

Inferencia Causal con Datos Observacionales

Una guía práctica para estudios de salud y ciencias sociales

Edgar Muñoz, Stat, MSc Epi, BMDSc

2025-12-31

Tabla de contenidos

Prefacio	7
¿Para quién es este libro?	7
Prerrequisitos	7
Estructura del libro	7
Software	8
Agradecimientos	8
Sobre el autor	8
Licencia	8
1 Introducción	9
1.1 ¿Qué es la inferencia causal?	9
1.2 Correlación no implica causalidad	9
1.3 El problema fundamental de la inferencia causal	9
1.4 Soluciones al problema	10
1.4.1 Experimentos aleatorizados	10
1.4.2 Estudios observacionales	10
1.5 El enfoque de este libro	11
1.6 Objetivos de aprendizaje	12
Referencias	12
I Parte I: Fundamentos	13
2 DAGs y Confusión	14
2.1 Objetivos de aprendizaje	14
2.2 ¿Qué es un DAG?	14
2.2.1 Componentes de un DAG	14
2.3 Causalidad vs Asociación	15
2.3.1 Asociación	15
2.3.2 Causalidad	16
2.4 Tipos de caminos en un DAG	16
2.4.1 Cadena (mediación)	16
2.4.2 Fork (confusión)	17
2.4.3 Collider (colisionador)	18

2.5	D-separación	19
2.5.1	Ejemplo	19
2.6	Identificación de confusores	21
2.7	Usando dagitty en R	21
2.8	Aplicación: Ejemplo con datos	22
2.9	Ejercicios	23
2.10	Resumen	23
	Referencias	23
3	Control de Confusión	24
3.1	Objetivos de aprendizaje	24
3.2	¿Qué es la confusión?	24
3.3	Métodos para controlar confusión	24
3.3.1	1. Estratificación	24
3.3.2	2. Matching (Emparejamiento)	25
3.3.3	3. Propensity Score	28
3.3.4	4. Ponderación (IPTW)	29
3.4	Comparación de métodos	31
3.5	Diagnósticos importantes	31
3.5.1	Balance de covariables	31
3.5.2	Distribución de pesos	32
3.6	Ejercicios	33
3.7	Resumen	33
	Referencias	33
4	Sesgo de Selección	34
4.1	Objetivos de aprendizaje	34
4.2	¿Qué es el sesgo de selección?	34
4.3	El sesgo de colisionador	34
4.3.1	Ejemplo: La paradoja del índice de colisión	35
4.4	Tipos de sesgo de selección	36
4.4.1	1. Sesgo de participación	36
4.4.2	2. Sesgo de pérdida de seguimiento	37
4.4.3	3. Sesgo del sobreviviente	38
4.4.4	4. Sesgo de Berkson	39
4.5	Estrategias de mitigación	40
4.5.1	1. Ponderación por probabilidad de selección	40
4.5.2	2. Análisis de sensibilidad	41
4.5.3	3. Diseño del estudio	41
4.6	Identificación usando DAGs	41
4.7	Ejercicios	43
4.8	Resumen	43
	Referencias	43

II Parte II: Métodos Avanzados	44
5 Variables Instrumentales	45
5.1 Objetivos de aprendizaje	45
5.2 ¿Qué es una variable instrumental?	45
5.3 Condiciones de validez	46
5.3.1 Verificación de relevancia	46
5.4 Estimación por 2SLS	48
5.4.1 Usando el paquete ivreg	48
5.5 Comparación con OLS	49
5.6 Efecto Local (LATE)	50
5.7 Ejemplos clásicos de instrumentos	51
5.8 Instrumentos débiles	51
5.9 Diagnósticos para IV	52
5.9.1 Test de instrumentos débiles	52
5.9.2 Test de sobreidentificación	53
5.10 Ejercicios	54
5.11 Resumen	54
Referencias	54
6 Causalidad y Supervivencia	55
6.1 Objetivos de aprendizaje	55
6.2 Análisis de supervivencia: repaso	55
6.2.1 Conceptos fundamentales	55
6.3 Sesgos en supervivencia	56
6.3.1 Sesgo de tiempo inmortal	56
6.3.2 Ejemplo de sesgo de tiempo inmortal	57
6.3.3 Corrección con tiempo dependiente	58
6.4 Ponderación en supervivencia (IPTW)	59
6.5 Riesgos competitivos	60
6.5.1 Enfoque de subdistribución (Fine-Gray)	61
6.5.2 Interpretación causal	63
6.6 Análisis causa-específico	63
6.7 Ejercicios	63
6.8 Resumen	64
Referencias	64
7 Análisis de Sensibilidad	65
7.1 Objetivos de aprendizaje	65
7.2 ¿Por qué análisis de sensibilidad?	65
7.3 El E-value	66
7.3.1 Interpretación del E-value	67

7.4	Análisis de sensibilidad con sensemakr	68
7.4.1	Visualización de contornos	70
7.4.2	Robustness Value (RV)	70
7.5	Fórmula de sesgo de confusión	71
7.6	Análisis de sensibilidad para IPTW	71
7.7	Comunicando resultados de sensibilidad	73
7.7.1	Buenas prácticas	73
7.7.2	Ejemplo de reporte	73
7.8	Ejercicios	74
7.9	Resumen	75
	Referencias	75
8	Análisis de Mediación	76
8.1	Objetivos de aprendizaje	76
8.2	¿Qué es la mediación?	76
8.3	Definiciones causales	77
8.3.1	Efectos naturales (NDE y NIE)	77
8.4	Asunciones para identificación	79
8.5	Método de Baron-Kenny (tradicional)	80
8.5.1	Limitaciones del método tradicional	81
8.6	Método de mediación causal	81
8.6.1	Interpretación de resultados	82
8.7	Mediación con resultados binarios	83
8.8	Análisis de sensibilidad para mediación	85
8.9	Múltiples mediadores	86
8.10	Ejercicios	87
8.11	Resumen	88
	Referencias	88
III	Parte III: Aplicación	89
9	Aplicación Integradora	90
9.1	Objetivos de aprendizaje	90
9.2	Caso de estudio: Efecto del ejercicio en la salud cardiovascular	90
9.2.1	Pregunta de investigación	90
9.2.2	Datos	90
9.3	Paso 1: Especificación del DAG	92
9.3.1	Identificación de confusores	93
9.4	Paso 2: Análisis descriptivo	94
9.5	Paso 3: Estimación por múltiples métodos	95
9.5.1	3.1 Análisis crudo	95
9.5.2	3.2 Regresión multivariable	96

9.5.3 3.3 Propensity Score Matching	96
9.5.4 3.4 IPTW	98
9.5.5 Visualización de balance	99
9.6 Paso 4: Análisis de mediación	100
9.7 Paso 5: Análisis de sensibilidad	101
9.8 Paso 6: Resumen de resultados	103
9.9 Conclusiones del caso de estudio	105
9.10 Lista de verificación para análisis causal	107
Referencias	107
Referencias	108

Prefacio

Este libro presenta los métodos modernos de **inferencia causal** aplicados a estudios observacionales en salud y ciencias sociales. El contenido se basa en mis notas sobre el tema a lo largo de varios proyectos en los últimos años y más recientemente en las notas de preparación del material para un curso dictado en la [Universidad ICESI](#), Cali, Colombia.

¿Para quién es este libro?

Este material está diseñado para:

- Estudiantes de posgrado en epidemiología, bioestadística y salud pública
- Investigadores que trabajan con datos observacionales
- Profesionales interesados en métodos causales modernos

Prerrequisitos

Se asume conocimiento previo de:

- Estadística básica (probabilidad, distribuciones, pruebas de hipótesis)
- Regresión lineal y logística
- Fundamentos de epidemiología
- Programación básica en R

Estructura del libro

El libro está organizado en tres partes:

1. **Fundamentos:** Diagramas causales (DAGs), confusión y sesgo de selección
2. **Métodos Avanzados:** Variables instrumentales, supervivencia, sensibilidad y mediación
3. **Aplicación:** Caso práctico integrando los métodos del curso

Software

Los ejemplos de código utilizan R con los siguientes paquetes principales:

```
# Paquetes principales
install.packages(c(
  "dagitty",      # DAGs
  "ggdag",        # Visualización de DAGs
  "MatchIt",      # Matching
  "WeightIt",     # Ponderación
  "survival",     # Análisis de supervivencia
  "mediation",   # Análisis de mediación
  "sensemakr",   # Análisis de sensibilidad
  "tidyverse"     # Manipulación de datos
))
```

Agradecimientos

Agradezco a la Universidad ICESI por la oportunidad de desarrollar este curso, y a los estudiantes cuyas preguntas y comentarios han mejorado este material.

Sobre el autor

Edgar Muñoz es biomedical data scientist, estadístico y epidemiólogo. Actualmente es data scientist en el University of Texas Health Science Center at San Antonio, con más de 25 años de experiencia en investigación en salud pública. Sus áreas de especialización incluyen la inferencia causal, la epidemiología espacial, los modelos multinivel, las disparidades en el cáncer y la inteligencia artificial aplicada a la salud.

- [ORCID](#)
- [Google Scholar](#)
- [Sitio web](#)

Licencia

Este trabajo está bajo licencia [Creative Commons Attribution-NonCommercial-ShareAlike 4.0](#).

1 Introducción

1.1 ¿Qué es la inferencia causal?

La **inferencia causal** es el proceso de determinar si una relación entre variables es causal o meramente asociativa. En otras palabras, queremos saber si un cambio en una variable *causa* un cambio en otra.

Definición

La inferencia causal busca responder la pregunta: “*¿Qué pasaría si...?*” — una pregunta contrafactual que va más allá de la simple asociación estadística.

1.2 Correlación no implica causalidad

Este principio fundamental de la estadística nos recuerda que observar una asociación entre dos variables no significa que una cause la otra. Considera estos ejemplos:

1. **Helados y ahogamientos:** Las ventas de helado están correlacionadas con las muertes por ahogamiento. ¿Los helados causan ahogamientos? No — ambos aumentan en verano.
2. **Cigüeñas y nacimientos:** En algunas regiones europeas, el número de cigüeñas está correlacionado con la tasa de natalidad. ¿Las cigüeñas traen bebés? No — ambos están asociados con áreas rurales.

1.3 El problema fundamental de la inferencia causal

El **problema fundamental de la inferencia causal** es que no podemos observar el mismo individuo bajo dos condiciones diferentes al mismo tiempo. Si una persona recibe un tratamiento, no podemos saber qué hubiera pasado si no lo hubiera recibido.

```

flowchart LR
    A[Individuo] --> B{Tratamiento}
    B -->|Sí| C[Resultado observado]
    B -->|No| D[Resultado contrafactual]
    style D stroke-dasharray: 5 5

```

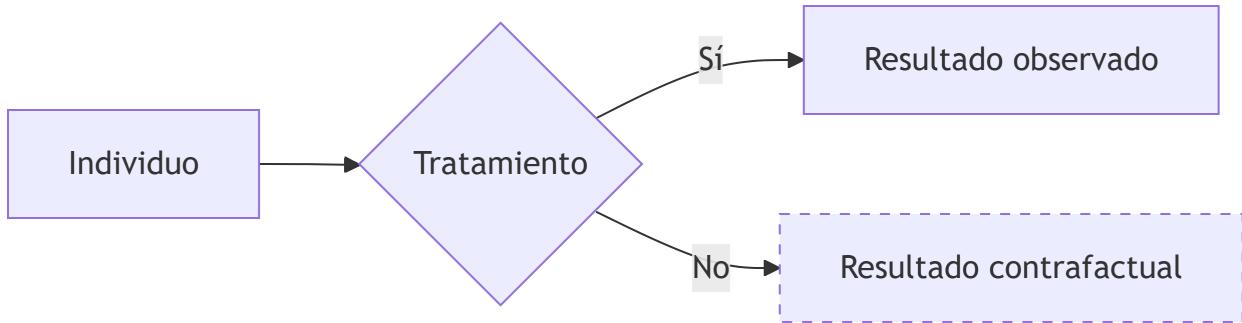


Figura 1.1: El contrafactual nunca se observa

1.4 Soluciones al problema

A lo largo de la historia, se han desarrollado diferentes enfoques para abordar este problema:

1.4.1 Experimentos aleatorizados

El **ensayo clínico aleatorizado** (RCT) es el estándar de oro porque:

- La aleatorización crea grupos comparables
- Elimina el sesgo de confusión
- Permite estimar el efecto causal promedio

Sin embargo, los RCTs no siempre son: - Éticos (no podemos asignar exposiciones dañinas) - Factibles (alto costo, tiempo) - Generalizables (poblaciones seleccionadas)

1.4.2 Estudios observacionales

Los **estudios observacionales** son frecuentemente la única opción disponible. Para hacer inferencias causales válidas, necesitamos:

1. **Identificar** las fuentes de sesgo

2. **Controlar** la confusión
3. **Evaluar** la sensibilidad de los resultados

1.5 El enfoque de este libro

Este libro adopta un enfoque basado en **Diagramas Acíclicos Dirigidos (DAGs)**, desarrollado principalmente por Judea Pearl (Pearl 2009). Este enfoque nos permite:

- Representar visualmente nuestras asunciones causales
- Identificar sistemáticamente las fuentes de sesgo
- Determinar qué variables debemos (y no debemos) controlar

```
library(ggdag)
library(ggplot2)

dag <- dagify(
  Y ~ X + C,
  X ~ C,
  exposure = "X",
  outcome = "Y",
  labels = c(
    Y = "Resultado",
    X = "Exposición",
    C = "Confusor"
  ),
  coords = list(
    x = c(X = 0, Y = 2, C = 1),
    y = c(X = 0, Y = 0, C = 1)
  )
)

ggdag(dag, text = FALSE, use_labels = "label") +
  theme_dag() +
  labs(title = "Confusión clásica")
```

