Regresión Penalizada: Ridge y Lasso

Clase 4: De James-Stein a Machine Learning
2025

Hoja de Ruta

Dónde hemos estado:

- Clase 3: James-Stein y shrinkage empírico

Hoy: El panorama de regresión penalizada

1. Repaso: Ridge como James-Stein para regresión
2. Lasso: Penalización L1 y selección de variables
3. Comparación: L1 vs L2 y sus propiedades
4. Cross-validation: Eligiendo el parámetro de penalización
5. Aplicación: Regresión de salarios con muchas covariables

Próximo: Métodos no-lineales y flexibilidad

El Problema Fundamental

Queremos estimar β en:

$$y = X\beta + \varepsilon$$

Cuando p es grande (muchas variables):

- OLS tiene alta varianza
- Puede sobreajustar (overfitting)
- Predicciones inestables

El Problema Fundamental

Queremos estimar β en:

$$y = X\beta + \varepsilon$$

Cuando p es grande (muchas variables):

- OLS tiene alta varianza
- Puede sobreajustar (overfitting)
- Predicciones inestables

La solución: Regularización

Introducir sesgo para reducir varianza (trade-off bias-variance)

Recordatorio: El Trade-off Bias-Variance

Para cualquier estimador $\hat{f}(x)$, el error de predicción esperado se descompone:

$$E[(y - \hat{f}(x))^2] = \underbrace{\text{Bias}^2[\hat{f}(x)]}_{\text{Aproximación}} + \underbrace{\text{Var}[\hat{f}(x)]}_{\text{Estimación}} + \underbrace{\sigma^2}_{\text{Irreducible}}$$

Recordatorio: El Trade-off Bias-Variance

Para cualquier estimador $\hat{f}(x)$, el error de predicción esperado se descompone:

$$E[(y - \hat{f}(x))^2] = \underbrace{\text{Bias}^2[\hat{f}(x)]}_{\text{Aproximación}} + \underbrace{\text{Var}[\hat{f}(x)]}_{\text{Estimación}} + \underbrace{\sigma^2}_{\text{Irreducible}}$$

OLS:

- Insesgado: Bias $= 0$
- Alta varianza cuando p grande o X mal condicionada

Recordatorio: El Trade-off Bias-Variance

Para cualquier estimador $\hat{f}(x)$, el error de predicción esperado se descompone:

$$E[(y - \hat{f}(x))^2] = \underbrace{\text{Bias}^2[\hat{f}(x)]}_{\text{Aproximación}} + \underbrace{\text{Var}[\hat{f}(x)]}_{\text{Estimación}} + \underbrace{\sigma^2}_{\text{Irreducible}}$$

OLS:

- Insesgado: Bias $= 0$
- Alta varianza cuando p grande o X mal condicionada

Métodos regularizados:

- Introducen sesgo (contraen coeficientes)
- Reducen varianza
- **Pueden mejorar predicción si el trade-off favorece la reducción en varianza**

Parte 1: Ridge Regression (Repaso con Nueva Perspectiva)

Ridge: Tres Perspectivas Equivalentes

1. **Penalización (optimización restringida)**:

$$\hat{\beta}_{Ridge} = \arg\min_{\beta} \left\{ \sum_{i=1}^{n} (y_i - x_i'\beta)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}$$

Ridge: Tres Perspectivas Equivalentes

1. **Penalización (optimización restringida)**:

$$\hat{\beta}_{Ridge} = \arg\min_{\beta} \left\{ \sum_{i=1}^{n} (y_i - x_i'\beta)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}$$

2. **Forma de Lagrange**:

$$\hat{\beta}_{Ridge} = \arg\min_{\beta} ||y - X\beta||^2 \quad \text{sujeto a} \quad \sum_{j=1}^{p} \beta_j^2 \leq t$$

Ridge: Tres Perspectivas Equivalentes

1. **Penalización (optimización restringida)**:

$$\hat{\beta}_{Ridge} = \arg\min_{\beta} \left\{ \sum_{i=1}^{n} (y_i - x_i'\beta)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}$$

2. **Forma de Lagrange**:

$$\hat{\beta}_{Ridge} = \arg\min_{\beta} ||y - X\beta||^2 \quad \text{sujeto a} \quad \sum_{j=1}^{p} \beta_j^2 \leq t$$

