Machine Learning y Inferencia Causal

Double ML, Model Selection y Sintesis

2025

Hoja de Ruta del Curso

Donde hemos estado:

- Clases 1-3: Fundamentos Bayesianos, Decision Theory, Shrinkage
- Clase 4: Regresion Penalizada (Ridge/Lasso)
- Clase 5: Mas alla de linealidad (Splines)
- Clase 6: Metodos basados en arboles (RF, Boosting)

Hoy: Clase final

1. El problema: ML para inferencia causal
2. Double Machine Learning (DML)
3. Otros metodos causales con ML
4. Model Selection profundo
5. Sintesis e integracion

El Dilema Fundamental

Dos objetivos diferentes en econometria:
Prediccion

- Minimizar MSE out-of-sample
- $\hat{Y}$ lo mas cercano a Y
- Regularizacion es buena
- Sesgo aceptable

El Dilema Fundamental

Dos objetivos diferentes en econometria:

Prediccion

- Minimizar MSE out-of-sample
- $\hat{Y}$ lo mas cercano a Y
- Regularizacion es buena
- Sesgo aceptable

Inferencia Causal

- Estimar parametro causal θ
- Necesitamos $\hat{\theta} \to \theta$
- Sesgo es problematico
- Necesitamos errores estandar

El Dilema Fundamental

Dos objetivos diferentes en econometria:

Prediccion

- Minimizar MSE out-of-sample
- $\hat{Y}$ lo mas cercano a Y
- Regularizacion es buena
- Sesgo aceptable

Inferencia Causal

- Estimar parametro causal θ
- Necesitamos $\hat{\theta} \to \theta$
- Sesgo es problematico
- Necesitamos errores estandar

La pregunta clave

¿Podemos usar ML para mejorar estimacion de efectos causales?

Parte 1: Por que ML Ingenuo Falla

Motivacion: Retornos a la Educacion

Pregunta economica clasica: ¿Cual es el retorno a un año adicional de educacion?

Modelo basico:

$$\log(\text{wage}_i) = \theta \cdot \text{educ}_i + X_i'\beta + \varepsilon_i$$

donde:

- θ = parametro causal de interes (retorno a educacion)
- X_i = controles (experiencia, region, etc.)
- β = parametros de molestia (nuisance parameters)

Problema: Con muchos controles y sus interacciones, ¿como estimar β?

Enfoque Ingenuo 1: ML Directo

Idea: Usar Lasso/RF para estimar el modelo completo

$$\log(\text{wage}_i) = \theta \cdot \text{educ}_i + f(X_i) + \varepsilon_i$$

donde $f(X_i)$ se estima con ML.

Enfoque Ingenuo 1: ML Directo

Idea: Usar Lasso/RF para estimar el modelo completo

$$\log(\text{wage}_i) = \theta \cdot \text{educ}_i + f(X_i) + \varepsilon_i$$

donde $f(X_i)$ se estima con ML.

Problema: Regularization Bias

- Lasso encoge TODOS los coeficientes, incluyendo θ
- RF/Boosting no dan errores estandar validos
- La regularizacion que mejora prediccion introduce sesgo en θ

Enfoque Ingenuo 1: ML Directo

Idea: Usar Lasso/RF para estimar el modelo completo

$$\log(\text{wage}_i) = \theta \cdot \text{educ}_i + f(X_i) + \varepsilon_i$$

donde $f(X_i)$ se estima con ML.

Problema: Regularization Bias

- Lasso encoge TODOS los coeficientes, incluyendo θ
- RF/Boosting no dan errores estandar validos
- La regularizacion que mejora prediccion introduce sesgo en θ

Resultado:

$$\hat{\theta}_{ML} \not\to \theta \quad \text{(sesgado)}$$

Enfoque Ingenuo 2: Post-Selection

Idea: Dos etapas

1. Usar Lasso para seleccionar variables importantes
2. Estimar modelo final con OLS usando solo variables seleccionadas

Enfoque Ingenuo 2: Post-Selection

Idea: Dos etapas

1. Usar Lasso para seleccionar variables importantes
2. Estimar modelo final con OLS usando solo variables seleccionadas

Problema: Selection Bias

- Si Lasso elimina una variable importante por casualidad
- OLS en el modelo seleccionado es sesgado
- Errores estandar ignoran incertidumbre de seleccion
- Inferencia invalida (tests de hipotesis incorrectos)