Confusión clásica

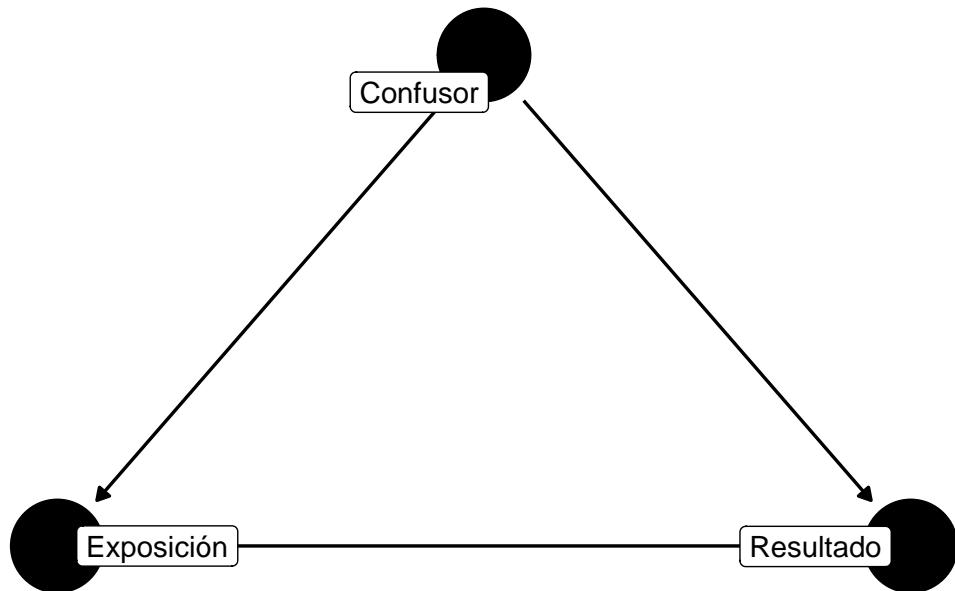


Figura 1.2: Un DAG simple mostrando confusión

1.6 Objetivos de aprendizaje

Al finalizar este libro, serás capaz de:

1. Construir e interpretar DAGs para problemas de investigación
2. Identificar y controlar fuentes de sesgo en estudios observacionales
3. Aplicar métodos de estratificación, matching y ponderación
4. Reconocer y manejar el sesgo de selección
5. Usar variables instrumentales cuando estén disponibles
6. Realizar análisis de supervivencia con enfoque causal
7. Evaluar la sensibilidad de tus resultados
8. Conducir análisis de mediación causal

Referencias

Part I

Parte I: Fundamentos

2 DAGs y Confusión

2.1 Objetivos de aprendizaje

Al finalizar este capítulo, serás capaz de:

- Definir y construir Diagramas Acíclicos Dirigidos (DAGs)
- Distinguir entre causalidad y asociación
- Identificar confusores usando DAGs
- Aplicar las reglas de d-separación

2.2 ¿Qué es un DAG?

Un **Diagrama Acíclico Dirigido** (DAG, por sus siglas en inglés) es una representación gráfica de las relaciones causales entre variables.

Definición

Un DAG es un grafo donde:

- **Dirigido:** Las flechas indican la dirección de la causalidad
- **Acíclico:** No hay ciclos (no puedes volver a una variable siguiendo las flechas)

2.2.1 Componentes de un DAG

```
library(ggdag)
library(ggplot2)

# DAG simple
dag <- dagify(
  Y ~ X + C,
  X ~ C,
  coords = list(
```

```

    x = c(X = 0, Y = 2, C = 1),
    y = c(X = 0, Y = 0, C = 1)
  )
)

ggdag(dag) +
  theme_dag() +
  labs(title = "DAG con exposición (X), resultado (Y) y confusor (C)")

```

DAG con exposición (X), resultado (Y) y confusor (C)

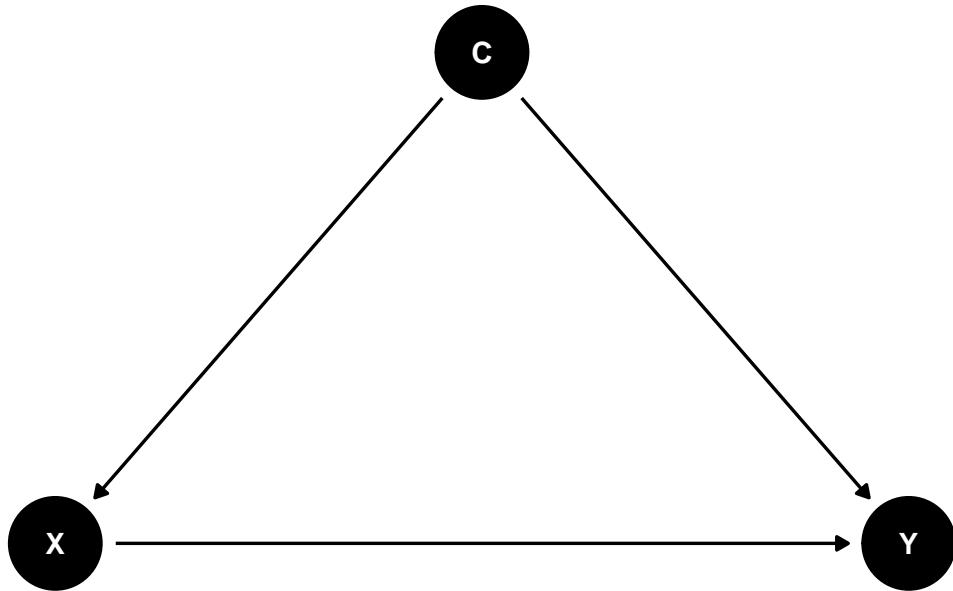


Figura 2.1: Componentes básicos de un DAG

2.3 Causalidad vs Asociación

2.3.1 Asociación

Dos variables están **asociadas** si conocer el valor de una proporciona información sobre el valor de la otra.

$$P(Y|X) \neq P(Y)$$

2.3.2 Causalidad

Una variable **causa** otra si intervenir sobre la primera cambia la distribución de la segunda.

$$P(Y|do(X)) \neq P(Y)$$

La notación $do(X)$ representa una intervención, no una observación.

2.4 Tipos de caminos en un DAG

2.4.1 Cadena (mediación)

```
chain <- dagify(
  M ~ X,
  Y ~ M,
  coords = list(
    x = c(X = 0, M = 1, Y = 2),
    y = c(X = 0, M = 0, Y = 0)
  )
)

ggdag(chain) +
  theme_dag() +
  labs(title = "Cadena (Mediación)")
```

Cadena (Mediación)



Figura 2.2: Cadena: $X \rightarrow M \rightarrow Y$

En una cadena, X causa Y *a través de* M. Si controlamos por M, bloqueamos el efecto.

2.4.2 Fork (confusión)

```
fork <- dagify(
  X ~ C,
  Y ~ C,
  coords = list(
    x = c(X = 0, C = 1, Y = 2),
    y = c(X = 0, C = 1, Y = 0)
  )
)

ggdag(fork) +
  theme_dag() +
  labs(title = "Fork (Confusión)")
```

Fork (Confusión)

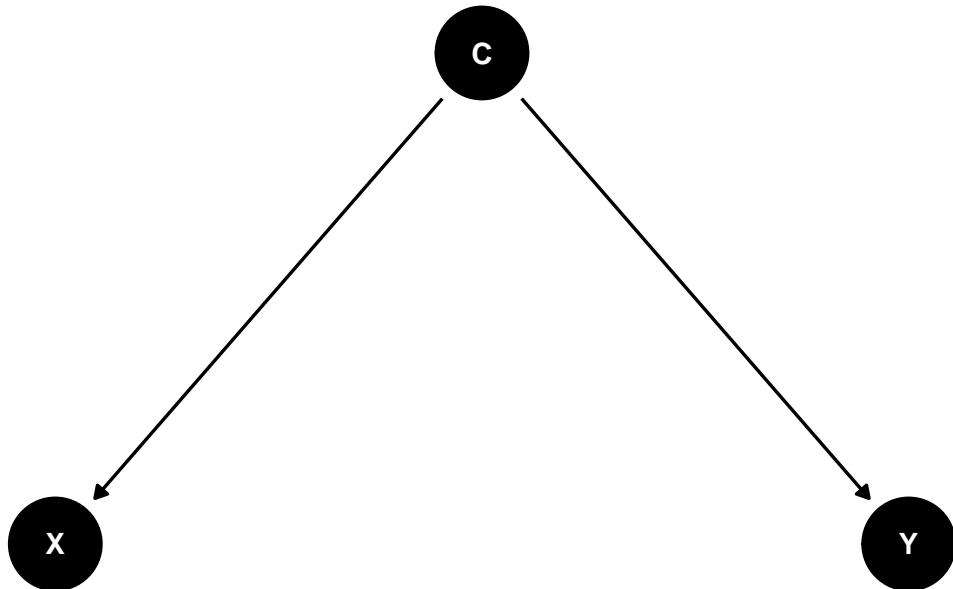


Figura 2.3: Fork: $X \leftarrow C \rightarrow Y$

En un fork, C es causa común de X e Y. X e Y están asociadas pero X no causa Y.

2.4.3 Collider (colisionador)

```
collider <- dagify(
  M ~ X + Y,
  coords = list(
    x = c(X = 0, M = 1, Y = 2),
    y = c(X = 0, M = -0.5, Y = 0)
  )
)

ggdag(collider) +
  theme_dag() +
  labs(title = "Collider")
```

Collider

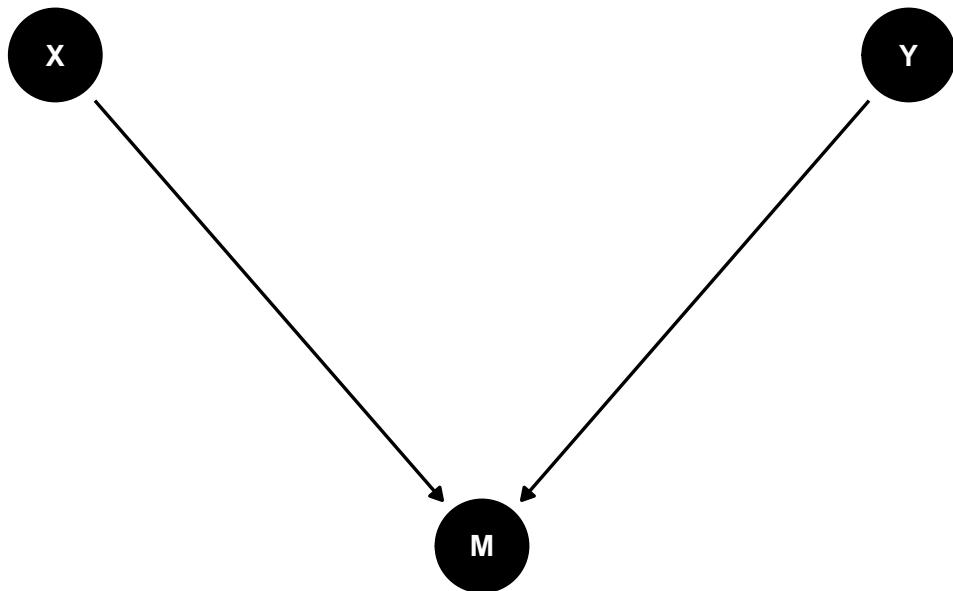


Figura 2.4: Collider: $X \rightarrow M \leftarrow Y$

En un collider, M es efecto de X e Y. X e Y no están asociadas, pero **se vuelven asociadas** si controlamos por M.

2.5 D-separación

La **d-separación** es un criterio para determinar si dos variables son independientes dado un conjunto de variables condicionantes.

! Reglas de d-separación

1. **Cadenas y forks** están bloqueados si condicionamos en el nodo intermedio
2. **Colliders** están bloqueados por defecto, pero se abren si condicionamos en ellos (o sus descendientes)

2.5.1 Ejemplo

```

example_dag <- dagify(
  Y ~ X + U,
  X ~ C,
  C ~ U,
  M ~ X + Y,
  exposure = "X",
  outcome = "Y",
  coords = list(
    x = c(X = 0, Y = 2, C = 0, U = 1, M = 1),
    y = c(X = 0, Y = 0, C = 1, U = 1, M = -1)
  )
)

ggdag_dseparated(example_dag, from = "X", to = "Y") +
  theme_dag() +
  labs(title = "¿X y Y son d-separadas?")

```

¿X y Y son d-separadas?

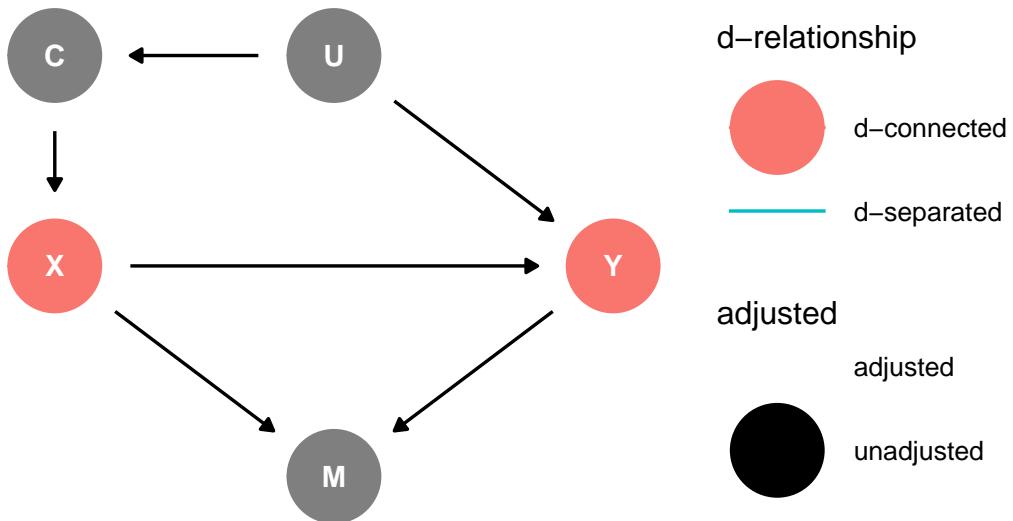


Figura 2.5: Ejemplo de d-separación

2.6 Identificación de confusores

Un **confusor** es una variable que:

1. Causa (o está asociada con) la exposición
2. Causa (o está asociada con) el resultado
3. No está en el camino causal de X a Y

```
library(dagitty)

g <- dagitty("dag {
  X -> Y
  C -> X
  C -> Y
}")

adjustmentSets(g, exposure = "X", outcome = "Y")
{ C }
```

2.7 Usando dagitty en R

El paquete **dagitty** permite:

1. Definir DAGs
2. Encontrar conjuntos de ajuste
3. Verificar d-separación

```
library(dagitty)

# Definir DAG
mi_dag <- dagitty("dag {
  Tratamiento -> Resultado
  Edad -> Tratamiento
  Edad -> Resultado
  Sexo -> Resultado
}")

# Encontrar variables de ajuste
adjustmentSets(mi_dag,
```

```
    exposure = "Tratamiento",
    outcome = "Resultado")
```

```
{ Edad }
```

2.8 Aplicación: Ejemplo con datos

Consideremos un estudio sobre el efecto del ejercicio en la presión arterial:

```
# Simular datos
set.seed(42)
n <- 500

edad <- rnorm(n, 50, 10)
ejercicio <- 0.5 * edad + rnorm(n, 0, 5) # Edad afecta ejercicio
presion <- 100 + 0.5 * edad - 0.3 * ejercicio + rnorm(n, 0, 10)

datos <- data.frame(edad, ejercicio, presion)

# Sin ajustar por edad (sesgado)
modelo_crudo <- lm(presion ~ ejercicio, data = datos)

# Ajustando por edad
modelo_ajustado <- lm(presion ~ ejercicio + edad, data = datos)

# Comparar
cat("Efecto crudo:", round(coef(modelo_crudo)[2], 3), "\n")
```

Efecto crudo: 0.208

```
cat("Efecto ajustado:", round(coef(modelo_ajustado)[2], 3), "\n")
```

Efecto ajustado: -0.298

2.9 Ejercicios

💡 Ejercicio 1

Construye un DAG para el siguiente escenario: Queremos estudiar el efecto del consumo de café en enfermedades cardíacas. Sabemos que el tabaquismo está asociado tanto con el consumo de café como con las enfermedades cardíacas.