3. **Bayesiano (MAP con prior normal)**:

$$\beta_j \sim N(0, \tau^2) \quad \Rightarrow \quad \hat{\beta}_{Ridge} = \text{moda posterior}$$

Solución Ridge en Forma Cerrada

Problema: Minimizar $||y - X\beta||^2 + \lambda||\beta||^2$

Solución Ridge en Forma Cerrada

Problema: Minimizar $||y - X\beta||^2 + \lambda||\beta||^2$

Solución:

$$\hat{\beta}_{Ridge} = (X'X + \lambda I)^{-1}X'y$$

Solución Ridge en Forma Cerrada

Problema: Minimizar $||y - X\beta||^2 + \lambda||\beta||^2$

Solución:

$$\hat{\beta}_{Ridge} = (X'X + \lambda I)^{-1}X'y$$

Comparar con OLS: $\hat{\beta}_{OLS} = (X'X)^{-1}X'y$

Solución Ridge en Forma Cerrada

Problema: Minimizar $||y - X\beta||^2 + \lambda||\beta||^2$

Solución:

$$\hat{\beta}_{Ridge} = (X'X + \lambda I)^{-1}X'y$$

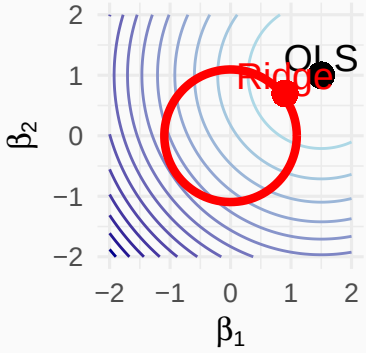
Comparar con OLS: $\hat{\beta}_{OLS} = (X'X)^{-1}X'y$

Observaciones:

1. λI "estabiliza" la inversión (siempre invertible)
2. Cuando $\lambda \to 0$: $\hat{\beta}_{Ridge} \to \hat{\beta}_{OLS}$
3. Cuando $\lambda \to \infty$: $\hat{\beta}_{Ridge} \to 0$

Ridge: Restricción L2

Círculo rojo: restricción, elipses: contornos de R



Ridge en la Práctica: Estandarización

Problema: Ridge penaliza todos los coeficientes por igual.

Si X_1 está en miles y X_2 en unidades, ¡sus coeficientes tienen escalas muy diferentes!

Ridge en la Práctica: Estandarización

Problema: Ridge penaliza todos los coeficientes por igual.

Si X_1 está en miles y X_2 en unidades, ¡sus coeficientes tienen escalas muy diferentes!

Regla práctica

Siempre estandarizar las variables antes de aplicar Ridge:

$$\tilde{X}_j = \frac{X_j - \bar{X}_j}{\text{sd}(X_j)}$$

Ridge en la Práctica: Estandarización

Problema: Ridge penaliza todos los coeficientes por igual.

Si X_1 está en miles y X_2 en unidades, ¡sus coeficientes tienen escalas muy diferentes!

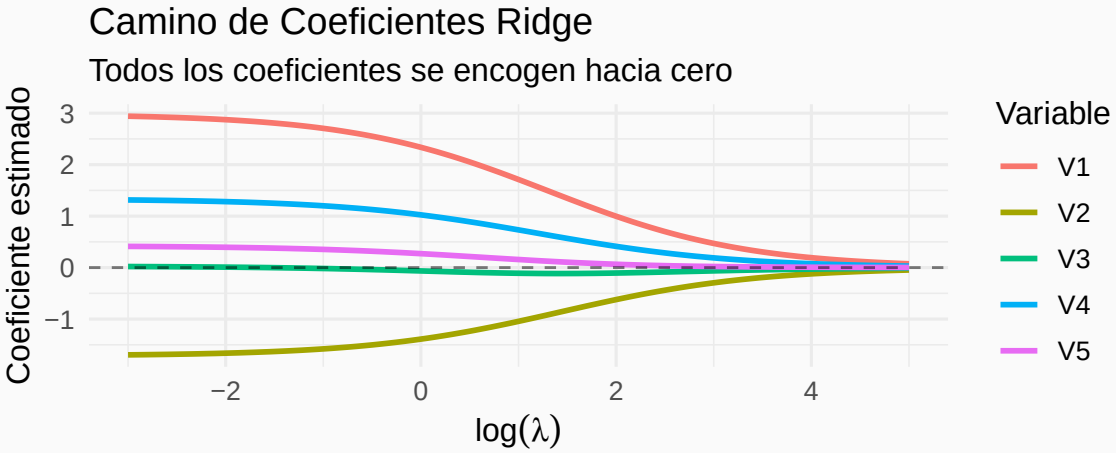
Regla práctica

Siempre estandarizar las variables antes de aplicar Ridge:

$$\tilde{X}_j = \frac{X_j - \bar{X}_j}{\text{sd}(X_j)}$$

Notas:

- No estandarizar y (queremos interpretar magnitudes de predicción)
- Intercepto usualmente no penalizado
- Después de estimar, re-escalar coeficientes a escala original si es necesario



Observación: Todos los coeficientes se encogen, pero **ninguno llega exactamente a cero**.

Parte 2: Lasso - Penalización L1

Introducción al Lasso

Lasso (Least Absolute Shrinkage and Selection Operator) - Tibshirani (1996)

$$\hat{\beta}_{Lasso} = \arg\min_{\beta} \left\{ \sum_{i=1}^{n} (y_i - x_i'\beta)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}$$

Introducción al Lasso

Lasso (Least Absolute Shrinkage and Selection Operator) - Tibshirani (1996)

$$\hat{\beta}_{Lasso} = \arg\min_{\beta} \left\{ \sum_{i=1}^{n} (y_i - x_i'\beta)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}$$

Diferencia clave con Ridge:

- Ridge: Penalización $\sum \beta_j^2$ (L2)
- Lasso: Penalización $\sum |\beta_j|$ (L1)

Introducción al Lasso

Lasso (Least Absolute Shrinkage and Selection Operator) - Tibshirani (1996)

$$\hat{\beta}_{Lasso} = \arg\min_{\beta} \left\{ \sum_{i=1}^{n} (y_i - x_i'\beta)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}$$

Diferencia clave con Ridge:

- Ridge: Penalización $\sum \beta_j^2$ (L2)
- Lasso: Penalización $\sum |\beta_j|$ (L1)

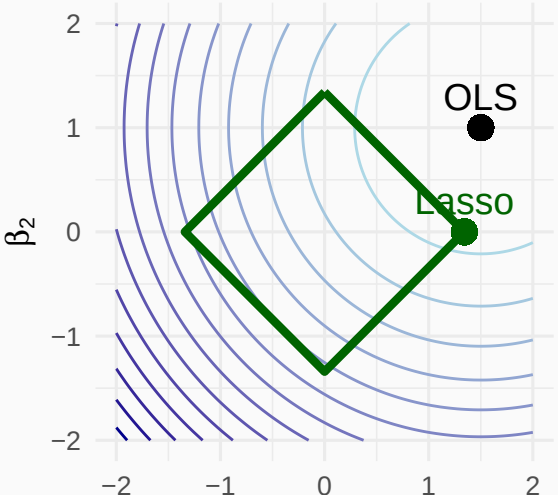
Consecuencia sorprendente:

Selección de Variables

Lasso puede hacer que algunos $\hat{\beta}_j = 0$ **exactamente**

Lasso: Restricción L1

Diamante verde: restricción, elipses: contornos de RS



¿Por Qué Lasso Hace Selección?

Intuición: La forma del conjunto de restricción importa.

¿Por Qué Lasso Hace Selección?

Intuición: La forma del conjunto de restricción importa.

L1 (Lasso): Diamante con esquinas en los ejes

- Alta probabilidad de que contornos de RSS toquen una esquina
- En las esquinas: algunos coeficientes son exactamente cero

¿Por Qué Lasso Hace Selección?

Intuición: La forma del conjunto de restricción importa.

L1 (Lasso): Diamante con esquinas en los ejes

- Alta probabilidad de que contornos de RSS toquen una esquina
- En las esquinas: algunos coeficientes son exactamente cero

L2 (Ridge): Círculo sin esquinas

- Punto de contacto usualmente no está en los ejes
- Coeficientes pequeños pero no exactamente cero

¿Por Qué Lasso Hace Selección?

Intuición: La forma del conjunto de restricción importa.