Enfoque Ingenuo 2: Post-Selection

Idea: Dos etapas

1. Usar Lasso para seleccionar variables importantes
2. Estimar modelo final con OLS usando solo variables seleccionadas

Problema: Selection Bias

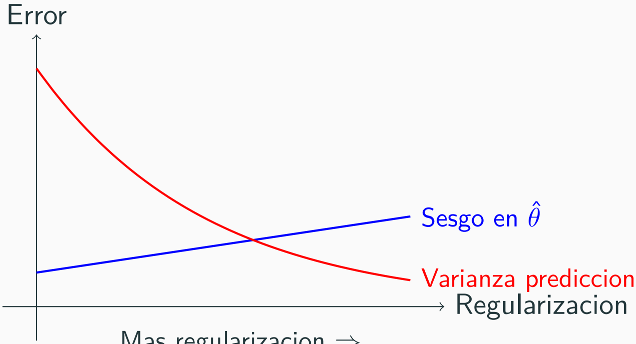
- Si Lasso elimina una variable importante por casualidad
- OLS en el modelo seleccionado es sesgado
- Errores estandar ignoran incertidumbre de seleccion
- Inferencia invalida (tests de hipotesis incorrectos)

Resultado:

$$\text{SE}(\hat{\theta}) \text{ subestimado} \Rightarrow \text{tests invalidos}$$

El Problema Fundamental

Trade-off inevitable con ML ingenuo:



Mas regularizacion $\Rightarrow$
mejor prediccion, peor inferencia

Necesitamos: Metodos que usen ML para reducir varianza SIN introducir sesgo en θ.

Parte 2: Double Machine Learning

La Idea Central de DML (Chernozhukov et al. 2018)

En muchos modelos econometricos:

$$Y = \theta D + g(X) + U$$

$$D = m(X) + V$$

- Y = outcome
- D = variable de interes (tratamiento, educacion, etc.)
- X = controles de alta dimension
- $g(X)$ y $m(X)$ = funciones de molestia (nuisance)

La Idea Central de DML (Chernozhukov et al. 2018)

En muchos modelos econometricos:

$$Y = \theta D + g(X) + U$$

$$D = m(X) + V$$

- Y = outcome
- D = variable de interes (tratamiento, educacion, etc.)
- X = controles de alta dimension
- $g(X)$ y $m(X)$ = funciones de molestia (nuisance)

Estrategia:

1. Usar ML para estimar $g(X)$ y $m(X)$ (prediccion)
2. Usar residuos para estimar θ (inferencia)
3. Cross-fitting para evitar overfitting bias

El Modelo Parcialmente Lineal

Setup formal:

$$Y_i = \theta D_i + g_0(X_i) + U_i$$
$$D_i = m_0(X_i) + V_i$$

Supuestos:

- $E[U_i|X_i, D_i] = 0$ (exogeneidad condicional)
- $E[V_i|X_i] = 0$ (por construccion)
- $g_0(\cdot)$ y $m_0(\cdot)$ son funciones desconocidas

El Modelo Parcialmente Lineal

Setup formal:

$$Y_i = \theta D_i + g_0(X_i) + U_i$$
$$D_i = m_0(X_i) + V_i$$

Supuestos:

- $E[U_i|X_i, D_i] = 0$ (exogeneidad condicional)
- $E[V_i|X_i] = 0$ (por construccion)
- $g_0(\cdot)$ y $m_0(\cdot)$ son funciones desconocidas

Objetivo: Estimar θ sin especificar forma funcional de g_0 y m_0.

Ventaja: Usar ML para las funciones no-lineales complejas.

Algoritmo DML: Version Simple

Naive approach (NO funciona bien):

1. Estimar $\hat{g}(X_i) = E[Y_i|X_i]$ con ML
2. Estimar $\hat{m}(X_i) = E[D_i|X_i]$ con ML
3. Computar residuos:
   - $\tilde{Y}_i = Y_i - \hat{g}(X_i)$
   - $\tilde{D}_i = D_i - \hat{m}(X_i)$
4. Regresar $\tilde{Y}_i$ en $\tilde{D}_i$:

$$\hat{\theta} = \frac{\sum_i \tilde{D}_i \tilde{Y}_i}{\sum_i \tilde{D}_i^2}$$

Algoritmo DML: Version Simple

Naive approach (NO funciona bien):