💡 Ejercicio 2

Usando el paquete `dagitty`, determina el conjunto mínimo de variables que debes controlar para estimar el efecto causal.

2.10 Resumen

- Los DAGs son herramientas gráficas para representar asunciones causales
- Existen tres estructuras básicas: cadenas, forks y colliders
- La d-separación nos ayuda a identificar qué variables controlar
- Controlar por un collider introduce sesgo
- El paquete `dagitty` facilita el trabajo con DAGs en R

Referencias

3 Control de Confusión

3.1 Objetivos de aprendizaje

Al finalizar este capítulo, serás capaz de:

- Definir confusión y sus consecuencias
- Aplicar métodos de estratificación
- Implementar matching (emparejamiento)
- Utilizar ponderación por propensity score

3.2 ¿Qué es la confusión?

La **confusión** ocurre cuando una variable externa está asociada tanto con la exposición como con el resultado, distorsionando la estimación del efecto causal.

⚠️ Consecuencia

Sin control adecuado de la confusión, podemos:

- Encontrar una asociación donde no existe efecto causal
- No detectar un efecto causal real
- Subestimar o sobreestimar la magnitud del efecto

3.3 Métodos para controlar confusión

3.3.1 1. Estratificación

La **estratificación** divide los datos en grupos homogéneos según el confusor.

```

# Simular datos
set.seed(123)
n <- 1000

# Confusor: edad (0 = joven, 1 = mayor)
edad <- rbinom(n, 1, 0.5)

# Exposición influenciada por edad
tratamiento <- rbinom(n, 1, 0.3 + 0.4 * edad)

# Resultado influenciado por ambos
resultado <- 50 + 10 * edad + 5 * tratamiento + rnorm(n, 0, 5)

datos <- data.frame(edad, tratamiento, resultado)

# Efecto crudo
cat("Efecto crudo:",
    round(mean(resultado[tratamiento == 1]) -
        mean(resultado[tratamiento == 0]), 2), "\n")

```

Efecto crudo: 9.01

```

# Efecto estratificado
efecto_jovenes <- mean(resultado[tratamiento == 1 & edad == 0]) -
    mean(resultado[tratamiento == 0 & edad == 0])
efecto_mayores <- mean(resultado[tratamiento == 1 & edad == 1]) -
    mean(resultado[tratamiento == 0 & edad == 1])

cat("Efecto en jóvenes:", round(efecto_jovenes, 2), "\n")

```

Efecto en jóvenes: 4.7

```
cat("Efecto en mayores:", round(efecto_mayores, 2), "\n")
```

Efecto en mayores: 5.17

3.3.2 2. Matching (Emparejamiento)

El **matching** empareja individuos tratados y no tratados con características similares.

```

library(MatchIt)

# Datos más complejos
set.seed(456)
n <- 500

datos_match <- data.frame(
  edad = rnorm(n, 50, 10),
  imc = rnorm(n, 25, 5),
  sexo = rbinom(n, 1, 0.5)
)

# Probabilidad de tratamiento
prob_trat <- plogis(-2 + 0.05 * datos_match$edad +
                      0.1 * datos_match$imc)
datos_match$tratamiento <- rbinom(n, 1, prob_trat)

# Resultado
datos_match$resultado <- 100 +
  0.5 * datos_match$edad +
  2 * datos_match$imc -
  5 * datos_match$tratamiento +
  rnorm(n, 0, 10)

# Matching
match_out <- matchit(tratamiento ~ edad + imc + sexo,
                      data = datos_match,
                      method = "nearest",
                      ratio = 1)

summary(match_out)

```

Call:

```
matchit(formula = tratamiento ~ edad + imc + sexo, data = datos_match,
        method = "nearest", ratio = 1)
```

Summary of Balance for All Data:

	Means Treated	Means Control	Std. Mean Diff.	Var. Ratio	eCDF	Mean
distance	0.9316	0.8815	0.9576	0.3455	0.2036	
edad	51.4860	44.4794	0.7323	1.0378	0.1903	
imc	25.1570	24.0066	0.2318	1.1417	0.0639	

sexo	0.4871	0.4722	0.0297	.	0.0148
	eCDF Max				
distance	0.3376				
edad	0.3103				
imc	0.1573				
sexo	0.0148				

Summary of Balance for Matched Data:

	Means	Treated	Means	Control	Std. Mean	Diff.	Var.	Ratio	eCDF	Mean
distance	0.9887		0.8815		2.0500	0.0010		0.6529		
edad	68.4310		44.4794		2.5033	0.2193		0.6264		
imc	28.4034		24.0066		0.8859	0.9017		0.2409		
sexo	0.5556		0.4722		0.1667	.		0.0833		
	eCDF	Max	Std.	Pair	Dist.					
distance	1.0000			2.0500						
edad	0.9722			2.5033						
imc	0.3889			1.3458						
sexo	0.0833			0.8336						

Sample Sizes:

	Control	Treated
All	36	464
Matched	36	36
Unmatched	0	428
Discarded	0	0

3.3.2.1 Balance después del matching

```
library(ggplot2)

plot(match_out, type = "jitter", interactive = FALSE)
```

Distribution of Propensity Scores

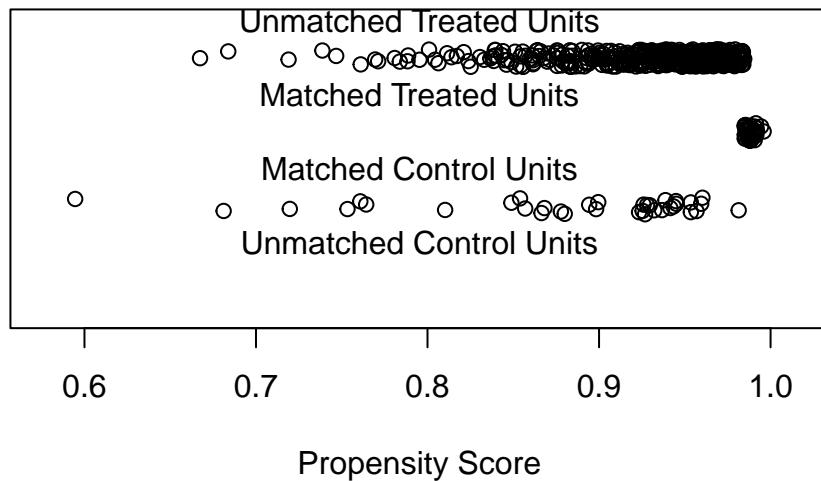


Figura 3.1: Balance de covariables antes y después del matching

3.3.2.2 Estimación del efecto

```
# Obtener datos emparejados
datos_emparejados <- match.data(match_out)

# Efecto en datos emparejados
modelo_match <- lm(resultado ~ tratamiento,
                     data = datos_emparejados,
                     weights = weights)

summary(modelo_match)$coefficients["tratamiento", ]
```

Estimate	Std. Error	t value	Pr(> t)
1.784918e+01	3.044303e+00	5.863141e+00	1.364022e-07

3.3.3 3. Propensity Score

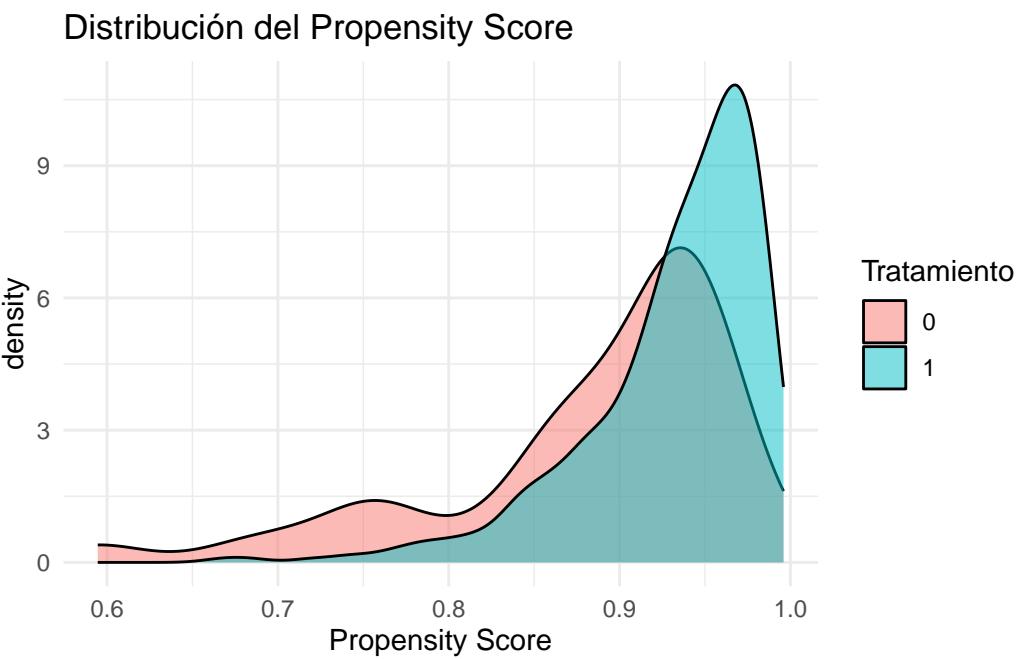
El **propensity score** es la probabilidad de recibir el tratamiento dado las covariables.

$$e(X) = P(T = 1|X)$$

```
# Estimar propensity score
ps_model <- glm(tratamiento ~ edad + imc + sexo,
                 data = datos_match,
                 family = binomial)

datos_match$ps <- predict(ps_model, type = "response")

# Visualizar distribución
ggplot(datos_match, aes(x = ps, fill = factor(tratamiento))) +
  geom_density(alpha = 0.5) +
  labs(title = "Distribución del Propensity Score",
       x = "Propensity Score",
       fill = "Tratamiento") +
  theme_minimal()
```



3.3.4 4. Ponderación (IPTW)

La **ponderación por el inverso del propensity score** (IPTW) crea una pseudo-población donde tratamiento y covariables son independientes.

$$w_i = \frac{T_i}{e(X_i)} + \frac{1 - T_i}{1 - e(X_i)}$$

```
library(WeightIt)

# Calcular pesos
pesos <- weightit(tratamiento ~ edad + imc + sexo,
                     data = datos_match,
                     method = "ps",
                     estimand = "ATE")

summary(pesos)
```

Summary of weights

- Weight ranges:

	Min	Max
treated	1.004	1.498
control	2.467	53.778

- Units with the 5 most extreme weights by group:

	450	253	207	60	33
treated	1.339	1.354	1.391	1.462	1.498
	27	24	21	14	13
control	21.514	23.175	24.725	25.17	53.778

- Weight statistics:

	Coef	of	Var	MAD	Entropy	# Zeros
treated	0.063	0.045	0.002			0
control	0.715	0.492	0.208			0

- Effective Sample Sizes:

	Control	Treated
Unweighted	36.	464.
Weighted	24.04	462.18

```
# Modelo ponderado
modelo_iptw <- lm(resultado ~ tratamiento,
                     data = datos_match,
                     weights = pesos$weights)

summary(modelo_iptw)$coefficients["tratamiento", ]
```

```
Estimate Std. Error t value Pr(>|t|)
-2.38609550 1.27698752 -1.86853471 0.06227447
```

3.4 Comparación de métodos

Método	Ventajas	Desventajas
Estratificación	Simple, transparente	Solo para pocos confusores
Matching	Intuitivo, balance visible	Pérdida de muestra
PS Matching	Reduce dimensionalidad	Depende del modelo de PS
IPTW	Usa toda la muestra	Pesos extremos posibles

3.5 Diagnósticos importantes

3.5.1 Balance de covariables

```
library(cobalt)

love.plot(pesos,
          thresholds = c(m = 0.1),
          abs = TRUE,
          var.order = "unadjusted")
```

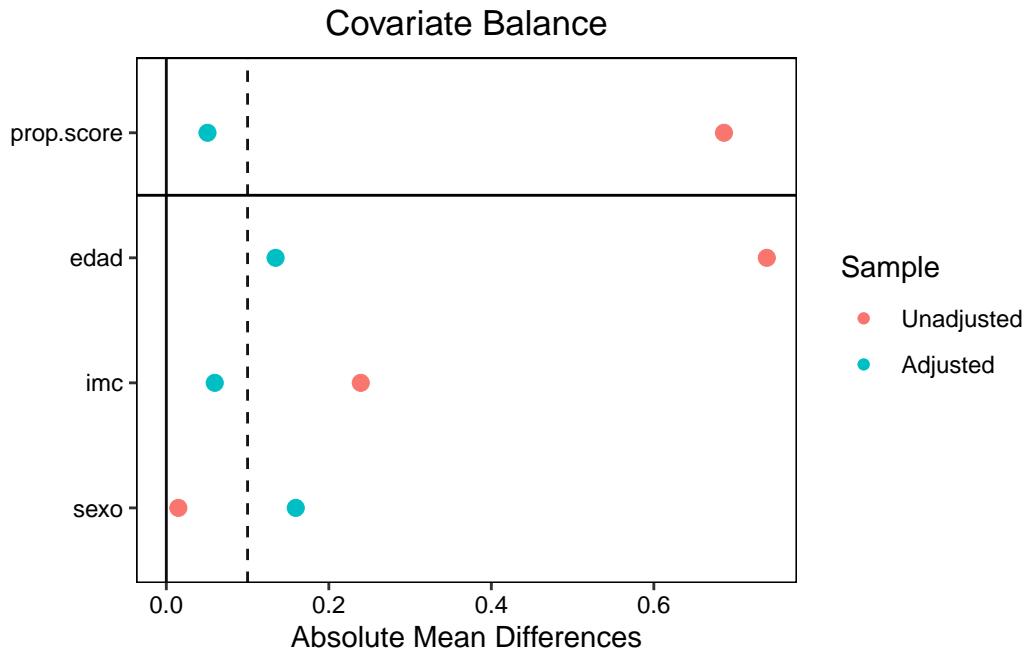


Figura 3.2: Love plot: comparación de balance

3.5.2 Distribución de pesos

```
summary(pesos$weights)
```

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
	1.004	1.033	1.063	1.955	1.119	53.778

```
# Pesos extremos pueden indicar violación de positividad
cat("Pesos > 10:", sum(pesos$weights > 10), "\n")
```

Pesos > 10: 19

3.6 Ejercicios

💡 Ejercicio 1

Usando el dataset simulado, compara los resultados de:

1. Análisis crudo
2. Matching por edad e IMC
3. IPTW

¿Cuál se acerca más al efecto verdadero (-5)?