L1 (Lasso): Diamante con esquinas en los ejes

- Alta probabilidad de que contornos de RSS toquen una esquina
- En las esquinas: algunos coeficientes son exactamente cero

L2 (Ridge): Círculo sin esquinas

- Punto de contacto usualmente no está en los ejes
- Coeficientes pequeños pero no exactamente cero

Consecuencia Práctica

Lasso hace **selección automática de variables**: elimina variables irrelevantes

No Hay Solución Cerrada para Lasso

A diferencia de Ridge: No existe fórmula cerrada para $\hat{\beta}_{Lasso}$

No Hay Solución Cerrada para Lasso

A diferencia de Ridge: No existe fórmula cerrada para $\hat{\beta}_{Lasso}$

Razón: La función $|\beta_j|$ no es diferenciable en cero

No Hay Solución Cerrada para Lasso

A diferencia de Ridge: No existe fórmula cerrada para $\hat{\beta}_{Lasso}$

Razón: La función $|\beta_j|$ no es diferenciable en cero

Algoritmos de optimización:

1. **Coordinate descent** (más común): Optimiza cada β_j manteniendo otros fijos
2. **LARS** (Least Angle Regression): Camino completo de soluciones eficientemente
3. **Proximal gradient descent**: Métodos generales de optimización

No Hay Solución Cerrada para Lasso

A diferencia de Ridge: No existe fórmula cerrada para $\hat{\beta}_{Lasso}$

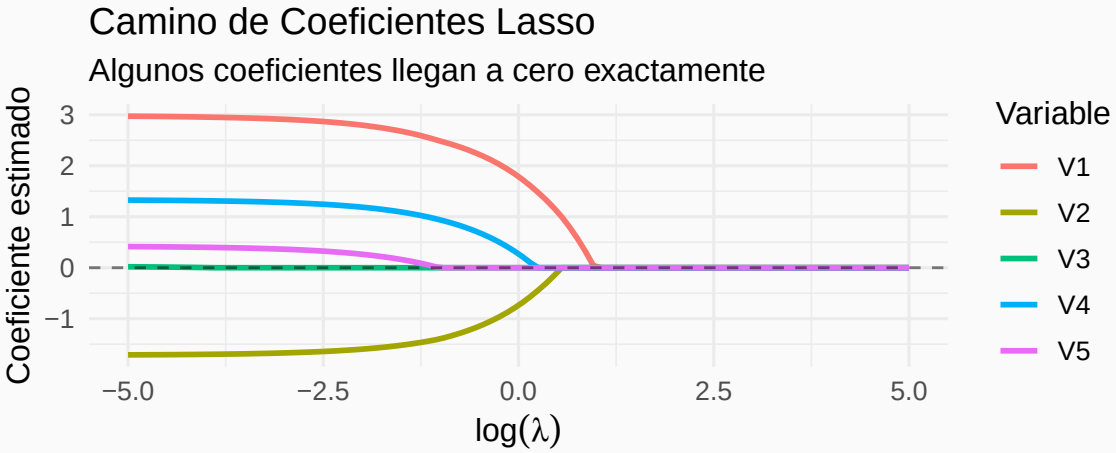
Razón: La función $|\beta_j|$ no es diferenciable en cero

Algoritmos de optimización:

1. **Coordinate descent** (más común): Optimiza cada β_j manteniendo otros fijos
2. **LARS** (Least Angle Regression): Camino completo de soluciones eficientemente
3. **Proximal gradient descent**: Métodos generales de optimización

En la práctica: Usamos implementaciones eficientes

- R: paquete `glmnet` (Friedman, Hastie, Tibshirani)
- Python: `scikit-learn` con `Lasso` o `ElasticNet`



Diferencia con Ridge: Coeficientes llegan a cero y permanecen ahí.

