

Modelos lineales: regresión

Curso de aprendizaje automático para
el INE

IGMAT

Alberto Torres Barrán

2019-04-10

Modelos lineales

Variables provienen de múltiples fuentes:

- variables cuantitativas
- transformaciones (logaritmo, raíz cuadrada, etc.)
- expansiones de base, $x_2 = x_2^2$
- variables *dummy*
- interacciones entre variables, $x_3 = x_1 \times x_2$

Pero siempre: modelo lineal en los parámetros

Regresión lineal y logística

La salida y es continua, $y \in \mathbb{R}$

- Regresión lineal (*MSE* o *RSS*)

$$\min_w ||y - \mathbf{X}w||_2^2$$

La salida y es discreta, $y \in \{0, 1\}$

- Regresión logística (*log-loss*)

$$\min_w -(y^T \log[\sigma(\mathbf{X}w)] + (1 - y)^T \log[1 - \sigma(\mathbf{X}w)])$$

Generalized linear models (GLM)

- Generalización de la regresión lineal que permite distribuciones de errores distintas de la distribución normal.
- Componentes:
 - Distribución de Y_i con media μ_i
 - Predictor lineal,

$$g(\mu_i) = w^T x_i$$

donde $g(\cdot)$ es la función de media

- La función de media proporciona la relación entre la media de la distribución y el predictor lineal
- El inverso de la función de media, $g^{-1}(\cdot)$ se conoce con el nombre de **función de enlace**

Ejemplo: distribución binomial

- La regresión logística es un caso particular de GLM donde la distribución de Y es la binomial
- La función de media es la logística,

$$\mu = g^{-1}(w^T x_i) = \frac{1}{1 + \exp(-w^T x_i)}$$

- La función de enlace es la inversa de la anterior,

$$w^T x_i = g(\mu) = \ln\left(\frac{\mu}{1 - \mu}\right)$$

- Para cada distribución, hay una función de enlace "canónica" que es la que se usa habitualmente

Ejemplo: distribución de Poisson

- Esta distribución está indicada cuando queremos modelizar una variable de salida entera y no real (por ej. conteos)
- Función de media

$$\mu = \exp(w^T x_i)$$

- Función de enlace

$$w^T x_i = \ln(\mu)$$

- Otras distribuciones posibles son la Gamma, Exponencial, Multinomial, etc.

GLMs en R

- La función para ajustar modelos lineales generalizados es `glm()`
- Tiene los mismos argumentos principales que `lm()`, pero además tenemos que especificar la distribución de la variables dependiente con el parámetro `family`
- Por defecto se usa la función de enlace "canónica", pero esto se puede modificar (ver ayuda)
- Implementa el algoritmo IRLS (Newton-Raphson), que se puede generalizar para cualquier GLM donde la distribución pertenece a la familia exponencial

Ejemplo: regresión logística

```
library(MASS)  
fit <- glm(type ~ ., data=Pima.tr, family=binomial)
```

Problemas de mínimos cuadrados

1. Calidad de predicción:

- poco sesgo pero potencialmente mucha varianza
- podemos mejorar las predicciones reduciendo el valor de algunos coeficientes
- aumenta ligeramente el sesgo pero disminuye mucho la varianza

2. Interpretación:

- el valor de los coeficientes nos da una idea de las variables mas relevantes
- nos gustaria encontrar un subconjunto de los mejores

Regularización

- Regresión *ridge* (MSE + regularización l_2):

$$\min_w ||y - \mathbf{X}w||_2^2 + ||w||_2^2$$

- ¿Regresión logística ridge?
- ¿Otras funciones de regularización?

Métodos de seleccion

Regresión *best subset*

- Mantenemos solo un subconjunto de las variables y eliminamos el resto del modelo
- Para $k \in \{0, 1, 2, \dots, d\}$ se resuelve

$$\min_w ||y - \mathbf{X}w||_2^2 \quad \text{s.t.} \quad ||w||_0 \leq k$$

donde $||w||_0 = \sum_{i=1}^d \mathbb{I}(w_i \neq 0)$

- $\mathbb{I}(\cdot)$ es la función indicatriz (cuenta el número de elementos distintos de 0)
- La restricción hace que el problema sea NP-completo,

$$C_{d,k} = \binom{d}{k} = \frac{d!}{k!(d-k)!}$$

- Algoritmos clásicos pueden resolver $d \approx 30$
- Avances recientes, (Bertsimas et al., 2015): $d \in [100, 1000]$