💡 Ejercicio 2

Evalúa el impacto de diferentes métodos de matching (nearest, optimal, genetic) en el balance y la estimación del efecto.

3.7 Resumen

- La confusión distorsiona las estimaciones de efectos causales
- La estratificación es útil para pocos confusores
- El matching empareja individuos similares
- El propensity score resume múltiples confusores en un solo número
- IPTW permite usar toda la muestra
- Siempre verificar el balance después del ajuste

Referencias

4 Sesgo de Selección

4.1 Objetivos de aprendizaje

Al finalizar este capítulo, serás capaz de:

- Definir sesgo de selección desde una perspectiva causal
- Identificar el sesgo de colisionador
- Reconocer diferentes tipos de sesgo de selección
- Proponer estrategias para mitigar estos sesgos

4.2 ¿Qué es el sesgo de selección?

El **sesgo de selección** ocurre cuando la asociación entre exposición y resultado difiere entre los participantes del estudio y la población objetivo.

! Perspectiva causal

Desde el marco de DAGs, el sesgo de selección surge cuando condicionamos en un **collider** o en descendientes de un collider.

4.3 El sesgo de colisionador

Un **colisionador** es una variable que es efecto común de dos o más variables.

```
library(ggdag)
library(ggplot2)

collider_dag <- dagify(
  S ~ X + Y,
  coords = list(
    x = c(X = 0, Y = 2, S = 1),
    y = c(X = 0, Y = 0, S = -0.5)
```

```

),
labels = c(
  X = "Exposición",
  Y = "Resultado",
  S = "Selección"
)
)

ggdag_adjust(collider_dag, var = "S",
             text = FALSE, use_labels = "label") +
theme_dag() +
labs(title = "Condicionar en el colisionador induce sesgo")

```

Condicionar en el colisionador induce sesgo

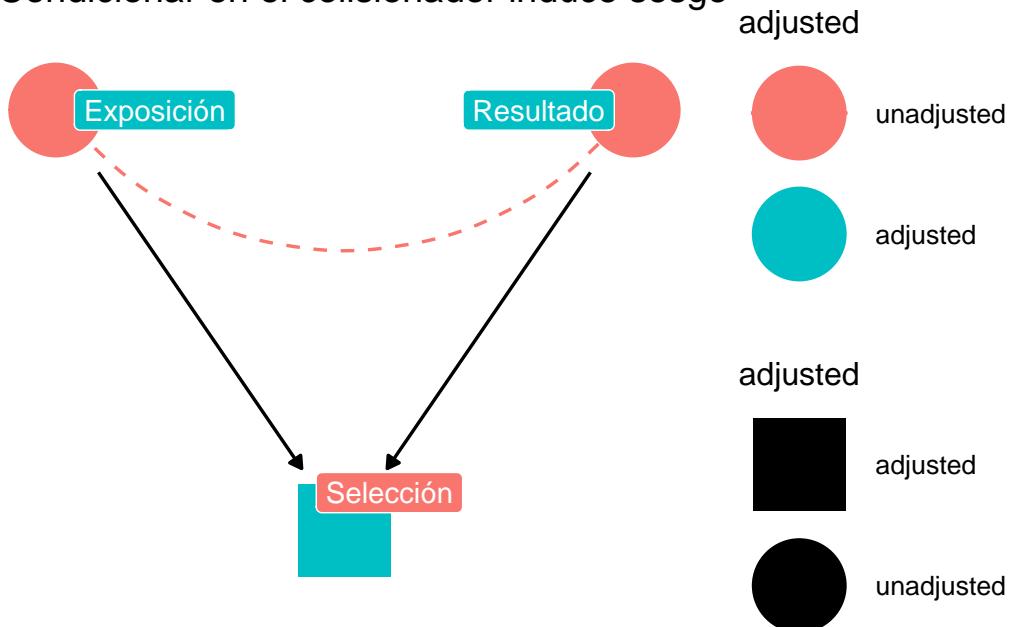


Figura 4.1: Sesgo de colisionador: condicionar en S abre el camino $X \rightarrow S \leftarrow Y$

4.3.1 Ejemplo: La paradoja del índice de colisión

```

set.seed(789)
n <- 10000

```

```
# Habilidad académica y habilidad deportiva (independientes)
academico <- rnorm(n, 0, 1)
deportivo <- rnorm(n, 0, 1)

# Admisión basada en ambas (collider)
admitido <- (academico + deportivo) > 1

# En la población general
cor(academico, deportivo)
```

[1] -0.01700133

```
# Entre los admitidos solamente
cor(academico[admitido], deportivo[admitido])
```

[1] -0.6322336

i Interpretación

Aunque las habilidades son independientes en la población, entre los admitidos aparece una correlación **negativa**. Esto es porque si una persona con alta habilidad académica fue admitida, no necesita alta habilidad deportiva (y viceversa).

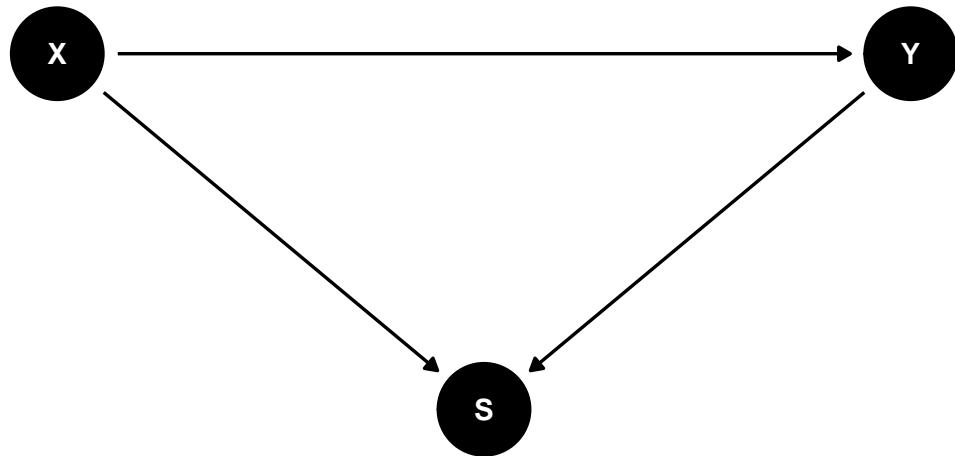
4.4 Tipos de sesgo de selección

4.4.1 1. Sesgo de participación

Occurre cuando la participación en el estudio depende de la exposición y el resultado.

```
part_dag <- dagify(
  Y ~ X,
  S ~ X + Y,
  coords = list(
    x = c(X = 0, Y = 2, S = 1),
    y = c(X = 0, Y = 0, S = -1)
  )
)
```

```
ggdag(part_dag) +
  theme_dag() +
  annotate("text", x = 1, y = -1.5,
           label = "Solo observamos S = 1",
           size = 3, color = "red")
```



Solo observamos S = 1

Figura 4.2: Sesgo de participación

4.4.2 2. Sesgo de pérdida de seguimiento

En estudios longitudinales, la pérdida de seguimiento puede estar relacionada con exposición y resultado.

```
# Simulación de pérdida de seguimiento
set.seed(101)
n <- 2000

datos_fup <- data.frame(
  tratamiento = rbinom(n, 1, 0.5)
)
```

```

# El resultado verdadero
datos_fup$resultado_verdadero <- 50 - 10 * datos_fup$tratamiento +
  rnorm(n, 0, 15)

# Pérdida de seguimiento más probable en no tratados con mal resultado
prob_perdida <- plogis(-2 + 1.5 * (1 - datos_fup$tratamiento) -
  0.05 * datos_fup$resultado_verdadero)
datos_fup$perdido <- rbinom(n, 1, prob_perdida)

# Resultado observado
datos_fup$resultado_observado <- ifelse(datos_fup$perdido == 1,
  NA,
  datos_fup$resultado_verdadero)

# Efecto verdadero
efecto_verdadero <- mean(datos_fup$resultado_verdadero[datos_fup$tratamiento == 1]) -
  mean(datos_fup$resultado_verdadero[datos_fup$tratamiento == 0])

# Efecto observado (sesgado)
observados <- datos_fup[!is.na(datos_fup$resultado_observado), ]
efecto_observado <- mean(observados$resultado_observado[observados$tratamiento == 1]) -
  mean(observados$resultado_observado[observados$tratamiento == 0])

cat("Efecto verdadero:", round(efecto_verdadero, 2), "\n")

```

Efecto verdadero: -8.99

```
cat("Efecto observado:", round(efecto_observado, 2), "\n")
```

Efecto observado: -9.43

```
cat("Sesgo:", round(efecto_observado - efecto_verdadero, 2), "\n")
```

Sesgo: -0.45

4.4.3 3. Sesgo del sobreviviente

Estudiamos solo a quienes “sobrevivieron” a un proceso de selección.

```

surv_dag <- dagify(
  Muerte ~ Exposicion + Factor,
  Resultado ~ Exposicion,
  coords = list(
    x = c(Exposicion = 0, Muerte = 1, Resultado = 2, Factor = 1),
    y = c(Exposicion = 0, Muerte = -0.5, Resultado = 0, Factor = 0.5)
  )
)

ggdag(surv_dag) +
  theme_dag() +
  labs(title = "Sesgo del sobreviviente: solo observamos si Muerte = 0")

```

Sesgo del sobreviviente: solo observamos si Muerte = 0

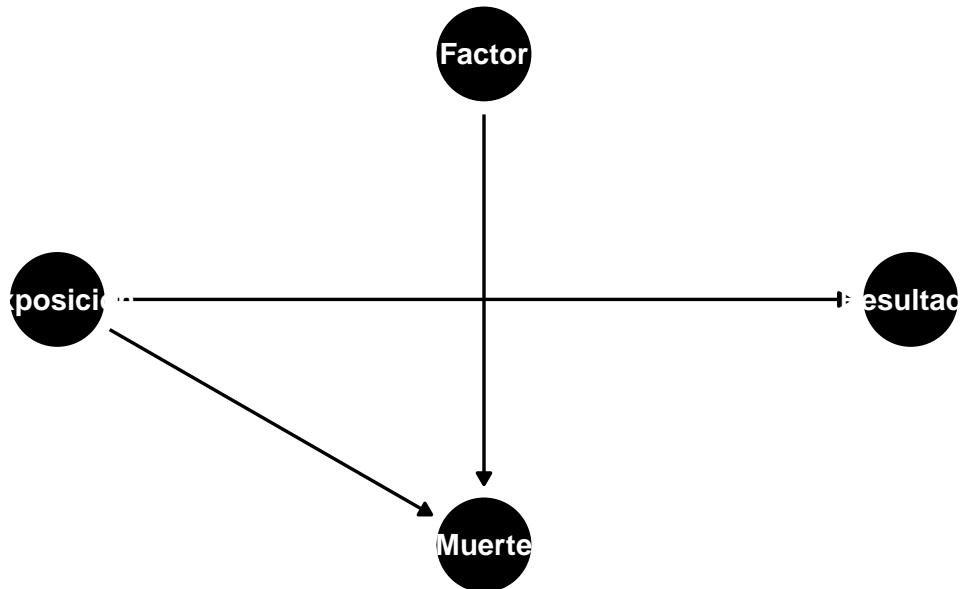


Figura 4.3: Sesgo del sobreviviente

4.4.4 4. Sesgo de Berkson

Selección basada en hospitalización u otra condición común.

```

# Ejemplo: Enfermedades A y B causan hospitalización
set.seed(202)

```

```

n_poblacion <- 50000

poblacion <- data.frame(
  enfermedad_A = rbinom(n_poblacion, 1, 0.1), # 10% prevalencia
  enfermedad_B = rbinom(n_poblacion, 1, 0.05) # 5% prevalencia
)

# Hospitalización si tiene alguna enfermedad
poblacion$hospitalizado <- as.numeric(
  poblacion$enfermedad_A == 1 | poblacion$enfermedad_B == 1
)

# En la población
cat("Correlación en población:",
  round(cor(poblacion$enfermedad_A, poblacion$enfermedad_B), 4), "\n")

```

Correlación en población: 0.0045

```

# Entre hospitalizados
hospitalizados <- poblacion[poblacion$hospitalizado == 1, ]
cat("Correlación entre hospitalizados:",
  round(cor(hospitalizados$enfermedad_A, hospitalizados$enfermedad_B), 4), "\n")

```

Correlación entre hospitalizados: -0.9204

4.5 Estrategias de mitigación

4.5.1 1. Ponderación por probabilidad de selección

Si conocemos los factores de selección, podemos usar **ponderación por inverso de la probabilidad de selección** (IPWS).

```

# Usando los datos de pérdida de seguimiento
datos_fup$prob_no_perdida <- 1 - prob_perdida
datos_fup$peso <- 1 / datos_fup$prob_no_perdida

# Análisis ponderado (solo en observados)
observados$peso <- datos_fup$peso[!is.na(datos_fup$resultado_observado)]

```

```

efecto_ponderado <- weighted.mean(
  observados$resultado_observado[observados$tratamiento == 1],
  observados$peso[observados$tratamiento == 1]) -
  weighted.mean(
  observados$resultado_observado[observados$tratamiento == 0],
  observados$peso[observados$tratamiento == 0])

cat("Efecto verdadero:", round(efecto_verdadero, 2), "\n")

```

Efecto verdadero: -8.99

```

cat("Efecto ponderado:", round(efecto_ponderado, 2), "\n")

```

Efecto ponderado: -9

4.5.2 2. Análisis de sensibilidad

Evaluar cómo diferentes asunciones sobre la selección afectan los resultados.