Lasso: Interpretación Bayesiana

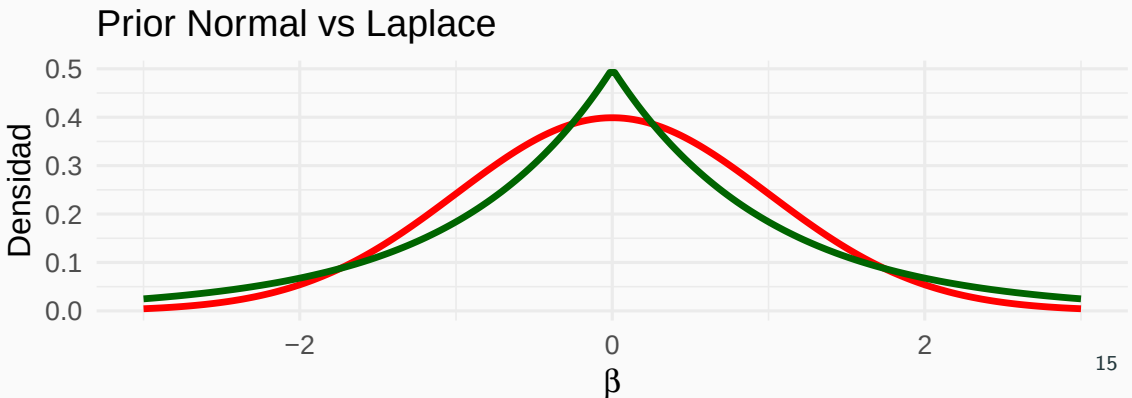
Lasso corresponde a MAP con **prior de Laplace** (double exponential):

$$\pi(\beta_j) = \frac{\lambda}{2} e^{-\lambda|\beta_j|}$$

Lasso: Interpretación Bayesiana

Lasso corresponde a MAP con **prior de Laplace** (double exponential):

$$\pi(\beta_j) = \frac{\lambda}{2} e^{-\lambda|\beta_j|}$$



Comparación: Ridge vs Lasso

Aspecto	Ridge	Lasso
Penalización	$\sum \beta_j^2$ (L2)	$\sum \lvert\beta_j\rvert$ (L1)
Prior Bayesiano	Normal $N(0, \tau^2)$	Laplace
Solución cerrada	Sí: $(X'X + \lambda I)^{-1}X'y$	No (optimización numérica)
Selección de variables	No (coef. pequeños)	Sí (coef. exactamente cero)
Correlaciones altas	Coef. similares	Elige una arbitrariamente
Interpretación	Shrinkage uniforme	Sparse + shrinkage

¿Cuándo Usar Cada Uno?

Ridge es preferible cuando:

- Todas (o la mayoría) las variables son relevantes
- Variables están altamente correlacionadas
- Objetivo principal es predicción (no interpretación)

¿Cuándo Usar Cada Uno?

Ridge es preferible cuando:

- Todas (o la mayoría) las variables son relevantes
- Variables están altamente correlacionadas
- Objetivo principal es predicción (no interpretación)

Lasso es preferible cuando:

- Se sospecha que muchas variables son irrelevantes (modelo sparse verdadero)
- Se desea interpretación (modelo más simple)
- Se necesita selección automática de variables

¿Cuándo Usar Cada Uno?

Ridge es preferible cuando:

- Todas (o la mayoría) las variables son relevantes
- Variables están altamente correlacionadas
- Objetivo principal es predicción (no interpretación)

Lasso es preferible cuando:

- Se sospecha que muchas variables son irrelevantes (modelo sparse verdadero)
- Se desea interpretación (modelo más simple)
- Se necesita selección automática de variables

En la práctica:

- Probar ambos y comparar con cross-validation
- Considerar Elastic Net (combina ambos) si hay duda

Parte 3: Eligiendo el Parámetro de Penalización

El Parámetro λ

Para Ridge y Lasso, debemos elegir λ:

$$\hat{\beta}(\lambda) = \arg\min_{\beta} \left\{ ||y - X\beta||^2 + \lambda \cdot \text{Penalty}(\beta) \right\}$$

El Parámetro λ

Para Ridge y Lasso, debemos elegir λ:

$$\hat{\beta}(\lambda) = \arg\min_{\beta} \left\{ ||y - X\beta||^2 + \lambda \cdot \text{Penalty}(\beta) \right\}$$

Valores extremos:

- $\lambda = 0$: No penalización $\rightarrow$ OLS (posible overfitting)
- $\lambda \rightarrow \infty$: Penalización total $\rightarrow \hat{\beta} = 0$ (underfitting)

El Parámetro λ

Para Ridge y Lasso, debemos elegir λ:

$$\hat{\beta}(\lambda) = \arg\min_{\beta} \left\{ ||y - X\beta||^2 + \lambda \cdot \text{Penalty}(\beta) \right\}$$

Valores extremos:

- $\lambda = 0$: No penalización $\rightarrow$ OLS (posible overfitting)
- $\lambda \rightarrow \infty$: Penalización total $\rightarrow \hat{\beta} = 0$ (underfitting)

Pregunta clave: ¿Cómo elegir λ óptimo?