1. Estimar $\hat{g}(X_i) = E[Y_i|X_i]$ con ML
2. Estimar $\hat{m}(X_i) = E[D_i|X_i]$ con ML
3. Computar residuos:
   - $\tilde{Y}_i = Y_i - \hat{g}(X_i)$
   - $\tilde{D}_i = D_i - \hat{m}(X_i)$
4. Regresar $\tilde{Y}_i$ en $\tilde{D}_i$:

$$\hat{\theta} = \frac{\sum_i \tilde{D}_i \tilde{Y}_i}{\sum_i \tilde{D}_i^2}$$

Problema: Overfitting bias - estimas $\hat{g}$ y $\hat{m}$ en los mismos datos donde calculas residuos.

Sample splitting + cross-fitting

Algoritmo (2-fold version):

1. **Dividir datos** aleatoriamente: I_1 y I_2
2. **Primera iteracion:**
   - Estimar $\hat{g}^{(-1)}$ y $\hat{m}^{(-1)}$ usando datos en I_1
   - Predecir en I_2: $\tilde{Y}_i^{(1)} = Y_i - \hat{g}^{(-1)}(X_i)$, $\tilde{D}_i^{(1)} = D_i - \hat{m}^{(-1)}(X_i)$
3. **Segunda iteracion:**
   - Estimar $\hat{g}^{(-2)}$ y $\hat{m}^{(-2)}$ usando datos en I_2
   - Predecir en I_1: $\tilde{Y}_i^{(2)} = Y_i - \hat{g}^{(-2)}(X_i)$, $\tilde{D}_i^{(2)} = D_i - \hat{m}^{(-2)}(X_i)$
4. **Pooled estimation:**

$$\hat{\theta}_{DML} = \frac{\sum_{i=1}^{n} \tilde{D}_i \tilde{Y}_i}{\sum_{i=1}^{n} \tilde{D}_i^2}$$

Por que Funciona DML

Propiedad clave: Neyman Orthogonality

La funcion de momento es aproximadamente insensible a errores de estimacion en g y m:

$$\psi(W_i; \theta, g, m) = (Y_i - \theta D_i - g(X_i))(D_i - m(X_i))$$

Matematicamente:

$$\frac{\partial}{\partial g} E[\psi(W; \theta_0, g, m)]\Big|_{g=g_0, m=m_0} = 0$$

Por que Funciona DML

Propiedad clave: Neyman Orthogonality

La funcion de momento es aproximadamente insensible a errores de estimacion en g y m:

$$\psi(W_i; \theta, g, m) = (Y_i - \theta D_i - g(X_i))(D_i - m(X_i))$$

Matematicamente:

$$\frac{\partial}{\partial g} E[\psi(W; \theta_0, g, m)]\Big|_{g=g_0, m=m_0} = 0$$

En palabras:

- Errores de primer orden en $\hat{g}$ y $\hat{m}$ no afectan $\hat{\theta}$
- Solo errores de segundo orden importan
- Con regularizacion: errores de segundo orden son pequenos

Error estandar:

$$\mathrm{SE}(\hat{\theta}_{DML}) = \sqrt{\frac{1}{n}\frac{\sum_i \hat{\psi}_i^2}{(\sum_i \tilde{D}_i^2/n)^2}}$$

donde $\hat{\psi}_i = \tilde{Y}_i - \hat{\theta}_{DML}\tilde{D}_i$ son residuos estimados.

Inferencia en DML

Error estandar:

$$\text{SE}(\hat{\theta}_{DML}) = \sqrt{\frac{1}{n}\frac{\sum_i \hat{\psi}_i^2}{(\sum_i \tilde{D}_i^2/n)^2}}$$

donde $\hat{\psi}_i = \tilde{Y}_i - \hat{\theta}_{DML}\tilde{D}_i$ son residuos estimados.

Propiedades asintoticas:

$$\sqrt{n}(\hat{\theta}_{DML} - \theta_0) \xrightarrow{d} N(0, V)$$

Crucialmente: Esto es valido incluso si $p > n$ (mas variables que observaciones)!

Ejemplo: Retornos a Educacion con DML

Setup:

- Y: log(wage)
- D: years of education
- X: experiencia, experiencia^2, edad, genero, region, ocupacion (30 dummies), industria (20 dummies), etc.