4.5.3 3. Diseño del estudio

- Minimizar pérdida de seguimiento
- Recopilar información sobre los que no participan
- Usar muestreo representativo

4.6 Identificación usando DAGs

```

library(dagitty)

full_dag <- dagitty("dag {
  X -> Y
  U -> X
  U -> Y
  X -> S
  Y -> S
}")

```

```
ggdag(full_dag) +
  theme_dag() +
  labs(title = "X: Exposición, Y: Resultado, S: Selección, U: Confusor")
```

X: Exposición, Y: Resultado, S: Selección, U: Confusor

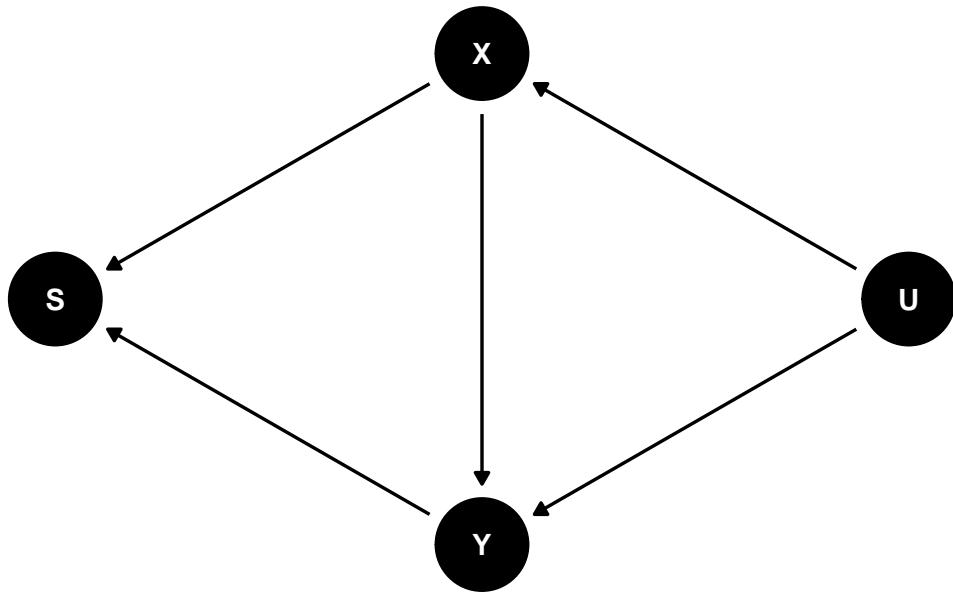


Figura 4.4: DAG completo con mecanismo de selección

```
# Verificar si el efecto es identificable
# dado que condicionamos en S
full_dag <- dagitty("dag {
  X -> Y
  U -> X
  U -> Y
  X -> S
  Y -> S
}")

# ¿Qué debemos ajustar si estamos condicionando en S?
cat("Conjuntos de ajuste cuando S está condicionado:\n")
```

Conjuntos de ajuste cuando S está condicionado:

```
print(adjustmentSets(full_dag, exposure = "X", outcome = "Y",
                      type = "all"))
```

{ U }

4.7 Ejercicios

💡 Ejercicio 1

Un estudio hospitalario encuentra que los pacientes con diabetes tienen menor riesgo de enfermedad pulmonar. Dibuja un DAG y explica por qué esto podría ser sesgo de Berkson.

💡 Ejercicio 2

En un ensayo clínico, 30% de los pacientes en el grupo placebo abandonan vs. 10% en el grupo tratamiento. Simula este escenario y cuantifica el sesgo resultante.

4.8 Resumen

- El sesgo de selección distorsiona la asociación entre exposición y resultado
- Desde la perspectiva de DAGs, surge al condicionar en un collider
- Incluye: sesgo de participación, pérdida de seguimiento, sobreviviente y Berkson
- La ponderación por probabilidad de selección puede corregir el sesgo
- El diseño cuidadoso del estudio es la mejor prevención

Referencias

Part II

Parte II: Métodos Avanzados

5 Variables Instrumentales

5.1 Objetivos de aprendizaje

Al finalizar este capítulo, serás capaz de:

- Definir una variable instrumental
- Verificar las condiciones de validez de un instrumento
- Aplicar estimación por mínimos cuadrados en dos etapas (2SLS)
- Interpretar efectos locales (LATE)

5.2 ¿Qué es una variable instrumental?

Una **variable instrumental** (IV) es una variable que afecta la exposición pero no tiene efecto directo sobre el resultado, excepto a través de la exposición.

```
library(ggdag)
library(ggplot2)

iv_dag <- dagify(
  Y ~ X + U,
  X ~ Z + U,
  coords = list(
    x = c(Z = 0, X = 1, Y = 2, U = 1.5),
    y = c(Z = 0, X = 0, Y = 0, U = 1)
  ),
  labels = c(
    Z = "Instrumento",
    X = "Exposición",
    Y = "Resultado",
    U = "Confusor\n(no observado)"
  )
)

ggdag(iv_dag, text = FALSE, use_labels = "label") +
```

```
theme_dag() +  
  labs(title = "Variable Instrumental")
```

Variable Instrumental

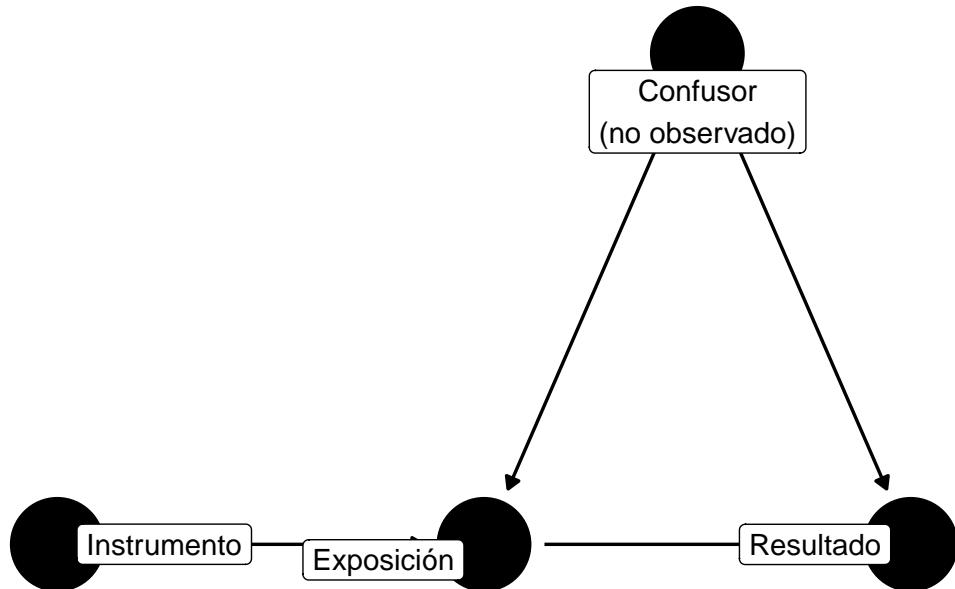


Figura 5.1: DAG de variable instrumental

5.3 Condiciones de validez

! Las tres condiciones

1. **Relevancia:** Z está asociada con X
2. **Independencia:** Z es independiente de confusores no observados (U)
3. **Exclusión:** Z no afecta Y excepto a través de X

5.3.1 Verificación de relevancia

La condición de relevancia es la única **verificable** empíricamente.

```
# Simulación  
set.seed(42)
```

```

n <- 5000

# Confusor no observado
U <- rnorm(n)

# Instrumento (ej: distancia a hospital)
Z <- rnorm(n)

# Exposición influenciada por Z y U
X <- 0.5 * Z + 0.8 * U + rnorm(n, 0, 0.5)

# Resultado influenciado por X y U
Y <- 2 * X + 1.5 * U + rnorm(n, 0, 1)

datos_iv <- data.frame(Z, X, Y, U)

# Primera etapa: Z -> X
primera_etapa <- lm(X ~ Z, data = datos_iv)
summary(primera_etapa)

```

Call:

```
lm(formula = X ~ Z, data = datos_iv)
```

Residuals:

Min	1Q	Median	3Q	Max
-3.1449	-0.6224	0.0190	0.6381	3.5124

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-0.01585	0.01331	-1.191	0.234
Z	0.50925	0.01321	38.549	<2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.941 on 4998 degrees of freedom
 Multiple R-squared: 0.2292, Adjusted R-squared: 0.229
 F-statistic: 1486 on 1 and 4998 DF, p-value: < 2.2e-16

```

# Regla de oro: F > 10
cat("\nF-statistic:",
  round(summary(primera_etapa)$fstatistic[1], 1), "\n")

```

F-statistic: 1486

5.4 Estimación por 2SLS

El método de **Mínimos Cuadrados en Dos Etapas** (2SLS):

1. **Primera etapa:** Predecir X usando Z
2. **Segunda etapa:** Usar X predicho para estimar el efecto sobre Y

```
# Método manual (para entender)

# Etapa 1: Predecir X
etapa1 <- lm(X ~ Z, data = datos_iv)
datos_iv$X_pred <- predict(etapa1)

# Etapa 2: Usar X predicho
etapa2 <- lm(Y ~ X_pred, data = datos_iv)

cat("Efecto estimado por 2SLS:",
    round(coef(etapa2)["X_pred"], 3), "\n")
```

Efecto estimado por 2SLS: 1.994

```
cat("Efecto verdadero: 2\n")
```

Efecto verdadero: 2

5.4.1 Usando el paquete ivreg

```
library(ivreg)

# Estimación correcta con errores estándar apropiados
modelo_iv <- ivreg(Y ~ X | Z, data = datos_iv)
summary(modelo_iv)
```

```

Call:
ivreg(formula = Y ~ X | Z, data = datos_iv)

Residuals:
    Min         1Q     Median         3Q        Max
-5.885405 -1.216900 -0.002617  1.223677  8.117727

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.01126    0.02559   -0.44    0.66
X            1.99359    0.04985   39.99  <2e-16 ***
                                         ***

Diagnostic tests:
            df1  df2 statistic p-value
Weak instruments     1 4998      1486 <2e-16 ***
Wu-Hausman          1 4997      1148 <2e-16 ***
Sargan              0    NA       NA     NA
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.808 on 4998 degrees of freedom
Multiple R-Squared: 0.7412, Adjusted R-squared: 0.7411
Wald test: 1599 on 1 and 4998 DF, p-value: < 2.2e-16

```

5.5 Comparación con OLS

```

# OLS (sesgado por confusión)
modelo_ols <- lm(Y ~ X, data = datos_iv)

cat("Efecto OLS (sesgado):", round(coef(modelo_ols)["X"], 3), "\n")

```

Efecto OLS (sesgado): 3.042

```

cat("Efecto IV:", round(coef(modelo_iv)["X"], 3), "\n")

```

Efecto IV: 1.994

```
cat("Efecto verdadero: 2\n")
```

Efecto verdadero: 2

5.6 Efecto Local (LATE)

El estimador IV identifica el **Efecto Promedio del Tratamiento Local** (LATE): el efecto entre los “compliers” (quienes cumplen con la asignación del instrumento).

```
tipos <- data.frame(
  Tipo = c("Compliers", "Always-takers", "Never-takers", "Defiers"),
  Z0_X = c("X=0", "X=1", "X=0", "X=1"),
  Z1_X = c("X=1", "X=1", "X=0", "X=0"),
  Descripcion = c(
    "Tratados solo si Z=1",
    "Siempre tratados",
    "Nunca tratados",
    "Tratados solo si Z=0"
  )
)

knitr::kable(tipos,
  col.names = c("Tipo", "Si Z=0", "Si Z=1", "Descripción"),
  caption = "Tipos de individuos según respuesta al instrumento")
```

Tabla 5.1: Tipos de individuos según respuesta al instrumento

Tipo	Si Z=0	Si Z=1	Descripción
Compliers	X=0	X=1	Tratados solo si Z=1
Always-takers	X=1	X=1	Siempre tratados
Never-takers	X=0	X=0	Nunca tratados
Defiers	X=1	X=0	Tratados solo si Z=0

Figura 5.2: Tipos de individuos según respuesta al instrumento

Supuesto de monotonía

Generalmente asumimos que no hay “defiers” (monotonía). Esto significa que el instrumento solo puede aumentar (o no cambiar) la probabilidad de tratamiento, nunca disminuirla.