El Parámetro λ

Para Ridge y Lasso, debemos elegir λ:

$$\hat{\beta}(\lambda) = \arg\min_{\beta} \left\{ ||y - X\beta||^2 + \lambda \cdot \text{Penalty}(\beta) \right\}$$

Valores extremos:

- $\lambda = 0$: No penalización $\rightarrow$ OLS (posible overfitting)
- $\lambda \rightarrow \infty$: Penalización total $\rightarrow \hat{\beta} = 0$ (underfitting)

Pregunta clave: ¿Cómo elegir λ óptimo?

Respuesta: Cross-Validation

Elegir λ que minimiza el error de predicción en datos no usados en el entrenamiento

K-Fold Cross-Validation

Procedimiento:

1. Dividir datos en K grupos (folds) de tamaño similar
2. Para cada valor de λ candidato:
   - Para $k = 1, \ldots, K$:
     - Entrenar en todos los folds excepto k
     - Predecir en fold k
     - Calcular error: $\text{MSE}_k(\lambda)$
   - Promedio: $\text{CV}(\lambda) = \frac{1}{K} \sum_{k=1}^{K} \text{MSE}_k(\lambda)$
3. Elegir $\lambda^* = \arg\min_\lambda \text{CV}(\lambda)$

K-Fold Cross-Validation

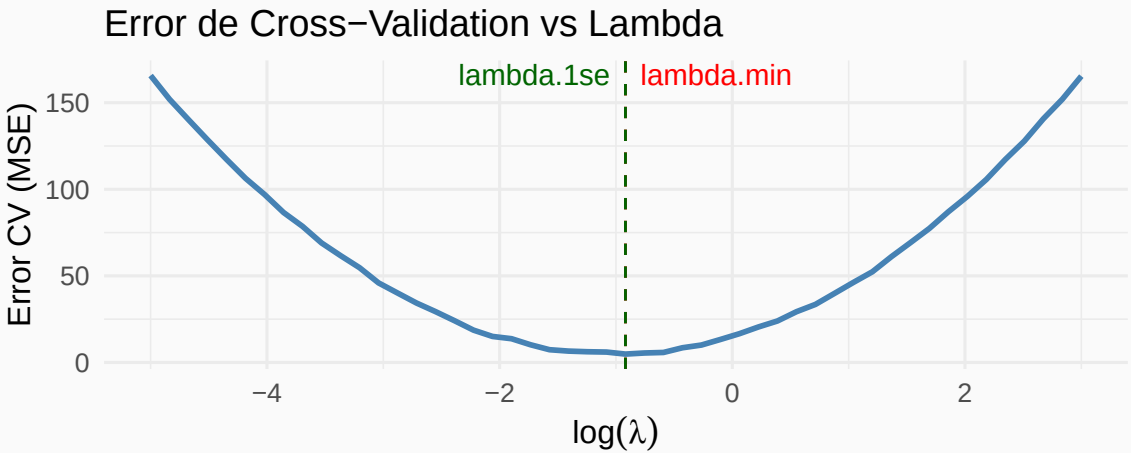
Procedimiento:

1. Dividir datos en K grupos (folds) de tamaño similar
2. Para cada valor de λ candidato:
   - Para $k = 1, \ldots, K$:
     - Entrenar en todos los folds excepto k
     - Predecir en fold k
     - Calcular error: $\text{MSE}_k(\lambda)$
   - Promedio: $\text{CV}(\lambda) = \frac{1}{K} \sum_{k=1}^{K} \text{MSE}_k(\lambda)$
3. Elegir $\lambda^* = \arg\min_\lambda \text{CV}(\lambda)$

Valores comunes de K:

- $K = 5$ o $K = 10$ (compromiso entre varianza y costo computacional)
- $K = n$ (leave-one-out CV): muy costoso, poca práctica común