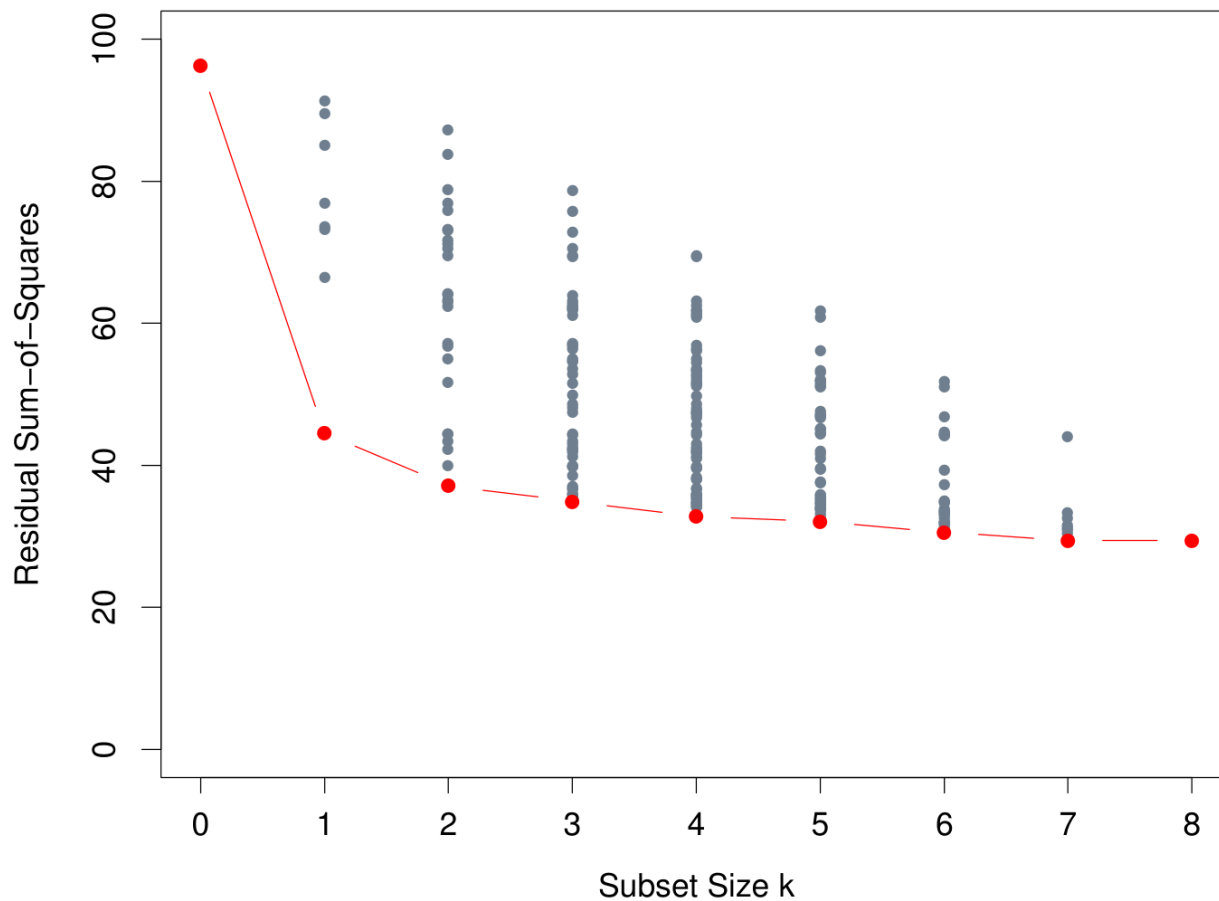


FIGURE 3.5. All possible subset models for the prostate cancer example. At each subset size is shown the residual sum-of-squares for each model of that size.

Regresión *stepwise*

- *Forward-Stepwise*:
 1. Añadir al modelo la variable que proporciona mejor ajuste
 2. Repetir hasta añadir k variables
- *Backward-Stepwise*
 1. Empezar con las d variables
 2. Eliminar iterativamente la menos relevante para el ajuste
- Algoritmos avariciosos
- No buscan entre todas las posibles combinaciones de subconjuntos de tamaño k
- En cada paso solo se ajustan $d - k$ modelos

Best subset y stepwise en R

Best subsets:

- Paquete `leaps`, función `regsubsets()`
- También regresión forward y backward stepwise

Stepwise:

- Función `step()`
- Procedimiento híbrido: se procede como *forward* pero en cada paso está la opción de eliminar alguna variable añadida previamente
- Añade o elimina variables en grupos (por ej. si son variables *dummy*)
- Selecciona automáticamente el valor óptimo de k

Selección de k

- Secuencia de modelos indexada por k (igual que *best subset*)
- Elegir k como el que minimiza el error de validación cruzada
- Validación cruzada: estimación del error de generalización o error *extra-sample*
- Error de entrenamiento es demasiado optimista (error *in-sample*)
- Alternativa: cuantificar el "optimismo" y minimizarlo (AIC, BIC y derivados)
- Más detalles: ESL secciones 7.4 en adelante

Métodos de reducción

Lasso: motivación

- Métodos de selección:
 - modelos interpretables
 - proceso discreto, las variables están incluidas o no
- Regresión *ridge*:
 - proceso continuo, todos los coeficientes se reducen
 - rara vez son exactamente 0, modelos no interpretables
- Lasso es una técnica intermedia:
 - reduce algunos coeficientes
 - pone el resto a 0

Lasso: formulación

- Problema optimización:

$$\min_w ||y - \mathbf{X}w||_2^2 \quad \text{s.t.} \quad ||w||_1 \leq t$$

- Equivalente:

$$\min_w ||y - \mathbf{X}w||_2^2 + \lambda ||w||_1$$

- λ o t son hiper-parámetros
 - $\uparrow \lambda$ o $\downarrow t$, se reducen los coeficientes (más regularización)
 - $\downarrow \lambda$ o $\uparrow t$, aumentan los coeficientes (menos regularización)
- t suficientemente pequeño (o λ suficientemente grande), algunos coeficientes = 0

$$\beta^* = \arg \min \frac{1}{2} \|\mathbf{y} - X\beta\|^2 \text{ s.t. } |\beta|_1 \leq t$$