5.7 Ejemplos clásicos de instrumentos

Contexto	Exposición	Instrumento
Retornos a educación	Años de educación	Trimestre de nacimiento
Efecto de cesárea	Tipo de parto	Preferencia del médico
Efecto de transfusión	Volumen transfundido	Distancia al banco de sangre
Randomización con incumplimiento	Tratamiento recibido	Asignación aleatoria

5.8 Instrumentos débiles

Un instrumento **débil** tiene baja correlación con la exposición.

```
# Instrumento débil
set.seed(123)
Z_debil <- rnorm(n)
X_debil <- 0.05 * Z_debil + 0.8 * U + rnorm(n, 0, 0.5) # Coeficiente pequeño
Y_debil <- 2 * X_debil + 1.5 * U + rnorm(n, 0, 1)

# Primera etapa
etapa1_debil <- lm(X_debil ~ Z_debil)
cat("F-statistic (instrumento débil):",
    round(summary(etapa1_debil)$fstatistic[1], 1), "\n")
```

F-statistic (instrumento débil): 6.7

```
# IV con instrumento débil
datos_debil <- data.frame(Z = Z_debil, X = X_debil, Y = Y_debil)
modelo_iv_debil <- ivreg(Y ~ X | Z, data = datos_debil)

cat("Efecto IV (instrumento débil):",
    round(coef(modelo_iv_debil)["X"], 3), "\n")
```

Efecto IV (instrumento débil): 1.459

```
cat("Error estándar:",  
    round(summary(modelo_iv_debil)$coefficients["X", "Std. Error"], 3), "\n")
```

Error estándar: 0.906

5.9 Diagnósticos para IV

5.9.1 Test de instrumentos débiles

```
# Test de Cragg-Donald / Kleibergen-Paap  
summary(modelo_iv, diagnostics = TRUE)
```

Call:

```
ivreg(formula = Y ~ X | Z, data = datos_iv)
```

Residuals:

Min	1Q	Median	3Q	Max
-5.885405	-1.216900	-0.002617	1.223677	8.117727

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-0.01126	0.02559	-0.44	0.66
X	1.99359	0.04985	39.99	<2e-16 ***

Diagnostic tests:

	df1	df2	statistic	p-value							
Weak instruments	1	4998	1486	<2e-16 ***							
Wu-Hausman	1	4997	1148	<2e-16 ***							
Sargan	0	NA	NA	NA							

Signif. codes:	0	'***'	0.001	'**'	0.01	'*'	0.05	'..'	0.1	' '	1

Residual standard error: 1.808 on 4998 degrees of freedom

Multiple R-Squared: 0.7412, Adjusted R-squared: 0.7411

Wald test: 1599 on 1 and 4998 DF, p-value: < 2.2e-16

5.9.2 Test de sobreidentificación

Cuando tienes más de un instrumento, puedes probar la validez del conjunto.

```
# Múltiples instrumentos
Z2 <- Z + rnorm(n, 0, 0.5) # Segundo instrumento
datos_iv$Z2 <- Z2

modelo_iv_multi <- ivreg(Y ~ X | Z + Z2, data = datos_iv)
summary(modelo_iv_multi, diagnostics = TRUE)
```

Call:

```
ivreg(formula = Y ~ X | Z + Z2, data = datos_iv)
```

Residuals:

Min	1Q	Median	3Q	Max
-5.885158	-1.216807	-0.002639	1.223392	8.117374

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-0.01126	0.02559	-0.44	0.66
X	1.99370	0.04984	40.00	<2e-16 ***

Diagnostic tests:

	df1	df2	statistic	p-value
Weak instruments	2	4997	743.291	<2e-16 ***
Wu-Hausman	1	4997	1148.871	<2e-16 ***
Sargan	1	NA	0.012	0.913

Signif. codes:	0	***	0.001	**
			0.01	*
			0.05	.
			0.1	'
			'	1

Residual standard error: 1.808 on 4998 degrees of freedom

Multiple R-Squared: 0.7412, Adjusted R-squared: 0.7412

Wald test: 1600 on 1 and 4998 DF, p-value: < 2.2e-16

5.10 Ejercicios

💡 Ejercicio 1

Un economista quiere estimar el efecto de los años de educación sobre el salario. Propone usar la distancia al colegio más cercano como instrumento.

1. Dibuja el DAG
2. Discute la plausibilidad de las tres condiciones
3. ¿Qué población representaría el LATE?

💡 Ejercicio 2

Simula un escenario donde el instrumento viola la restricción de exclusión (Z afecta Y directamente). ¿Qué sucede con la estimación IV?

5.11 Resumen

- Las variables instrumentales permiten identificar efectos causales con confusores no observados
- Requieren tres condiciones: relevancia, independencia y exclusión
- Solo la relevancia es verificable empíricamente
- El método 2SLS es el más común para estimación
- El efecto identificado es el LATE (efecto entre compliers)
- Los instrumentos débiles producen estimaciones sesgadas e imprecisas

Referencias

6 Causalidad y Supervivencia

6.1 Objetivos de aprendizaje

Al finalizar este capítulo, serás capaz de:

- Integrar conceptos causales con análisis de supervivencia
- Reconocer los sesgos específicos de datos de tiempo al evento
- Aplicar métodos de ponderación para estimación causal
- Manejar riesgos competitivos desde una perspectiva causal

6.2 Análisis de supervivencia: repaso

El **análisis de supervivencia** estudia el tiempo hasta que ocurre un evento de interés.

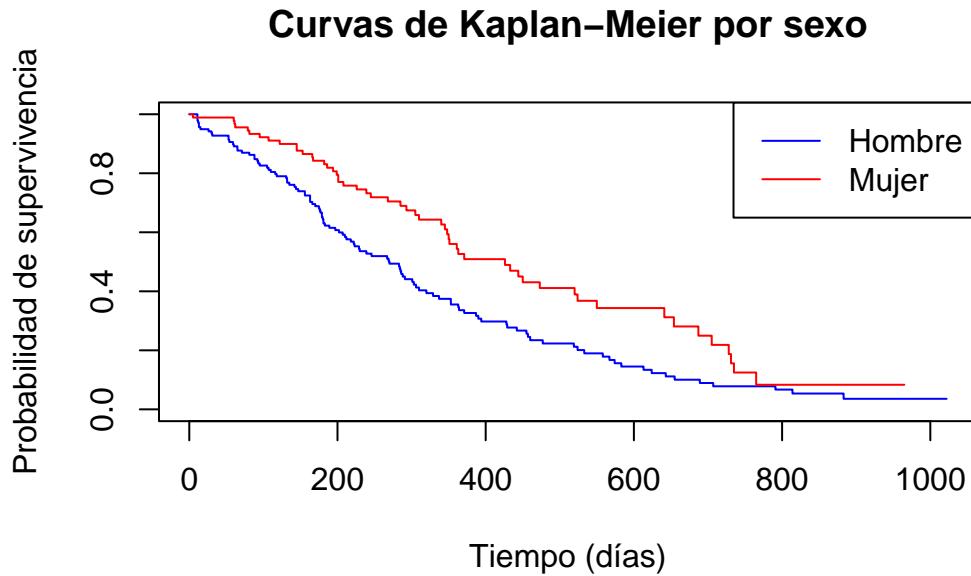
6.2.1 Conceptos fundamentales

```
library(survival)
library(ggplot2)

# Datos de ejemplo
data(lung)

# Modelo de Kaplan-Meier
km_fit <- survfit(Surv(time, status) ~ sex, data = lung)

# Visualización
plot(km_fit, col = c("blue", "red"),
      xlab = "Tiempo (días)",
      ylab = "Probabilidad de supervivencia",
      main = "Curvas de Kaplan-Meier por sexo")
legend("topright", c("Hombre", "Mujer"), col = c("blue", "red"), lty = 1)
```



6.3 Sesgos en supervivencia

6.3.1 Sesgo de tiempo inmortal

El **sesgo de tiempo inmortal** ocurre cuando hay un período durante el cual los sujetos expuestos no pueden experimentar el evento.

```
library(ggdag)

it_dag <- dagify(
  Y ~ X + T,
  X ~ T,
  coords = list(
    x = c(T = 0, X = 1, Y = 2),
    y = c(T = 0, X = 0.3, Y = 0)
  ),
  labels = c(
    T = "Tiempo\n(sobrevivir\nhasta exposición)",
    X = "Exposición",
    Y = "Muerte"
  )
)
```

```
)
ggdag(it_dag, text = FALSE, use_labels = "label") +
  theme_dag() +
  labs(title = "El tiempo de supervivencia necesario para exponerse crea sesgo")
```

El tiempo de supervivencia necesario para exponerse crea sesgo

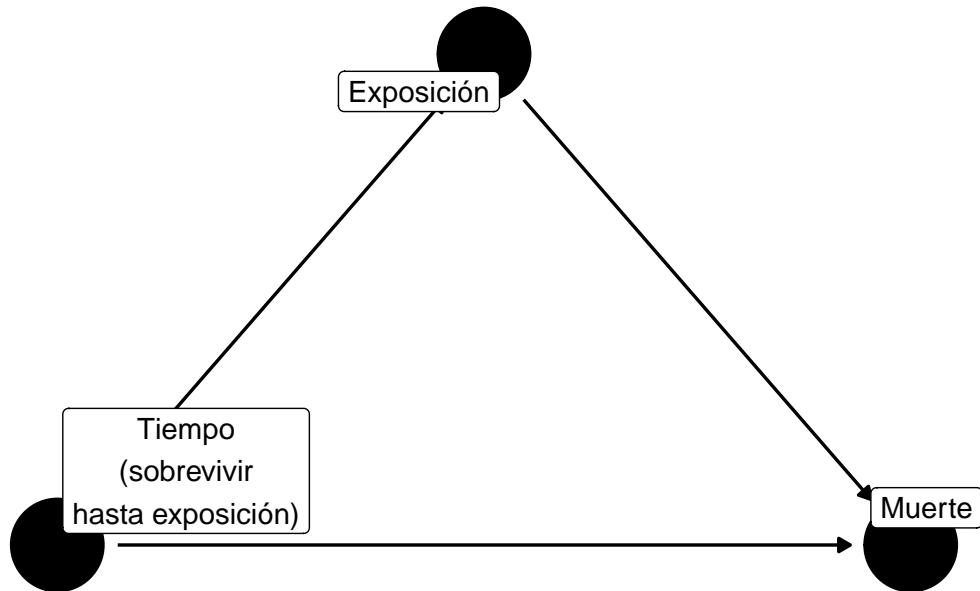


Figura 6.1: Sesgo de tiempo inmortal

6.3.2 Ejemplo de sesgo de tiempo inmortal

```

# Simulación
set.seed(303)
n <- 1000

# Tiempo hasta exposición (si ocurre)
tiempo_exposición <- rexp(n, 0.1)

# Tiempo de supervivencia verdadero (independiente de exposición)
tiempo_muerte_base <- rexp(n, 0.05)
```

```

# Asignación de exposición (solo si sobrevive hasta entonces)
datos_it <- data.frame(
  id = 1:n,
  tiempo_exposicion = tiempo_exposicion,
  tiempo_muerte_base = tiempo_muerte_base
)

# ¿Se expuso? (solo si sobrevivió hasta el momento de exposición)
datos_it$expuesto <- datos_it$tiempo_muerte_base > datos_it$tiempo_exposicion

# Tiempo observado
datos_it$tiempo_observado <- pmin(datos_it$tiempo_muerte_base, 100)
datos_it$evento <- datos_it$tiempo_muerte_base <= 100

# Análisis INCORRECTO (ignora tiempo inmortal)
modelo_incorrecto <- coxph(Surv(tiempo_observado, evento) ~ expuesto,
                             data = datos_it)

cat("HR (análisis incorrecto, sesgo de tiempo inmortal):",
    round(exp(coef(modelo_incorrecto)), 3), "\n")

```

HR (análisis incorrecto, sesgo de tiempo inmortal): 0.18

```
cat("HR verdadero: 1.0 (la exposición no tiene efecto)\n")
```

HR verdadero: 1.0 (la exposición no tiene efecto)

6.3.3 Corrección con tiempo dependiente

```

library(survival)

# Crear datos en formato largo (tiempo-dependiente)
datos_tv <- survSplit(Surv(tiempo_observado, evento) ~ .,
                       data = datos_it,
                       cut = datos_it$tiempo_exposicion[datos_it$expuesto],
                       episode = "periodo")

# Exposición como variable tiempo-dependiente
datos_tv$expuesto_tv <- with(datos_tv,

```

```

            expuesto & tstart >= tiempo_exposicion)

# Análisis CORRECTO
modelo_correcto <- coxph(Surv(tstart, tiempo_observado, evento) ~ expuesto_tv,
                           data = datos_tv)

cat("HR (análisis correcto):",
    round(exp(coef(modelo_correcto)), 3), "\n")

```

HR (análisis correcto): 0.889

6.4 Ponderación en supervivencia (IPTW)

Podemos combinar propensity scores con análisis de supervivencia.

```

library(WeightIt)

# Datos con confusión
set.seed(404)
n <- 1500

datos_surv <- data.frame(
  edad = rnorm(n, 60, 10),
  comorbilidad = rbinom(n, 1, 0.3)
)

# Tratamiento influenciado por confusores
prob_trat <- plogis(-3 + 0.05 * datos_surv$edad + 1 * datos_surv$comorbilidad)
datos_surv$tratamiento <- rbinom(n, 1, prob_trat)

# Tiempo de supervivencia
hazard <- 0.01 * exp(0.03 * datos_surv$edad +
                      0.5 * datos_surv$comorbilidad -
                      0.4 * datos_surv$tratamiento) # Efecto protector
datos_surv$tiempo <- rexp(n, hazard)
datos_surv$tiempo <- pmin(datos_surv$tiempo, 100)
datos_surv$evento <- datos_surv$tiempo < 100

# Calcular pesos
pesos_surv <- weightit(tratamiento ~ edad + comorbilidad,

```

```

        data = datos_surv,
        method = "ps",
        estimand = "ATE")

# Modelo sin ponderación (sesgado)
modelo_crudo <- coxph(Surv(tiempo, evento) ~ tratamiento,
                        data = datos_surv)

# Modelo ponderado
modelo_ponderado <- coxph(Surv(tiempo, evento) ~ tratamiento,
                            data = datos_surv,
                            weights = pesos_surv$weights,
                            robust = TRUE)

cat("HR crudo:", round(exp(coef(modelo_crudo)), 3), "\n")

```

HR crudo: 0.863

```
cat("HR ponderado:", round(exp(coef(modelo_ponderado)), 3), "\n")
```

HR ponderado: 0.701

```
cat("HR verdadero: exp(-0.4) =", round(exp(-0.4), 3), "\n")
```

HR verdadero: exp(-0.4) = 0.67

6.5 Riesgos competitivos

Los **riesgos competitivos** ocurren cuando múltiples tipos de eventos pueden terminar el seguimiento.

```

cr_dag <- dagify(
  D1 ~ X + C,
  D2 ~ X + C,
  coords = list(
    x = c(X = 0, C = 1, D1 = 2, D2 = 2),
    y = c(X = 0, C = 1, D1 = 0.5, D2 = -0.5)
  ),

```

```

labels = c(
  X = "Exposición",
  C = "Confusor",
  D1 = "Muerte por\ncáncer",
  D2 = "Muerte por\notra causa"
)
)

ggdag(cr_dag, text = FALSE, use_labels = "label") +
  theme_dag() +
  labs(title = "Riesgos competitivos")