Ilustración de CV



La Regla de Un Error Estándar

Problema: λ_{min} puede ser inestable (depende de los datos específicos)

La Regla de Un Error Estándar

Problema: λ_{min} puede ser inestable (depende de los datos específicos)

Regla de un error estándar (Breiman et al., 1984):

Elegir el modelo más parsimonioso (mayor λ) cuyo error esté dentro de un error estándar del mínimo:

$$\lambda_{1SE} = \max \{\lambda : \text{CV}(\lambda) \leq \text{CV}(\lambda_{min}) + SE(\lambda_{min})\}$$

La Regla de Un Error Estándar

Problema: λ_{min} puede ser inestable (depende de los datos específicos)

Regla de un error estándar (Breiman et al., 1984):

Elegir el modelo más parsimonioso (mayor λ) cuyo error esté dentro de un error estándar del mínimo:

$$\lambda_{1SE} = \max \{\lambda : \text{CV}(\lambda) \leq \text{CV}(\lambda_{min}) + SE(\lambda_{min})\}$$

Intuición:

- Modelos más simples son más interpretables
- Diferencias pequeñas en error CV no son estadísticamente significativas
- Preferir simplicidad cuando el rendimiento es comparable

Implementación en R con glmnet

```r
# Cargar paquete
library(glmnet)

# Preparar datos (X debe ser matriz, y vector)
X <- as.matrix(datos[, -1])  # Variables predictoras
y <- datos$y                  # Variable respuesta

# Cross-validation para Lasso
cv_lasso <- cv.glmnet(X, y, alpha = 1, nfolds = 10)

# Extraer lambdas óptimos
lambda_min <- cv_lasso$lambda.min    # Lambda con error mínimo
lambda_1se <- cv_lasso$lambda.1se    # Lambda con regla 1SE
```

Alternativa: Criterios de Información

Además de CV, existen criterios basados en teoría de la información:

AIC (Akaike Information Criterion):

$$\text{AIC} = n \log(\text{RSS}/n) + 2 \cdot df$$

Alternativa: Criterios de Información

Además de CV, existen criterios basados en teoría de la información:

AIC (Akaike Information Criterion):

$$\text{AIC} = n \log(\text{RSS}/n) + 2 \cdot df$$

BIC (Bayesian Information Criterion):

$$\text{BIC} = n \log(\text{RSS}/n) + \log(n) \cdot df$$

Alternativa: Criterios de Información

Además de CV, existen criterios basados en teoría de la información:

AIC (Akaike Information Criterion):

$$\text{AIC} = n \log(\text{RSS}/n) + 2 \cdot df$$

BIC (Bayesian Information Criterion):

$$\text{BIC} = n \log(\text{RSS}/n) + \log(n) \cdot df$$

Para modelos penalizados:

- df = grados de libertad efectivos (no simplemente número de coeficientes no-cero)
- BIC penaliza complejidad más fuertemente que AIC
- Útiles como complemento a CV, pero CV generalmente preferido para predicción

Parte 4: Aplicación Económica

Aplicación: Determinantes de Salarios

Pregunta: ¿Qué factores determinan los salarios?

Aplicación: Determinantes de Salarios

Pregunta: ¿Qué factores determinan los salarios?

Datos: Current Population Survey (CPS)

- Variable respuesta: log(salario por hora)
- Predictores:
  - Educación (años, dummies por nivel)
  - Experiencia (lineal, cuadrática, cúbica)
  - Demografía (edad, género, raza, estado civil)
  - Geografía (región, urbano/rural)
  - Ocupación (dummies para ~30 categorías)
  - Industria (dummies para ~20 sectores)

Aplicación: Determinantes de Salarios

Pregunta: ¿Qué factores determinan los salarios?

Datos: Current Population Survey (CPS)

- Variable respuesta: log(salario por hora)
- Predictores:
  - Educación (años, dummies por nivel)
  - Experiencia (lineal, cuadrática, cúbica)
  - Demografía (edad, género, raza, estado civil)
  - Geografía (región, urbano/rural)
  - Ocupación (dummies para ~30 categorías)
  - Industria (dummies para ~20 sectores)

Problema: Con interacciones, ¡podríamos tener 100+ variables!