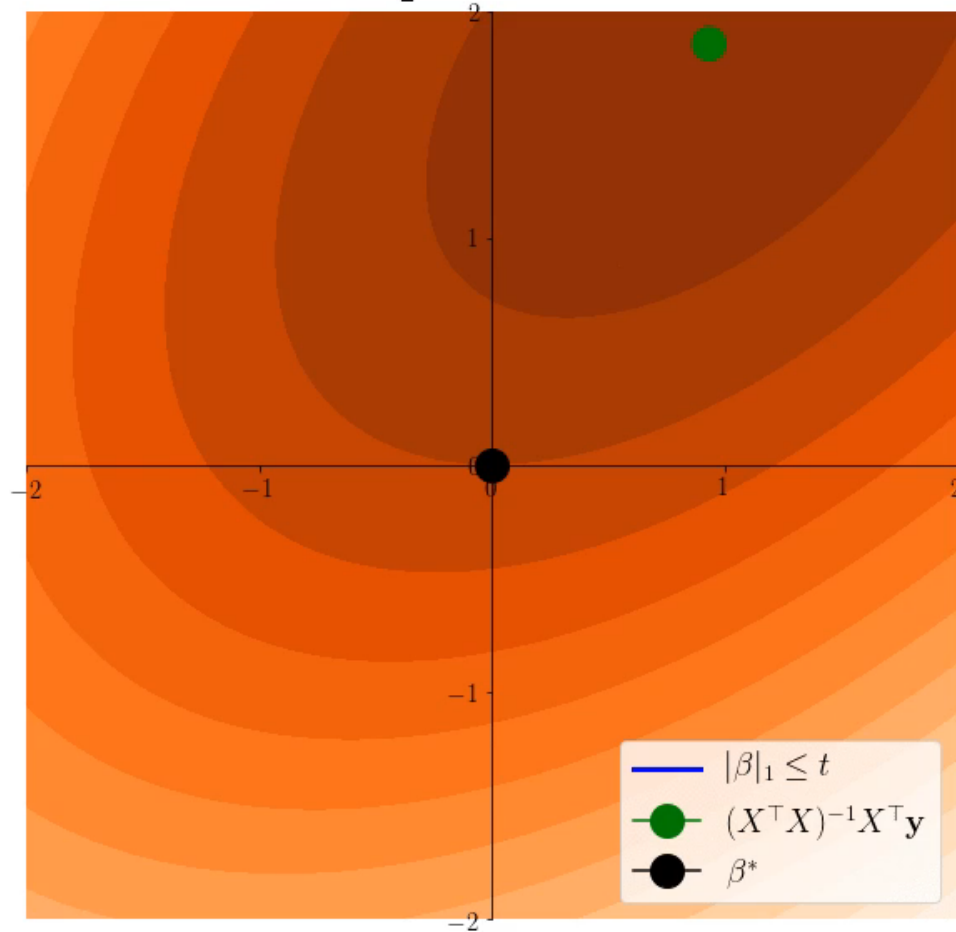
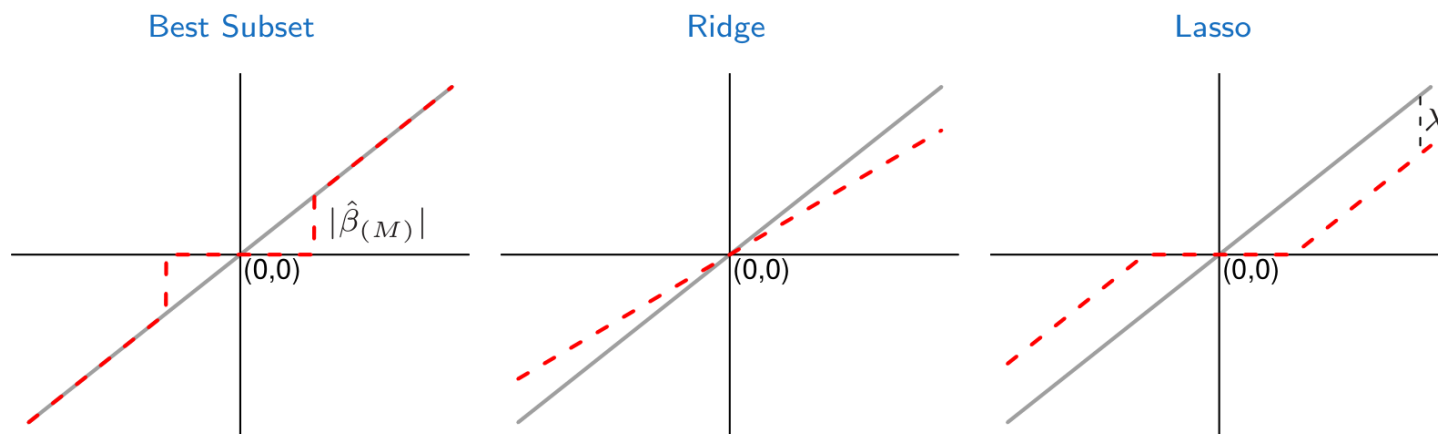


TABLE 3.4. Estimators of β_j in the case of orthonormal columns of \mathbf{X} . M and λ are constants chosen by the corresponding techniques; sign denotes the sign of its argument (± 1), and x_+ denotes “positive part” of x . Below the table, estimators are shown by broken red lines. The 45° line in gray shows the unrestricted estimate for reference.

Estimator	Formula
Best subset (size M)	$\hat{\beta}_j \cdot I(\hat{\beta}_j \geq \hat{\beta}_{(M)})$
Ridge	$\hat{\beta}_j / (1 + \lambda)$
Lasso	$\text{sign}(\hat{\beta}_j)(\hat{\beta}_j - \lambda)_+$