Total: $\sim$100 variables en X con interacciones.

Ejemplo: Retornos a Educacion con DML

Setup:

- Y: log(wage)
- D: years of education
- X: experiencia, experiencia^2, edad, genero, region, ocupacion (30 dummies), industria (20 dummies), etc.

Total: $\sim$100 variables en X con interacciones.

Comparacion de metodos:

1. **OLS basico:** Solo controles lineales
2. **Lasso directo:** Sesgo de regularizacion
3. **DML con Lasso:** Para $g(X)$ y $m(X)$
4. **DML con Random Forest:** Mas flexible

Implementacion en R: Setup

```r
# Cargar paquetes necesarios
library(DoubleML)
library(mlr3)
library(mlr3learners)
library(data.table)

# Preparar datos en formato DoubleML
data_dml <- double_ml_data_from_data_frame(
  df = wage_data,
  y_col = "log_wage",        # Outcome
  d_cols = "education",       # Tratamiento
  x_cols = control_vars       # Vector de controles
)
```

Implementacion: Estimacion

```r
# Definir learner para funciones de molestia
# Opcion 1: Random Forest
learner_g <- lrn("regr.ranger", num.trees = 500)
learner_m <- lrn("regr.ranger", num.trees = 500)

# Opcion 2: Lasso
learner_g <- lrn("regr.cv_glmnet", s = "lambda.min")
learner_m <- lrn("regr.cv_glmnet", s = "lambda.min")

# Estimacion DML
dml_obj <- DoubleMLPLR$new(
  data_dml,
  ml_g = learner_g,
  ml_m = learner_m,
```

Demo en RStudio

Vamos a ver el codigo completo:

Archivo: `doubleml_example.R`

Lo que haremos:

1. Simular datos con efecto heterogeneo
2. Comparar OLS, Lasso, DML con Lasso, DML con RF
3. Examinar sesgo y cobertura de intervalos
4. Aplicar a datos reales de salarios

Resultados Esperados

Simulacion tipica:

Metodo	Estimacion	Error Std	95% CI
OLS (controles basicos)	0.082	0.015	[0.053, 0.111]
Lasso (directo)	0.073	0.011	[0.051, 0.095]
DML + Lasso	0.098	0.014	[0.071, 0.125]
DML + Random Forest	0.101	0.013	[0.076, 0.126]
Verdad	0.100	NA	NA

Observaciones:

- OLS subestima (sesgo de variable omitida)
- Lasso directo tiene sesgo de regularizacion
- DML recupera el parametro correcto con IC validos

Extensiones de DML

1. **Modelos con multiples tratamientos:**

$$Y_i = \theta_1 D_{1i} + \theta_2 D_{2i} + g_0(X_i) + U_i$$

2. **Variables instrumentales:**

$$Y_i = \theta D_i + g_0(X_i) + U_i$$
$$D_i = m_0(Z_i, X_i) + V_i$$

3. **Efectos parciales:**

$$Y_i = g_0(D_i, X_i) + U_i$$

Estimar $\theta(d) = \frac{\partial}{\partial d} g_0(d, X)$

Cuando Usar DML

DML es apropiado cuando:

1. Tienes muchas variables de control (p grande)
2. Forma funcional de controles es desconocida/compleja
3. Necesitas estimacion causal + errores estandar validos
4. Exogeneidad condicional es defendible

Cuando Usar DML

DML es apropiado cuando:

1. Tienes muchas variables de control (p grande)
2. Forma funcional de controles es desconocida/compleja
3. Necesitas estimacion causal + errores estandar validos
4. Exogeneidad condicional es defendible

DML NO resuelve:

- Endogeneidad fundamental (necesitas IV o experimento)
- Causalidad inversa
- Variables omitidas no observables

Cuando Usar DML

DML es apropiado cuando:

1. Tienes muchas variables de control (p grande)
2. Forma funcional de controles es desconocida/compleja
3. Necesitas estimacion causal + errores estandar validos
4. Exogeneidad condicional es defendible

DML NO resuelve:

- Endogeneidad fundamental (necesitas IV o experimento)
- Causalidad inversa
- Variables omitidas no observables

En resumen: DML es una herramienta para "high-dimensional controls", no para identificacion causal per se.