```

Riesgos competitivos

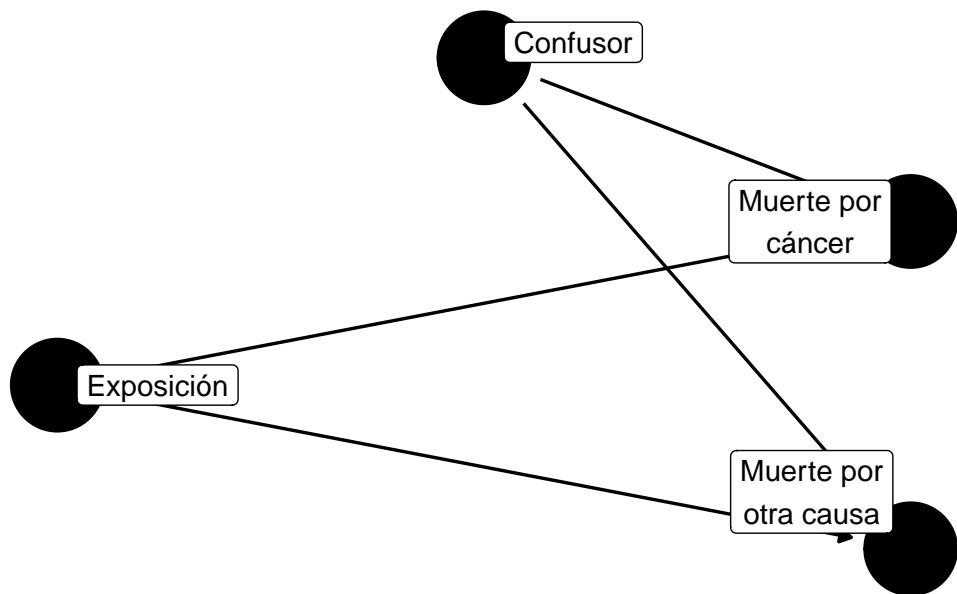


Figura 6.2: Estructura de riesgos competitivos

6.5.1 Enfoque de subdistribución (Fine-Gray)

```

library(cmprsk)

# Simular riesgos competitivos
set.seed(505)

```

```

n <- 800

datos_cr <- data.frame(
  tratamiento = rbinom(n, 1, 0.5)
)

# Dos tipos de eventos
tiempo_cancer <- rexp(n, 0.03 - 0.01 * datos_cr$tratamiento)
tiempo_otro <- rexp(n, 0.02)

# Evento observado
datos_cr$tiempo <- pmin(tiempo_cancer, tiempo_otro, 50)
datos_cr$tipo_evento <- ifelse(datos_cr$tiempo >= 50, 0,
                                 ifelse(tiempo_cancer < tiempo_otro, 1, 2))

# Modelo Fine-Gray para muerte por cáncer
fg_fit <- crr(datos_cr$tiempo,
                datos_cr$tipo_evento,
                datos_cr[, "tratamiento", drop = FALSE],
                failcode = 1)

summary(fg_fit)

```

Competing Risks Regression

Call:

```
crr(ftime = datos_cr$tiempo, fstatus = datos_cr$tipo_evento,
  cov1 = datos_cr[, "tratamiento", drop = FALSE], failcode = 1)
```

coef	exp(coef)	se(coef)	z	p-value
tratamiento	-0.381	0.683	0.105	-3.63 0.00028

	exp(coef)	exp(-coef)	2.5%	97.5%
tratamiento	0.683	1.46	0.556	0.839

```

Num. cases = 800
Pseudo Log-likelihood = -2340
Pseudo likelihood ratio test = 13.2 on 1 df,

```

6.5.2 Interpretación causal

⚠️ Cuidado con la interpretación

El modelo de Fine-Gray estima el efecto sobre la **incidencia acumulada**, no sobre el riesgo causa-específico. Esto tiene implicaciones para la interpretación causal porque incluye implícitamente los efectos sobre el riesgo competitivo.

6.6 Análisis causa-específico

```
# Análisis causa-específico (censurar el otro evento)
datos_cs <- datos_cr
datos_cs$evento_cancer <- as.numeric(datos_cs$tipo_evento == 1)

modelo_cs <- coxph(Surv(tiempo, evento_cancer) ~ tratamiento,
                     data = datos_cs)

cat("HR causa-específico (muerte por cáncer):",
    round(exp(coef(modelo_cs)), 3), "\n")
```

HR causa-específico (muerte por cáncer): 0.731

6.7 Ejercicios

💡 Ejercicio 1

Un estudio encuentra que los pacientes que reciben un trasplante de riñón tienen mejor supervivencia que los que permanecen en diálisis. Sin embargo, los pacientes deben sobrevivir en lista de espera para recibir el trasplante.

1. Identifica el sesgo de tiempo inmortal
2. Propón un diseño analítico correcto

💡 Ejercicio 2

En un estudio de cáncer, algunos pacientes mueren por causas cardiovasculares antes de morir por cáncer.

1. ¿Cómo afecta esto la estimación del efecto del tratamiento?
2. Compara las estimaciones Fine-Gray vs causa-específica

6.8 Resumen

- El análisis de supervivencia requiere consideraciones causales especiales
- El sesgo de tiempo inmortal surge cuando la exposición requiere sobrevivir
- La solución es usar exposición como variable tiempo-dependiente
- IPTW se puede aplicar a modelos de supervivencia
- Los riesgos competitivos requieren elegir entre estimandos alternativos
- Fine-Gray vs causa-específico tienen diferentes interpretaciones causales

Referencias

7 Análisis de Sensibilidad

7.1 Objetivos de aprendizaje

Al finalizar este capítulo, serás capaz de:

- Entender la importancia del análisis de sensibilidad
- Calcular e interpretar el E-value
- Aplicar métodos de sensibilidad cuantitativos
- Comunicar la robustez de hallazgos causales

7.2 ¿Por qué análisis de sensibilidad?

En estudios observacionales, **nunca podemos estar seguros** de haber controlado todos los confusores. El análisis de sensibilidad evalúa qué tan fuertes tendrían que ser los confusores no medidos para explicar nuestros resultados.

```
library(ggdag)
library(ggplot2)

sens_dag <- dagify(
  Y ~ X + C + U,
  X ~ C + U,
  coords = list(
    x = c(X = 0, Y = 2, C = 1, U = 1),
    y = c(X = 0, Y = 0, C = 0.7, U = -0.7)
  ),
  labels = c(
    X = "Exposición",
    Y = "Resultado",
    C = "Confusor\n(meido)",
    U = "Confusor\n(no medido)"
  )
)
```

```
ggdag(sens_dag, text = FALSE, use_labels = "label") +
  theme_dag() +
  labs(title = "U podría explicar la asociación observada")
```

U podría explicar la asociación observada

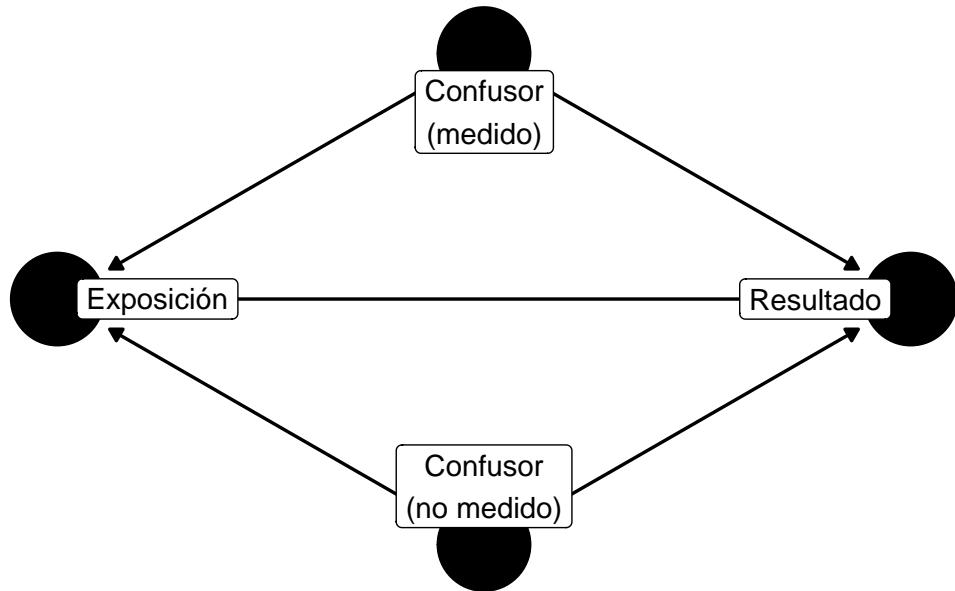


Figura 7.1: El confusor no medido (U) amenaza la validez

7.3 El E-value

El **E-value** (VanderWeele y Ding 2017) es la asociación mínima que un confusor no medido tendría que tener con *tanto* la exposición *como* el resultado para explicar completamente la asociación observada.

$$E\text{-value} = RR + \sqrt{RR \times (RR - 1)}$$

```
library(EValue)

# Ejemplo: RR observado = 2.5
rr_observado <- 2.5
ic_inferior <- 1.8 # Límite inferior del IC 95%
```

```
# Calcular E-value
evals <- evals.RR(est = rr_observado, lo = ic_inferior)
evals
```

	point	lower	upper
RR	2.500000	1.8	NA
E-values	4.436492	3.0	NA

7.3.1 Interpretación del E-value

```
# Crear datos para el gráfico
rr_seq <- seq(1.1, 5, 0.1)
evalue_seq <- rr_seq + sqrt(rr_seq * (rr_seq - 1))

plot_data <- data.frame(
  RR = rr_seq,
  EValue = evalue_seq
)

# Valores del ejemplo anterior
rr_ejemplo <- 2.5
evalue_ejemplo <- 2.5 + sqrt(2.5 * (2.5 - 1))

ggplot(plot_data, aes(x = RR, y = EValue)) +
  geom_line(size = 1.2, color = "steelblue") +
  geom_vline(xintercept = rr_ejemplo, linetype = "dashed", color = "red") +
  geom_hline(yintercept = evalue_ejemplo, linetype = "dashed", color = "red") +
  annotate("point", x = rr_ejemplo, y = evalue_ejemplo, size = 4, color = "red") +
  labs(x = "Riesgo Relativo observado",
       y = "E-value",
       title = "E-value en función del RR observado") +
  theme_minimal()
```

E-value en función del RR observado

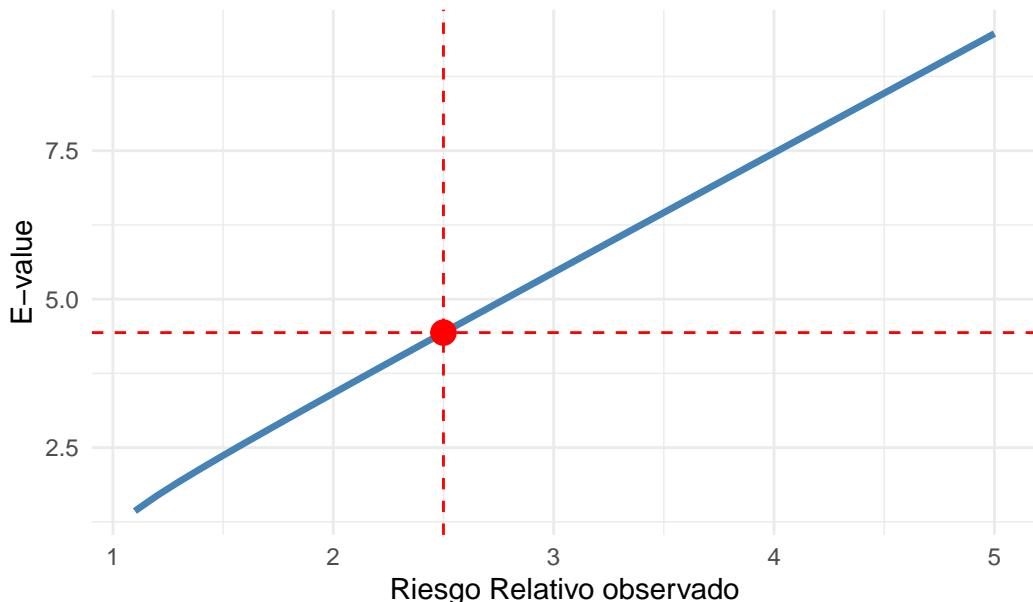


Figura 7.2: Interpretación gráfica del E-value

Interpretación

Para un RR de 2.5, el E-value es aproximadamente 4.4. Esto significa que un confusor no medido tendría que tener un RR de al menos 4.4 con tanto la exposición como el resultado para explicar completamente la asociación observada.

7.4 Análisis de sensibilidad con `sensemakr`

El paquete `sensemakr` (Cinelli y Hazlett 2020) proporciona herramientas más sofisticadas.

```
library(sensemakr)

# Ejemplo con datos de discriminación laboral
data("darfur")

# Modelo ajustado
modelo <- lm(peacefactor ~ directlyharmed + age + farmer_dar +
               herder_dar + pastvoted + female, data = darfur)
```

```

# Análisis de sensibilidad
sens <- sensemakr(model = modelo,
                     treatment = "directlyharmed",
                     benchmark_covariates = "female",
                     kd = 1:3,
                     q = 1)

summary(sens)

```

Sensitivity Analysis to Unobserved Confounding

Model Formula: peacefactor ~ directlyharmed + age + farmer_dar + herder_dar + pastvoted + female

Null hypothesis: q = 1 and reduce = TRUE

-- This means we are considering biases that reduce the absolute value of the current estimate

-- The null hypothesis deemed problematic is $H_0: \tau = 0$

Unadjusted Estimates of 'directlyharmed':

Coef. estimate: 0.0489
 Standard Error: 0.0184
 t-value ($H_0: \tau = 0$): 2.6648

Sensitivity Statistics:

Partial R2 of treatment with outcome: 0.0056
 Robustness Value, q = 1: 0.0721
 Robustness Value, q = 1, alpha = 0.05: 0.0195

Verbal interpretation of sensitivity statistics:

-- Partial R2 of the treatment with the outcome: an extreme confounder (orthogonal to the covariates)

-- Robustness Value, q = 1: unobserved confounders (orthogonal to the covariates) that explain the outcome

-- Robustness Value, q = 1, alpha = 0.05: unobserved confounders (orthogonal to the covariates) that explain the outcome with 95% confidence

Bounds on omitted variable bias:

--The table below shows the maximum strength of unobserved confounders with association with the outcome

Bound	Label	R2dz.x	R2yz.dx	Treatment	Adjusted Estimate	Adjusted Se
1x	female	0.0027	0.1314	directlyharmed	0.0366	0.0171

2x female	0.0054	0.2627	directlyharmed	0.0243	0.0158
3x female	0.0081	0.3941	directlyharmed	0.0119	0.0144
Adjusted T	Adjusted	Lower CI	Adjusted	Upper CI	
2.1357		0.0030		0.0703	
1.5341		-0.0068		0.0553	
0.8273		-0.0163		0.0401	