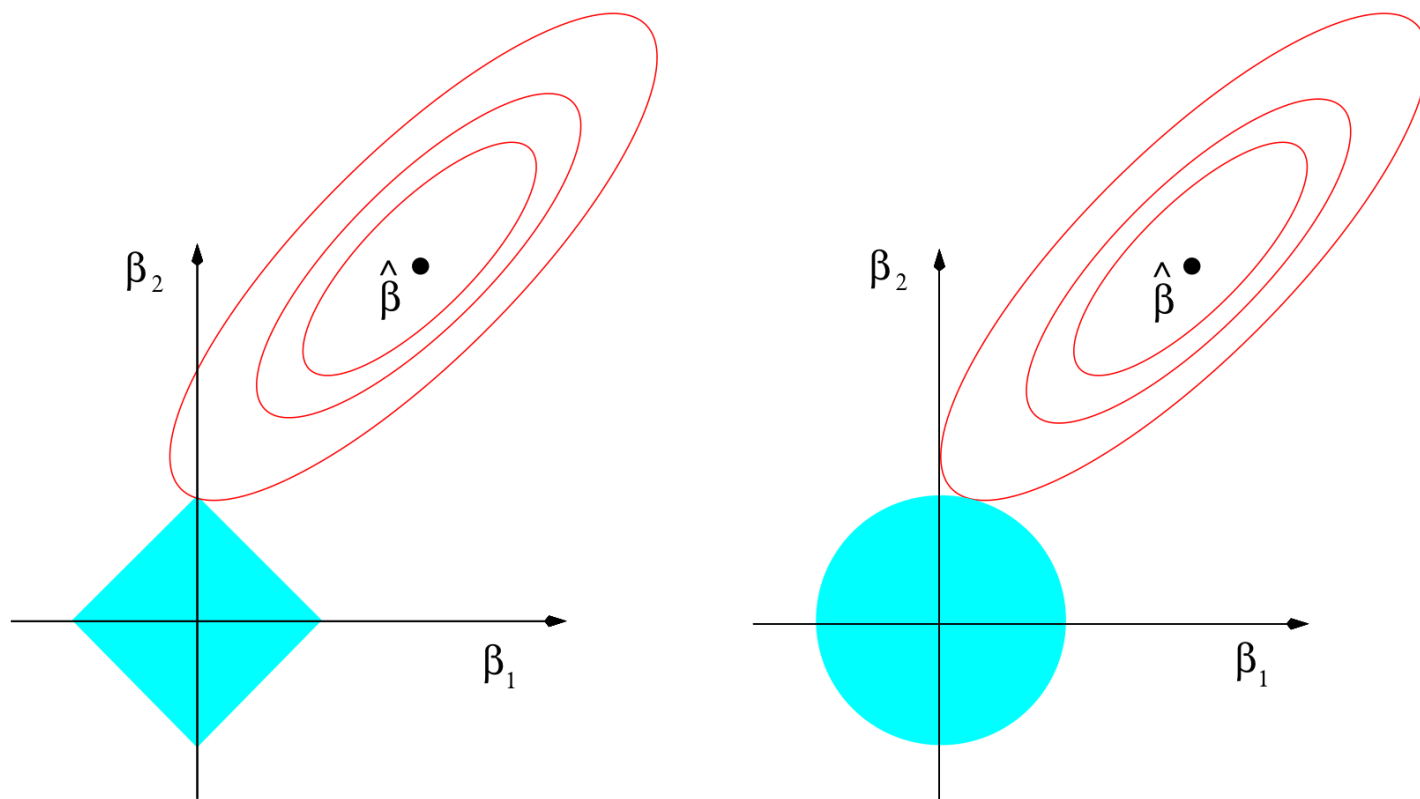


FIGURE 3.11. *Estimation picture for the lasso (left) and ridge regression (right). Shown are contours of the error and constraint functions. The solid blue areas are the constraint regions $|\beta_1| + |\beta_2| \leq t$ and $\beta_1^2 + \beta_2^2 \leq t^2$, respectively, while the red ellipses are the contours of the least squares error function.*

Lasso, best subset y forward stepwise

- Comparación Lasso, *best subset* y *forward stepwise* (Hastie et al., 2017)
- Experimentos en [Github, best-subset](#)
- *Relaxed Lasso*: ajustar otro modelo sobre las variables que selecciona el Lasso

Lasso: limitaciones

1. Si $d > n$, como mucho n coeficientes son $\neq 0$
 - Limitación desde el punto de vista de selección de variables
2. Variables con correlación alta \implies Lasso selecciona una "aleatoriamente"
3. Si $n > d$ y hay variables con correlación alta \implies error de Ridge $<$ error de Lasso

Otras penalizaciones

- Podemos generalizar Lasso, Ridge y Best subset como

$$\min_w ||y - \mathbf{X}w||_2^2 + \lambda ||w||_p^p$$

donde,

$$||w||_p^p = \sum_{i=1}^d |x_i|^p$$

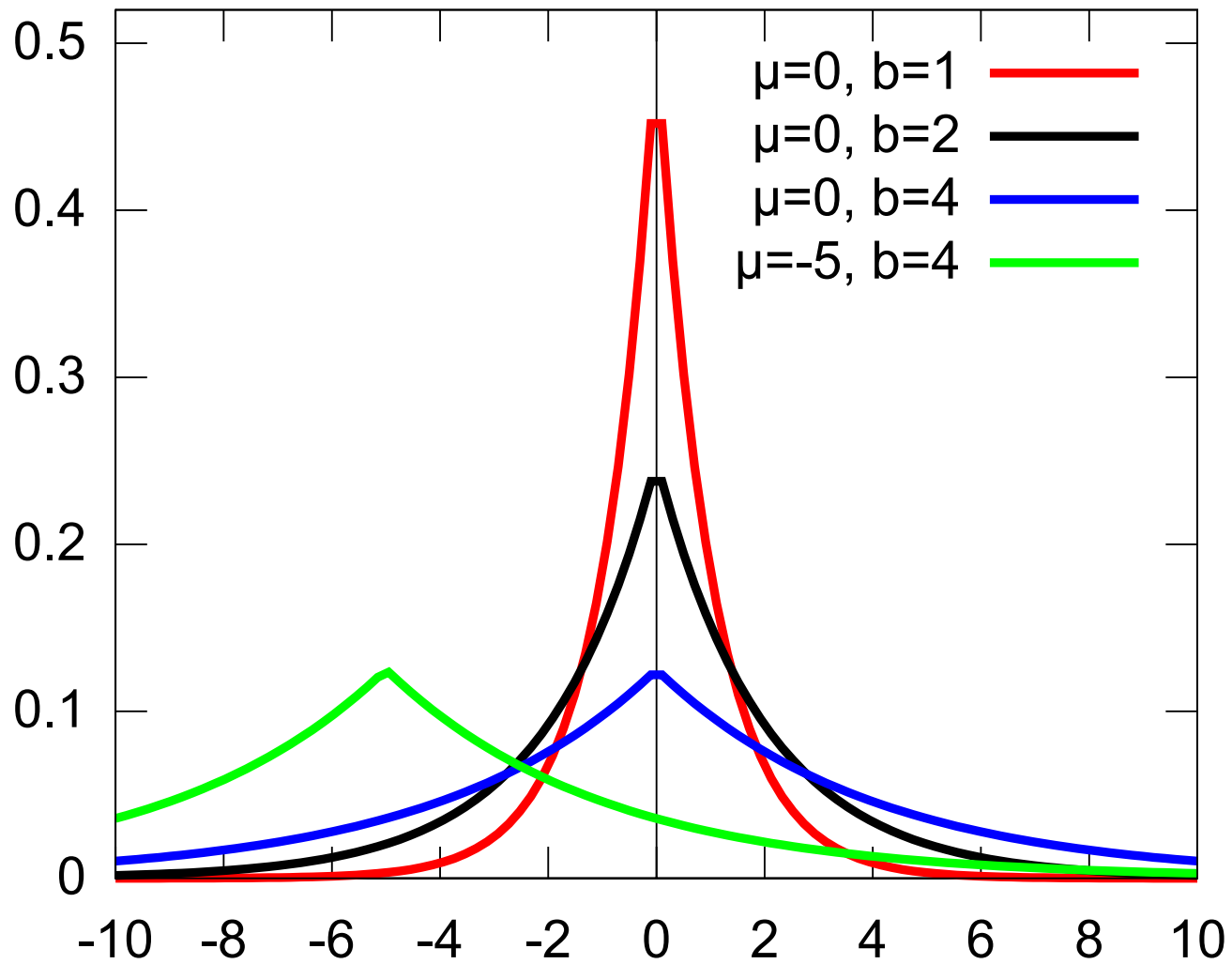
- $0 \leq p < 1$, no convexas (**NP-completo!**)
- $p = 1$, convexa y no diferenciable
- $p > 1$, convexas y diferenciables