Parte 3: Otros Metodos Causales con ML

Panorama de Metodos

Familia de metodos que usan ML para parametros causales:

1. **Double ML**: Estimacion de efectos promedio
2. **Causal Forests**: Efectos heterogeneos de tratamiento
3. **Post-Double-Selection**: Seleccion robusta de controles
4. **Generalized Random Forests**: Extension flexible
5. **Targeted Learning**: Estimadores doblemente robustos

Enfoque comun: Separar prediccion (ML) de inferencia causal.

Causal Forests (Athey & Wager 2019)

Objetivo: Estimar efectos heterogeneos de tratamiento

$$\tau(x) = E[Y_i(1) - Y_i(0)|X_i = x]$$

Idea:

- Modificar Random Forests para estimar efectos causales
- Cada arbol construye particiones que maximizan heterogeneidad en efectos
- Agregar arboles para estimar $\tau(x)$ con intervalos de confianza

Causal Forests (Athey & Wager 2019)

Objetivo: Estimar efectos heterogeneos de tratamiento

$$\tau(x) = E[Y_i(1) - Y_i(0)|X_i = x]$$

Idea:

- Modificar Random Forests para estimar efectos causales
- Cada arbol construye particiones que maximizan heterogeneidad en efectos
- Agregar arboles para estimar $\tau(x)$ con intervalos de confianza

Ventajas:

- Estima efectos heterogeneos sin especificar interacciones
- Inferencia valida (errores estandar honestos)
- Robusto a especificacion

Ejemplo Conceptual: Causal Forest

Pregunta: ¿El efecto de entrenamiento laboral varia con educacion y experiencia?

Post-Double-Selection (Belloni et al.)

Problema: Con muchas variables, ¿cuales incluir como controles?

Estrategia de dos pasos:

1. **Primera seleccion:** Usar Lasso para regresar Y en X
   - Seleccionar variables $\hat{S}_Y$
2. **Segunda seleccion:** Usar Lasso para regresar D en X
   - Seleccionar variables $\hat{S}_D$
3. **OLS final:** Incluir D y union $\hat{S}_Y \cup \hat{S}_D$ como controles

Post-Double-Selection (Belloni et al.)

Problema: Con muchas variables, ¿cuales incluir como controles?

Estrategia de dos pasos:

1. **Primera seleccion:** Usar Lasso para regresar Y en X
   - Seleccionar variables $\hat{S}_Y$
2. **Segunda seleccion:** Usar Lasso para regresar D en X
   - Seleccionar variables $\hat{S}_D$
3. **OLS final:** Incluir D y union $\hat{S}_Y \cup \hat{S}_D$ como controles

Ventaja sobre Lasso directo:

- Incluye variables que predicen D (confounders) aunque no predigan Y bien
- Inferencia valida (bajo supuestos)

Comparacion Rapida

Metodo	Estimando	ML usado	Inferencia	Ventaja
DML	ATE	Cualquiera	Si	Flexible
Causal Forest	CATE $\tau(x)$	Arboles mod.	Si	Heterogeneidad
Post-Dbl-Selection	ATE	Lasso	Si	Simple

Regla practica:

- ATE con controles flexibles: DML
- Efectos heterogeneos: Causal Forests
- Seleccion de controles: Post-Double-Selection

Parte 4: Model Selection Profundo

Estrategias de Cross-Validation

1. **Leave-One-Out CV (LOOCV):**
   - $K = n$ (cada observacion es un fold)
   - Menos sesgo pero mas varianza y costo computacional
   - Formula cerrada para algunos modelos (Ridge, splines)
2. **Repeated K-fold CV:**
   - Repetir K-fold multiples veces con diferentes splits
   - Reduce varianza de estimacion de error
   - Recomendado cuando n es moderado
3. **Nested CV:**
   - CV externo para evaluar, CV interno para tuning
   - Evita sesgo optimista por "snooping"

Nested Cross-Validation

Problema: Si usas CV para elegir λ Y evaluar rendimiento con los mismos folds, eres demasiado optimista.

