka, Kiev, 1979.

[8] Larson, R., Edwards, B., *Cálculo 1. De una variable*, McGraw-Hill Interamericana Editores, México, 2010.

[9] Trahan, D., *The Mixed Partial Derivates and the Double Derivate*, American Mathematical Monthly, 1969.