Interpretación Bayesiana

- Regularización = distribución a priori de los parámetros w
- Ridge regresión: distribución Normal
- Lasso: distribución de Laplace, $\tau = 1/\lambda$

$$f(w) = \frac{1}{2\tau} \exp \left(- \frac{|w|}{\tau} \right)$$

- Estimadores: máximo de la distribución a posteriori (MAP)
 - Ridge: coincide con la media
 - Lasso y Best subset: moda



Wikipedia, "Laplace distribution"

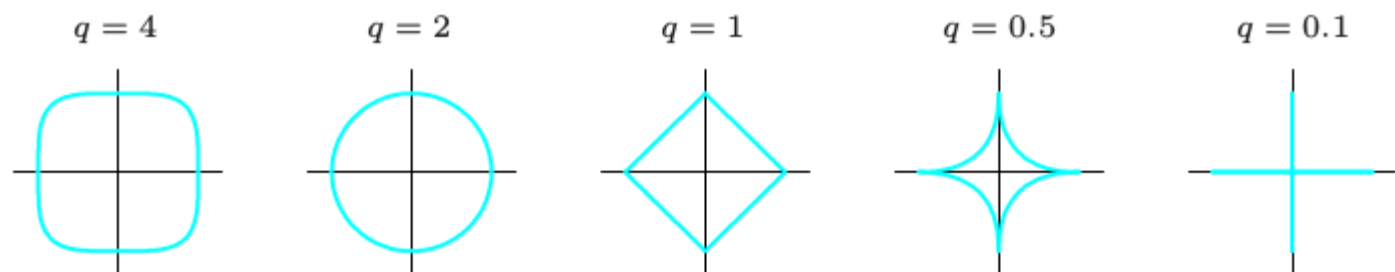


FIGURE 3.12. Contours of constant value of $\sum_j |\beta_j|^q$ for given values of q .

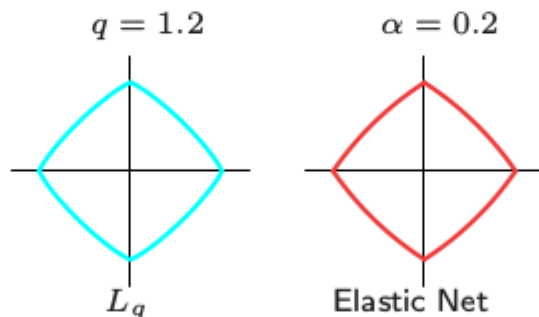


FIGURE 3.13. Contours of constant value of $\sum_j |\beta_j|^q$ for $q = 1.2$ (left plot), and the elastic-net penalty $\sum_j (\alpha \beta_j^2 + (1-\alpha)|\beta_j|)$ for $\alpha = 0.2$ (right plot). Although visually very similar, the elastic-net has sharp (non-differentiable) corners, while the $q = 1.2$ penalty does not.

Elastic Net

- Combina regularización Lasso y Ridge:

$$\min_w ||y - \mathbf{X}w||_2^2 + \lambda_1 ||w||_1 + \lambda_2 ||w||_2^2$$

- Otra parametrización con $\alpha \in (0, 1)$:

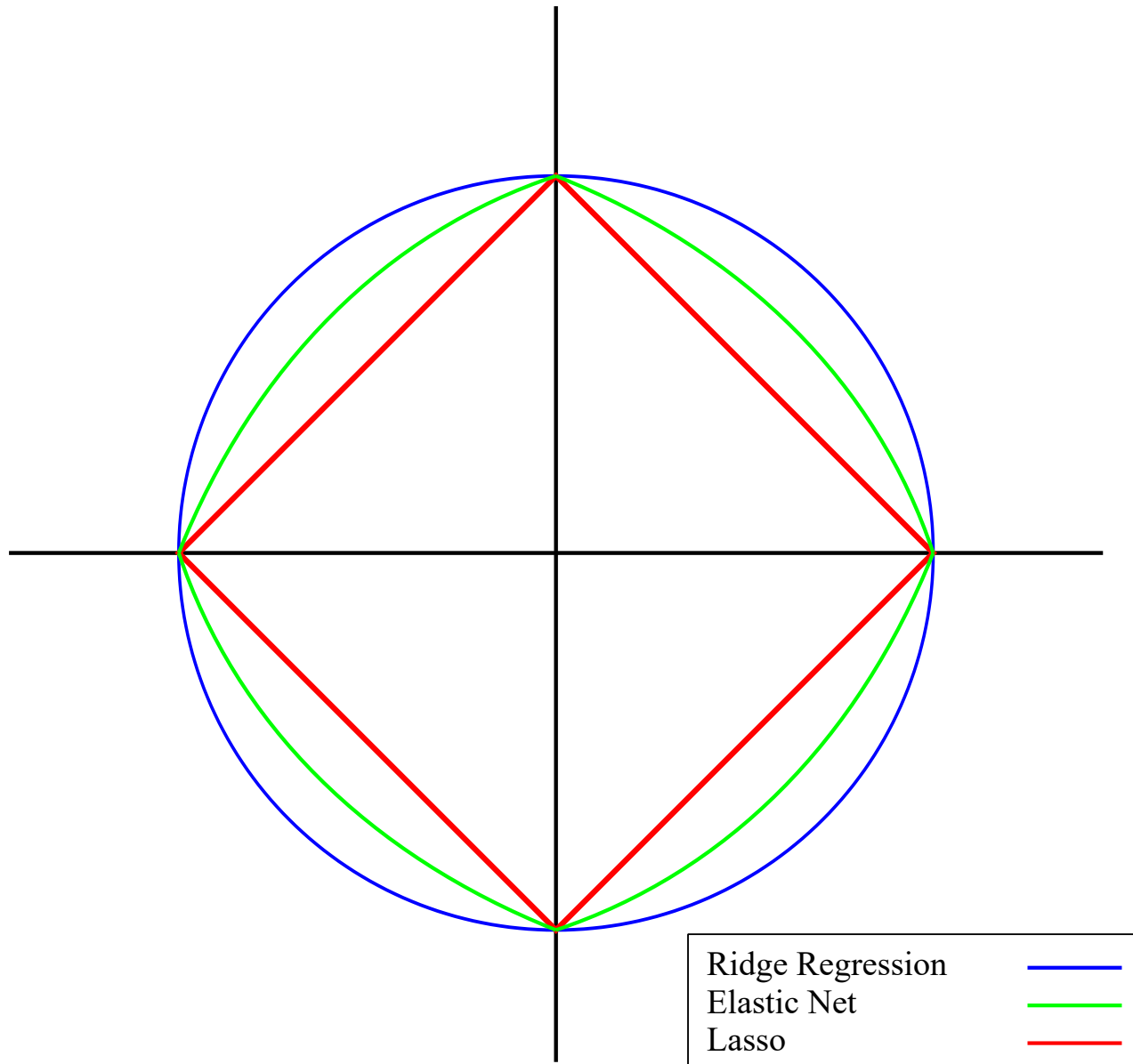
$$\min_w ||y - \mathbf{X}w||_2^2 + \lambda(\alpha ||w||_1 + (1 - \alpha) ||w||_2^2)$$