Solucion: Nested CV

Loop externo (evaluacion):

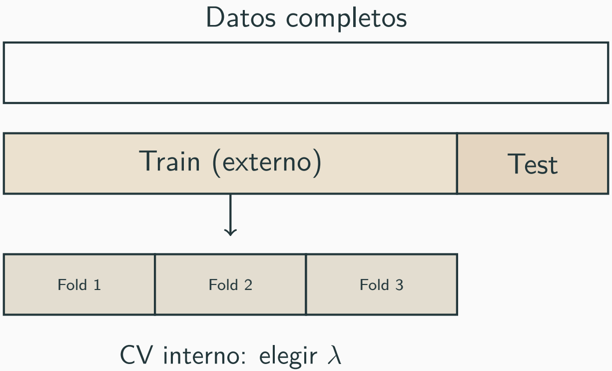
- Dividir en train/test

Loop interno (tuning):

- Dentro de train, hacer CV para elegir λ
- Entrenar modelo final con λ elegido
- Evaluar en test (externo)

Resultado: Estimacion no sesgada del error de generalizacion.

Ilustracion de Nested CV



Comparando Modelos No Anidados

Pregunta: ¿Ridge es mejor que Lasso? ¿RF es mejor que XGBoost?

No podemos usar test F. ¿Qué hacer?

Comparando Modelos No Anidados

Pregunta: ¿Ridge es mejor que Lasso? ¿RF es mejor que XGBoost?

No podemos usar test F. ¿Qué hacer?

1. **Cross-validation + test de significancia**
   - Comparar errores CV con test pareado (Dietterich 1998)
   - Toma en cuenta correlacion entre folds
2. **Information Criteria (AIC/BIC)**
   - Para modelos con misma verosimilitud
   - Penalizacion por complejidad
3. **Out-of-sample testing**
   - Dividir datos en train/validation/test
   - Comparar MSE en test

Test de Comparacion de Modelos

Setup: Dos modelos A y B, queremos probar $H_0 : \text{MSE}_A = \text{MSE}_B$

Procedimiento (Dietterich test):

1. Hacer K-fold CV para ambos modelos
2. Computar diferencia de errores en cada fold: $d_k = \text{MSE}_{A,k} - \text{MSE}_{B,k}$
3. Test t pareado:

$$t = \frac{\bar{d}}{\text{SE}(\bar{d})} \sim t_{K-1}$$

Test de Comparacion de Modelos

Setup: Dos modelos A y B, queremos probar $H_0 : \text{MSE}_A = \text{MSE}_B$

Procedimiento (Dietterich test):

1. Hacer K-fold CV para ambos modelos
2. Computar diferencia de errores en cada fold: $d_k = \text{MSE}_{A,k} - \text{MSE}_{B,k}$
3. Test t pareado:

$$t = \frac{\bar{d}}{\text{SE}(\bar{d})} \sim t_{K-1}$$

Advertencia:

- Los folds NO son independientes (comparten datos)
- Test puede ser anticonservador
- Mejor usar validacion cruzada repetida

El Trade-off Prediccion-Inferencia

Observacion importante:

Prediccion vs Inferencia

- **Prediccion**: Minimizar MSE de $\hat{Y}$
- **Inferencia**: Estimar θ con sesgo minimo y SE validos

El Trade-off Prediccion-Inferencia

Observacion importante:

Prediccion vs Inferencia

- **Prediccion**: Minimizar MSE de $\hat{Y}$
- **Inferencia**: Estimar θ con sesgo minimo y SE validos

Estos objetivos pueden estar en tension:

- Ridge mejora prediccion pero sesga $\hat{\theta}$
- RF predice muy bien pero no da inferencia sobre parametros
- OLS no predice tan bien pero tiene inferencia standard

El Trade-off Prediccion-Inferencia

Observacion importante:

Prediccion vs Inferencia

- **Prediccion**: Minimizar MSE de $\hat{Y}$
- **Inferencia**: Estimar θ con sesgo minimo y SE validos

Estos objetivos pueden estar en tension:

- Ridge mejora prediccion pero sesga $\hat{\theta}$
- RF predice muy bien pero no da inferencia sobre parametros
- OLS no predice tan bien pero tiene inferencia standard

Implicacion: El "mejor" modelo depende del objetivo.

Estrategia Recomendada

Para proyectos de investigacion:

1. **Definir objetivo claramente:**
   - ¿Predecir Y? â†' Usar CV, elegir modelo con menor MSE
   - ¿Estimar efecto causal? â†' Cuidado con regularizacion, usar DML
2. **Reportar multiple metricas:**
   - MSE in-sample y out-of-sample
   - R-squared (si es relevante)
   - Para causales: estimacion puntual, SE, p-value
3. **Robustez:**
   - Probar multiples especificaciones
   - Reportar rango de estimaciones
   - Discutir sensibilidad a supuestos

Checklist: ML en Economia Aplicada

Antes de empezar:

- ☐ ¿Tu pregunta es causal o predictiva?
- ☐ ¿Tienes suficientes datos para validacion out-of-sample?
- ☐ ¿Las variables predictoras estan disponibles en aplicacion real?