El Problema de la Multicolinealidad

Con muchas variables categóricas y sus interacciones:

El Problema de la Multicolinealidad

Con muchas variables categóricas y sus interacciones:

Problemas de OLS:

1. Coeficientes inestables (alta varianza)
2. Algunos efectos difíciles de separar (ocupación vs industria)
3. Riesgo de overfitting
4. Interpretación difícil con 100+ coeficientes

El Problema de la Multicolinealidad

Con muchas variables categóricas y sus interacciones:

Problemas de OLS:

1. Coeficientes inestables (alta varianza)
2. Algunos efectos difíciles de separar (ocupación vs industria)
3. Riesgo de overfitting
4. Interpretación difícil con 100+ coeficientes

Solución con regularización:

- **Ridge**: Estabiliza estimaciones, mejora predicción
- **Lasso**: Identifica variables más importantes, simplifica modelo

Demostración en RStudio

Vamos a trabajar con código en vivo:

Pasos:

1. Cargar y preparar datos de salarios
2. Crear matriz de diseño con muchas variables
3. Comparar OLS, Ridge y Lasso
4. Usar CV para elegir λ
5. Interpretar resultados y coeficientes
6. Comparar predicción out-of-sample

Archivo: `clase4_wage_regression.R`

Demostración en RStudio

Vamos a trabajar con código en vivo:

Pasos:

1. Cargar y preparar datos de salarios
2. Crear matriz de diseño con muchas variables
3. Comparar OLS, Ridge y Lasso
4. Usar CV para elegir λ
5. Interpretar resultados y coeficientes
6. Comparar predicción out-of-sample

Archivo: `clase4_wage_regression.R`

Preguntas para considerar durante la demo:

- ¿Cuántas variables selecciona Lasso?
- ¿Son las mismas que esperaríamos por teoría económica?

Reflexiones Finales

Conexión con Clases Anteriores

De James-Stein a Métodos Generales:

Conexión con Clases Anteriores

De James-Stein a Métodos Generales:

Clase 3 (James-Stein):

- Shrinkage de estimaciones individuales hacia media común
- Mejora MSE cuando $p \geq 3$
- Interpretación Bayesiana empírica

Conexión con Clases Anteriores

De James-Stein a Métodos Generales:

Clase 3 (James-Stein):

- Shrinkage de estimaciones individuales hacia media común
- Mejora MSE cuando $p \geq 3$
- Interpretación Bayesiana empírica

Hoy (Ridge/Lasso):

- Shrinkage en contexto de regresión
- Ridge: shrinkage uniforme (como James-Stein)
- Lasso: shrinkage + selección
- Mismo principio: reducir varianza a costo de introducir sesgo

Conexión con Clases Anteriores

De James-Stein a Métodos Generales:

Clase 3 (James-Stein):

- Shrinkage de estimaciones individuales hacia media común
- Mejora MSE cuando $p \geq 3$
- Interpretación Bayesiana empírica

Hoy (Ridge/Lasso):

- Shrinkage en contexto de regresión
- Ridge: shrinkage uniforme (como James-Stein)
- Lasso: shrinkage + selección
- Mismo principio: reducir varianza a costo de introducir sesgo

Próxima clase: Extenderemos a métodos no-lineales (splines, GAMs)

Puntos Clave para Recordar

1. **Trade-off bias-variance** es fundamental en aprendizaje estadístico
2. **Ridge** (L2): Encoge todos los coeficientes, estabiliza estimación
3. **Lasso** (L1): Encoge + selecciona variables (sparsity)
4. **Cross-validation** es el estándar para elegir λ
5. **Regla 1SE**: Preferir modelos más simples cuando el rendimiento es comparable
6. **Interpretación**: Cuidado con inferencia causal usando coeficientes penalizados

Guía Práctica: ¿Qué Usar Cuándo?

Situación	Recomendación
$p < n$, variables no correlacionadas	OLS puede ser suficiente
$p < n$, alta multicolinealidad	Ridge para estabilizar
p grande, mayoría variables relevantes	Ridge
p grande, modelo sparse verdadero	Lasso
No sabes si sparse o no	Probar ambos con CV
Objetivo es solo predicción	Cualquiera (CV decide)
Necesitas interpretación	Lasso (menos variables)
$p > n$	Lasso o Elastic Net