- Equivalentes con

- $\alpha = \lambda_1 / (\lambda_1 + \lambda_2)$

- $\lambda = \lambda_1 + \lambda_2$

- Selecciona variables y reduce el resto de coeficientes



Notación

- Lasso (Elastic Net) suelen hacer referencia a: minimizar MSE + norma l_1 (+ norma l_2)
- MSE puede reemplazarse por otras funciones de pérdida
- Por ej. cualquier GLM
- En esos casos hablamos de regresión logística, Poisson, Gamma + regularización Ridge/Lasso/Elastic Net

Elastic Net en R

- Paquete `glmnet`
- Implementa descenso coordinado cíclico (detalles más adelante)
- Resuelve GLMs con regularización l_1 (Lasso), l_2 (Ridge) o ambas (ElasticNet)
- Sin interfaz de fórmulas
 - Hay que crear la matriz \mathbf{X} "a mano"
- Elige automáticamente el valor óptimo de λ (pero no α !)
- Puede ser interesante la extensión `glmnetUtils`

Métodos de reducción

Principal Component Regression

- Calcular las m componentes principales, $\mathbf{V} \in \mathbb{R}^{d \times m}$
- Crear combinaciones lineales de las variables originales: $\mathbf{Z} = \mathbf{XV} \in \mathbb{R}^{n \times m}$
- Ajustar una regresión lineal, $y = \mathbf{Z}\theta$,

$$\theta^* = (\mathbf{Z}^T \mathbf{Z})^{-1} \mathbf{Z}^T y$$

- \mathbf{Z} es ortogonal, por lo que los θ_m son independientes

$$\theta_m = \frac{z_m^T y}{z_m^T z_m}$$

- Coeficientes sobre los datos originales \mathbf{X} :

$$y = \mathbf{Z}\theta = \mathbf{XV}\theta = \mathbf{X}w \Rightarrow w = \mathbf{V}\theta$$

PCR, OLS y Ridge Regression

- Componentes principales dependientes de la escala \Rightarrow estandarizar
- Si $m = d$, se obtienen el estimador de mínimos cuadrados (OLS)
- Si $m < d$, se obtiene una versión reducida de la regresión
- Ridge Regression:
 - reduce los coeficientes de las componentes principales
 - reduce más cuanto más grande sea el autovalor
- PCR descarta las $d - m$ componentes principales con menor autovalor
- m se puede elegir por validación cruzada

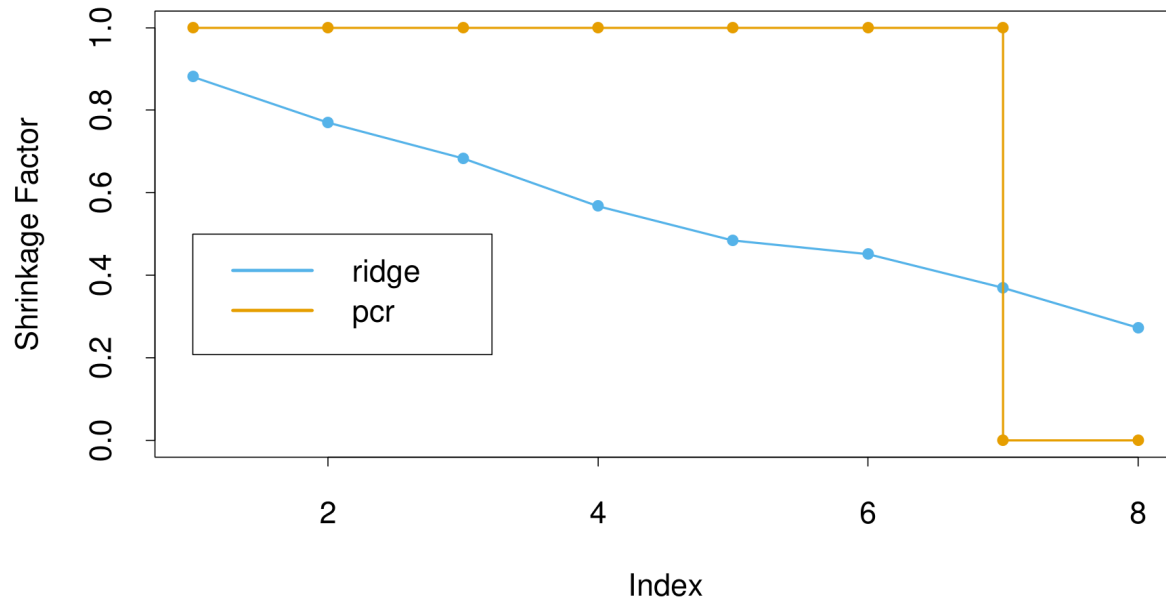


FIGURE 3.17. Ridge regression shrinks the regression coefficients of the principal components, using shrinkage factors $d_j^2/(d_j^2 + \lambda)$ as in (3.47). Principal component regression truncates them. Shown are the shrinkage and truncation patterns corresponding to Figure 3.7, as a function of the principal component index.