Durante analisis:

- ☐ Separar train/validation/test apropiadamente
- ☐ Usar nested CV si estas tuneando hiperparametros
- ☐ Verificar supuestos (si es causal)

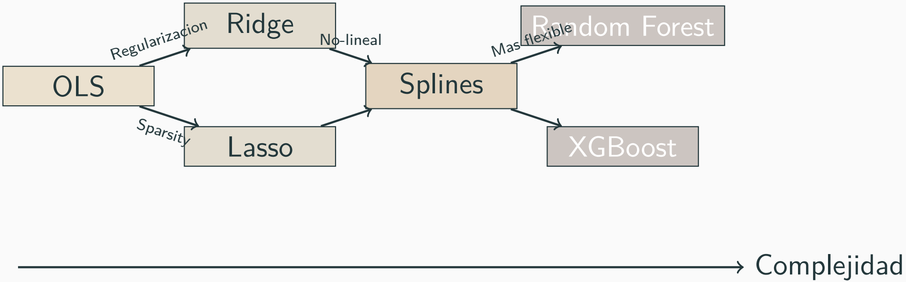
Al reportar:

- ☐ Reportar rendimiento out-of-sample
- ☐ Describir proceso de seleccion de modelo
- ☐ Discutir limitaciones y robustez

Parte 5: Sintesis e Integracion

Mapa del Curso: Prediccion

Objetivo: Minimizar MSE de $\hat{Y}$

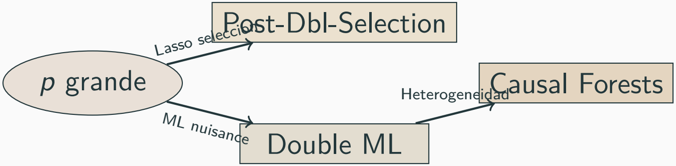


Izquierda: Interpretable

Derecha: Mejor prediccion

Mapa del Curso: Inferencia Causal

Objetivo: Estimar θ (efecto causal)



Todos: ML para prediccion
Inferencia separada

Mensaje clave: Separar prediccion (ML) de estimacion causal.

Lecciones Clave del Curso

1. **El trade-off sesgo-varianza es fundamental**
   - Complejidad vs generalizacion
   - Regularizacion introduce sesgo para reducir varianza
2. **Separar prediccion de inferencia**
   - ML excelente para prediccion, no para efectos causales
3. **No hay “mejor” metodo universal**
   - Depende de objetivo, datos, estructura del problema
4. **Interpretabilidad vs rendimiento**
   - OLS/Lasso: interpretables
   - RF/XGBoost: mejores predicciones, menos transparentes
   - DML: mejor de ambos mundos

Errores Comunes a Evitar

1. **Sobreajuste por evaluacion repetida**
   - NO: Ajustar multiples modelos, reportar solo el mejor
   - SÍ: Pre-registrar analisis o usar nested CV
2. **Confundir correlacion con causacion**
   - NO: "RF predice bien, por tanto X causa Y"
   - SÍ: Prediction $\neq$ causation sin identificacion
3. **Ignorar incertidumbre**
   - NO: Reportar solo predicciones puntuales
   - SÍ: Incluir intervalos de confianza/prediccion

Recursos para Profundizar

Libros:

- James et al. (2021): *Introduction to Statistical Learning* (2nd ed)
- Hastie et al. (2009): *Elements of Statistical Learning*
- Molnar (2022): *Interpretable Machine Learning*

Aplicaciones en Economia

Donde ML ha tenido impacto:

1. **Prediccion de politicas**
   - Targeting de programas sociales (Kleinberg et al. 2015)
   - Prediccion de riesgo de desempleo (Athey 2018)
2. **Efectos heterogeneos**
   - Retornos a educacion por tipo de estudiante
   - Efectos de salario minimo por region/industria
3. **Nowcasting**
   - Prediccion de GDP en tiempo real
   - Indices de actividad economica