PCR, Lasso y Best subset

- Lasso y Best subset obtienen modelos interpretables
- PCR reduce las variables, pero son combinaciones lineales de las originales
- Computacionalmente:
 - Best subset es factible para $d \approx 100$
 - Lasso y PCR tienen aprox. el mismo coste
- PCR puede ser útil para situaciones con muchas variables altamente correladas

Partial Least Squares

- Igual que con PCR, es importante estandarizar las x_j
- Algoritmo (simplificado):

1. Calcular $\phi_m = x_j^T y$ para cada $j = 1, \dots, d$

2. Calcular $z_m = \sum_j \phi_m x_j$

3. Resolver regresión lineal $y = z_m \theta_m$

4. Actualizar salida, $y^{(m)} = y^{(m-1)} + \theta_m^* z_m$

5. Ortogonalizar x_j con respecto a z_m

6. Repetir hasta un $m < d$

PLS vs PCR

- PCR:
 - no supervisado (solo usa \mathbf{X} para calcular las componentes principales)
 - elige direcciones z_m que maximizan la varianza
- PLS:
 - crea combinaciones lineales de las variables originales, pero de manera supervisada (usando el valor de y)
 - si $m = d$ obtenemos el estimador de mínimos cuadrados
 - produce una series de direcciones ortogonales z_1, z_2, \dots, z_m
 - elige direcciones que maximizan la varianza y tienen mucha correlación con la salida y

PLS y PCR en R

- Paquete `pls`
- Funciones `pcr()` y `pls()`
- Ambas eligen el valor óptimo de k usando validación cruzada

Descenso coordinado

Motivación

- Lasso es un problema
 1. cuadrático
 2. convexo
 3. sin restricciones
- Pero
 1. No diferenciable en 0
 2. No es fuertemente convexo \implies peor tasa de convergencia algoritmos estándar

Descenso coordinado

- Resuelve de forma eficiente el problema de optimización de los GLM con:
 - regularización l_1
 - regularización $l_1 + l_2$
- Esquema básico:
 1. Seleccionar una coordenada $j \in \{1, \dots, d\}$
 2. Fijar el valor del resto
 3. Optimizar con respecto a la coordenada j
 - solución analítica!
 4. Repetir para el resto de las coordenadas varias veces

Descenso coordinado: Lasso

- Función que minimiza Lasso, $f(w) = ||y - \mathbf{X}w||_2^2 + \lambda ||w||_1$
- Fijamos $w_k = \tilde{w}_k$ para $k \neq j$
- Aislamos w_j ,

$$f(\tilde{w}, w_j) = \frac{1}{2} \sum_{i=1}^n (y_i - \sum_{k \neq j} x_{ik} \tilde{w}_k - x_{ij} w_j)^2 + \lambda \sum_{k \neq j} |\tilde{w}_k| + \lambda |w_j|$$

- Derivada, si $w_j \neq 0$:

$$\begin{aligned} \frac{\partial f}{\partial w_j} &= - \sum_{i=1}^n x_{ij} (y_i - \sum_{k \neq j} x_{ik} \tilde{w}_k - x_{ij} w_j) + \lambda \text{sign}(w_j) = \\ &= - \underbrace{\sum_{i=1}^n x_{ij} (y_i - \sum_{k \neq j} x_{ik} \tilde{w}_k)}_{a_j} + \underbrace{\sum_{i=1}^n x_{ij}^2 w_j}_{b_j} + \lambda \text{sign}(w_j) \end{aligned}$$

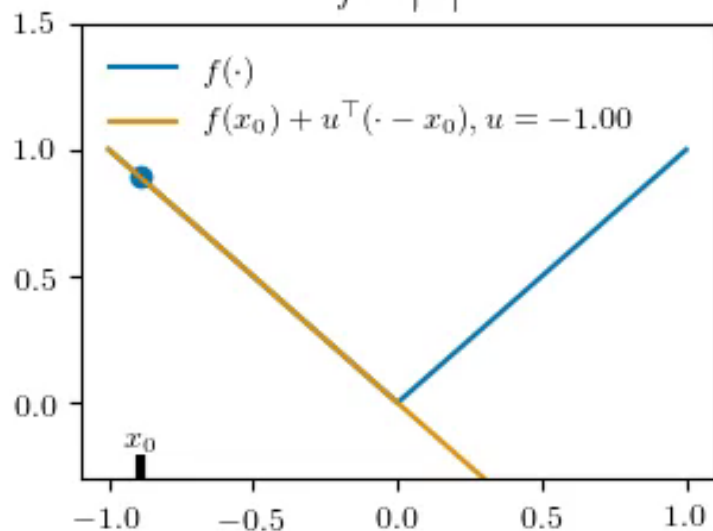
- Si estandarizamos las variables, $b_j = 1$

$$\partial f(x) = \{u : \forall y, f(y) \geq f(x) + u^\top (y - x)\}$$

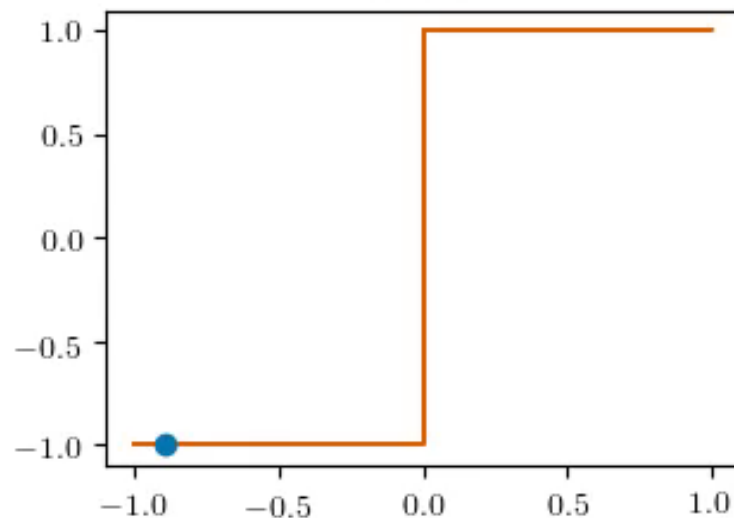
Fermat's rule: $x^* = \min f(x) \Leftrightarrow 0 \in \partial f(x^*)$

f cvx differentiable: $\partial f(x) = \{\nabla f(x)\}$, Fermat: $x^* = \min f(x) \Leftrightarrow \nabla f(x^*) = 0$

$$f = |\cdot|$$



$$\partial f(x_0)$$



Mathurin Massias, Twitter

Solución: operador *soft-thresholding*

1. Si $w_j < 0$, $-a_j + w_j + \lambda = 0 \Rightarrow w_j = a_j + \lambda$ para $a_j < -\lambda$

2. Si $w_j > 0$, $-a_j + w_j - \lambda = 0 \Rightarrow w_j = a_j - \lambda$ para $a_j > \lambda$

3. Si $w_j = 0$, $0 \in [-a_j - \lambda, -a_j + \lambda] \Rightarrow -\lambda \leq a_j \leq \lambda$

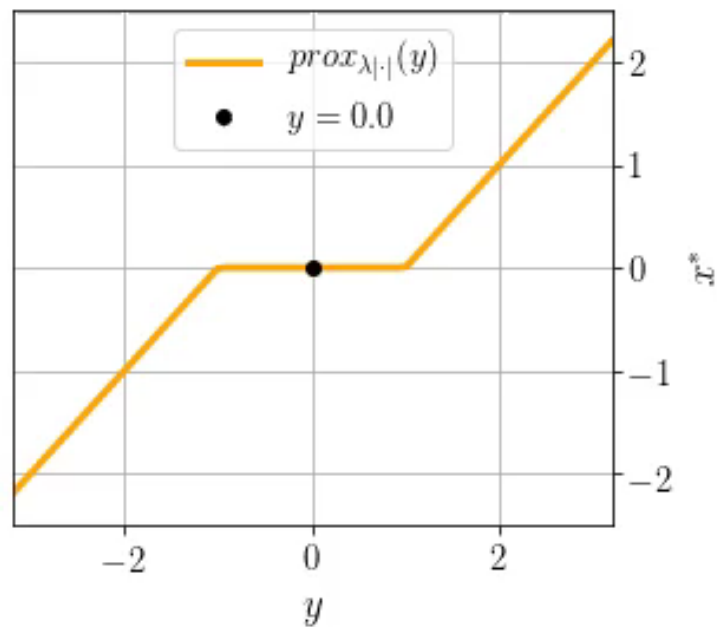
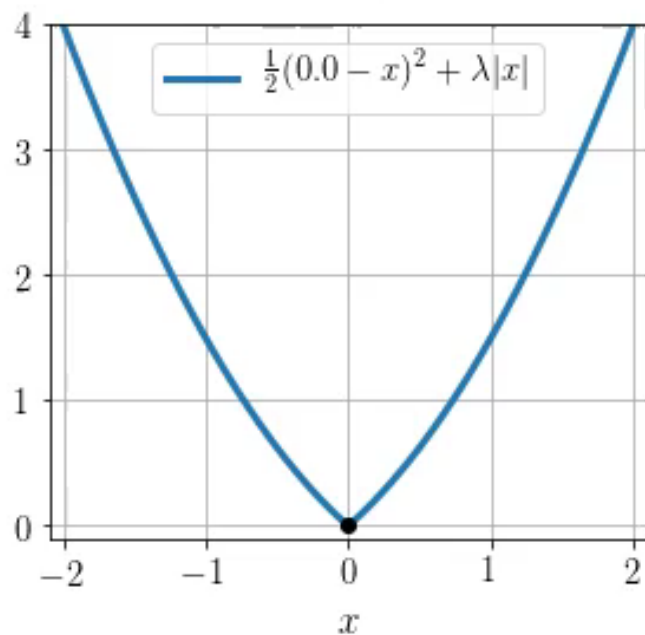
- Operador *soft-thresholding*,

$$S_\gamma(z) = \text{sign}(z) \max(|z| - \gamma, 0)$$

- Actualización:

$$\tilde{w}_j = S_\lambda(a_j) = \left(\sum_{i=1}^n x_{ij} \left(y_i - \sum_{k \neq j} x_{ik} \tilde{w}_k \right) \right)$$

$$\text{prox}_{\lambda|\cdot|}(y) = \arg \min_x \frac{1}{2}(y - x)^2 + \lambda|x|$$



Pierre Ablin, Twitter

Descenso coordinado: implementación

- *Regularization path*: resolver el problema para valores decrecientes de λ
 1. Empezamos por el valor más pequeño de λ para el cual $w^* = 0$, λ_{\max}
 2. Terminamos con λ_{\min} tal que $\lambda_{\min}/\lambda_{\max} = \epsilon$
 3. Creamos una rejilla de M valores en escala logarítmica
- *Warm-starts*: valor inicial de los pesos w es el valor anterior de $w^*(\lambda)$
- Reducir el coste computacional:
 1. Calcular y almacenar $x_j^T y$ (cache)
 2. Aprovechar dispersión: multiplicar solo elementos distintos de 0
 3. Descartar coeficientes 0 antes de tiempo

Variantes del Lasso

Variantes del Lasso

Múltiples variantes:

- Group Lasso
- Fused Lasso
- Generalized Lasso
- Relaxed Lasso
- ...

Ejemplo: Group Lasso

- Variables tienen J grupos predefinidos
- Regularización: norma $l_{2,1}$, $\|w\|_{2,1} = \sum_{j=1}^J \sqrt{\|w_j\|_2}$
- Coeficientes de grupo j son o bien todos 0 o todos $\neq 0$

FISTA

- Descenso por gradiente proyectado:
 1. Paso de descenso por gradiente
 2. Proyectamos a la región de las restricciones usando operador proximal
- En general, el operador proximal es otro problema de optimización:

$$\text{prox}_g(v) = \arg \min_z \left(g(z) + \frac{1}{2} \|z - v\|_2^2 \right)$$

- En ocasiones tiene solución analítica:
 - Lasso: $\text{prox}_{\lambda \|\cdot\|_1}(v) = S_\lambda(v)$
- Existen "trucos" para acelerar la convergencia (Nesterov)

Referencias

1. Tibshirani (1996). **Regression Shrinkage and Selection via the Lasso**
2. Zou, Hastie (2004). **Regularization and variable selection via the elastic net**
3. Beck, Teboulle (2008). **A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems**
4. Friedman, Hastie, Tibshirani (2009). <https://web.stanford.edu/~hastie/Papers/glmnet.pdf>