7.4.1 Visualización de contornos

```
plot(sens)
```

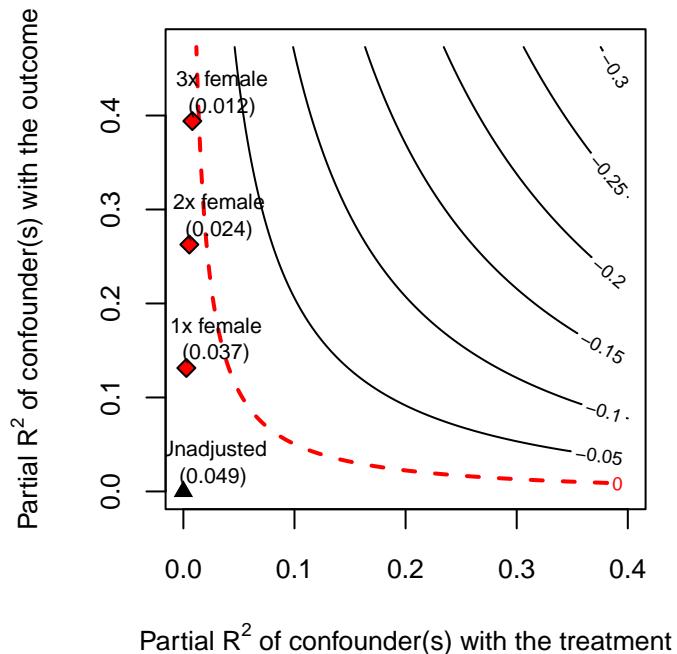


Figura 7.3: Contornos de sensibilidad

7.4.2 Robustness Value (RV)

```
# El Robustness Value indica qué tan fuerte tendría que ser
# un confusor para reducir el efecto a cero
```

```
cat("Robustness Value (q=1):",
  round(sens$bounds$r2yz.dx[1], 3), "\n")
```

```
Robustness Value (q=1): 0.131
```

```
cat("\nInterpretación: Un confusor tendría que explicar al menos",
  round(sens$bounds$r2yz.dx[1] * 100, 1), "% de la varianza\n",
  "residual tanto de X como de Y para eliminar el efecto.\n")
```

Interpretación: Un confusor tendría que explicar al menos 13.1 % de la varianza residual tanto de X como de Y para eliminar el efecto.

7.5 Fórmula de sesgo de confusión

La fórmula de sesgo permite calcular cuánto cambiaría la estimación dado un confusor específico:

$$\text{Sesgo} = \frac{(RR_{UY} - 1) \times (RR_{UX} - 1)}{RR_{UY} + (RR_{UX} - 1) \times P(U)}$$

```
# Función para calcular sesgo
calcular_sesgo <- function(rr_uy, rr_ux, p_u = 0.5) {
  # Fórmula simplificada
  bias <- (rr_uy - 1) * (rr_ux - 1) /
    (rr_uy + (rr_ux - 1) * p_u)
  return(bias)
}

# Ejemplo: ¿Cuánto sesgo introduciría un confusor con RR=1.5
# con exposición y resultado?
sesgo <- calcular_sesgo(rr_uy = 1.5, rr_ux = 1.5)
cat("Sesgo aproximado:", round(sesgo, 3), "\n")
```

```
Sesgo aproximado: 0.143
```

7.6 Análisis de sensibilidad para IPTW

```

# Simulación con confusor no medido
set.seed(606)
n <- 2000

# Confusor medido
C_medido <- rnorm(n)

# Confusor NO medido
U <- rnorm(n)

# Exposición
prob_X <- plogis(-1 + 0.5 * C_medido + 0.8 * U)
X <- rbinom(n, 1, prob_X)

# Resultado
Y <- 2 + 3 * X + 1.5 * C_medido + 2 * U + rnorm(n)

datos_sens <- data.frame(C_medido, U, X, Y)

# Efecto verdadero (controlando U)
efecto_verdadero <- coef(lm(Y ~ X + C_medido + U, data = datos_sens))["X"]

# Efecto sin U (sesgado)
efecto_sesgado <- coef(lm(Y ~ X + C_medido, data = datos_sens))["X"]

# IPTW sin U
library(WeightIt)
pesos <- weightit(X ~ C_medido, data = datos_sens, method = "ps")
efecto_iptw <- coef(lm(Y ~ X, data = datos_sens, weights = pesos$weights))["X"]

cat("Efecto verdadero (con U):", round(efecto_verdadero, 3), "\n")

```

Efecto verdadero (con U): 2.958

```
cat("Efecto sin U (OLS):", round(efecto_sesgado, 3), "\n")
```

Efecto sin U (OLS): 4.468

```
cat("Efecto IPTW (sin U):", round(efecto_iptw, 3), "\n")
```

Efecto IPTW (sin U): 4.432

```
cat("Sesgo por U:", round(efecto_iptw - efecto_verdadero, 3), "\n")
```

Sesgo por U: 1.473

7.7 Comunicando resultados de sensibilidad

7.7.1 Buenas prácticas

1. **Reportar E-values** para el efecto y su intervalo de confianza
2. **Comparar con confusores conocidos:** ¿Es plausible un confusor tan fuerte?
3. **Usar benchmarks:** Comparar con la fuerza de confusores medidos
4. **Ser transparente** sobre las limitaciones

7.7.2 Ejemplo de reporte

```
# Supongamos un RR observado
rr_obs <- 1.8
rr_lo <- 1.3
rr_hi <- 2.4

# E-values
ev <- evals.RR(est = rr_obs, lo = rr_lo, hi = rr_hi)

cat("REPORTE DE SENSIBILIDAD\n")
```

REPORTE DE SENSIBILIDAD

```
cat("=====\\n\\n")
```

=====

```
cat("RR observado: ", rr_obs, " (IC 95%: ", rr_lo, "-", rr_hi, " )\\n\\n", sep = "")
```

RR observado: 1.8 (IC 95%: 1.3-2.4)

```
cat("E-value para el estimado puntual:", round(ev["E-values", "point"], 2), "\n")  
  
E-value para el estimado puntual: 3  
  
cat("E-value para el límite inferior del IC:", round(ev["E-values", "lower"], 2), "\n\n")  
  
E-value para el límite inferior del IC: 1.92  
  
cat("Interpretación: Para explicar completamente la asociación observada,\n")  
  
Interpretación: Para explicar completamente la asociación observada,  
  
cat("un confusor no medido tendría que estar asociado con tanto la\n")  
  
un confusor no medido tendría que estar asociado con tanto la  
  
cat("exposición como el resultado con un RR de al menos",  
    round(ev["E-values", "point"], 2), ".\n\n")  
  
exposición como el resultado con un RR de al menos 3 .  
  
cat("Para mover el IC inferior a 1.0, la asociación del confusor\n")  
  
Para mover el IC inferior a 1.0, la asociación del confusor  
  
cat("tendría que ser de al menos RR =", round(ev["E-values", "lower"], 2), ".\n")  
  
tendría que ser de al menos RR = 1.92 .
```

7.8 Ejercicios

💡 Ejercicio 1

Un estudio observacional encuentra que el consumo de vegetales está asociado con menor mortalidad (HR = 0.75, IC 95%: 0.65-0.85).

1. Calcula el E-value
2. ¿Qué tan fuerte tendría que ser un confusor para explicar esto?
3. Considera confusores plausibles (nivel socioeconómico, acceso a salud)

💡 Ejercicio 2

Usando **sensemakr**, realiza un análisis de sensibilidad completo para un modelo de tu elección. Interpreta los resultados y determina si el efecto es robusto.

7.9 Resumen

- El análisis de sensibilidad evalúa la robustez de hallazgos causales
- El E-value cuantifica la fuerza mínima de un confusor para explicar el efecto
- **sensemakr** proporciona visualizaciones y métricas avanzadas
- Siempre comparar con la fuerza de confusores conocidos
- Reportar transparentemente las limitaciones del estudio

Referencias

8 Análisis de Mediación

8.1 Objetivos de aprendizaje

Al finalizar este capítulo, serás capaz de:

- Distinguir entre efectos directos e indirectos
- Definir efectos de mediación causal (NDE, NIE)
- Identificar las asunciones para mediación
- Aplicar el paquete `mediation` en R

8.2 ¿Qué es la mediación?

La **mediación** examina *cómo* o *por qué* una exposición afecta un resultado. El efecto puede ser:

- **Directo:** $X \rightarrow Y$
- **Indirecto:** $X \rightarrow M \rightarrow Y$ (a través de un mediador)

```
library(ggdag)
library(ggplot2)

med_dag <- dagify(
  M ~ X,
  Y ~ X + M,
  coords = list(
    x = c(X = 0, M = 1, Y = 2),
    y = c(X = 0, M = 0.5, Y = 0)
  ),
  labels = c(
    X = "Exposición",
    M = "Mediador",
    Y = "Resultado"
  )
)
```

```
ggdag(med_dag, text = FALSE, use_labels = "label") +  
  theme_dag() +  
  labs(title = "X afecta Y directamente y a través de M")
```

X afecta Y directamente y a través de M

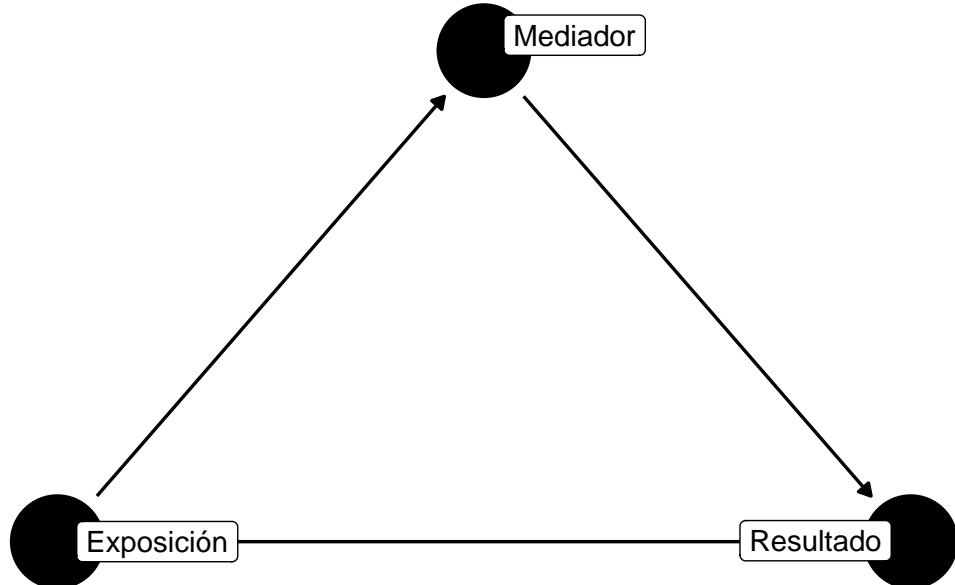


Figura 8.1: Diagrama de mediación simple

8.3 Definiciones causales

8.3.1 Efectos naturales (NDE y NIE)

- **Efecto Directo Natural (NDE):** Efecto de X sobre Y manteniendo M en el nivel que habría tenido sin la exposición
- **Efecto Indirecto Natural (NIE):** Efecto de cambiar M desde el nivel sin exposición al nivel con exposición, manteniendo X fijo

$$\text{Efecto Total} = \text{NDE} + \text{NIE}$$

```

library(ggplot2)

# Datos para visualización
efectos <- data.frame(
  Tipo = c("Directo (NDE)", "Indirecto (NIE") ,
  Valor = c(0.6, 0.4),
  ymin = 0,
  ymax = c(0.6, 1.0)
)

ggplot(efectos, aes(x = 1, y = Valor, fill = Tipo)) +
  geom_col(position = "stack", width = 0.5) +
  coord_flip() +
  labs(y = "Proporción del efecto total",
       x = "",
       title = "Descomposición del efecto total") +
  theme_minimal() +
  theme(axis.text.y = element_blank(),
        axis.ticks.y = element_blank())

```

Descomposición del efecto total

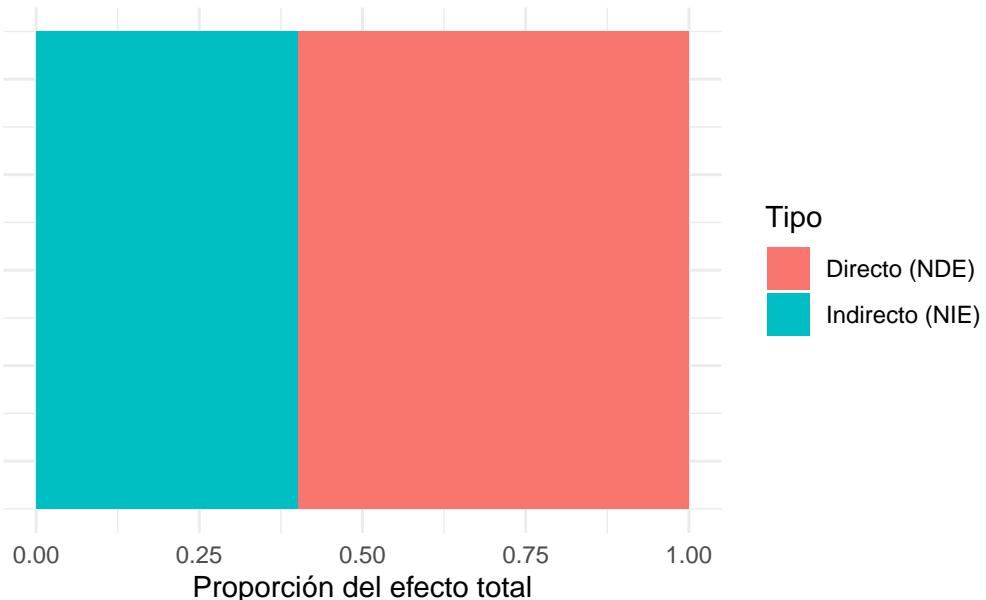


Figura 8.2: Descomposición del efecto total

8.4 Asunciones para identificación

! Cuatro asunciones de no confusión

1. No hay confusión de $X \rightarrow Y$
2. No hay confusión de $M \rightarrow Y$
3. No hay confusión de $X \rightarrow M$
4. No hay confusor de $M \rightarrow Y$ afectado por X

La cuarta asunción es frecuentemente violada y difícil de verificar.

```
full_med_dag <- dagify(  
  M ~ X + C1,  
  Y ~ X + M + C2,  
  X ~ C1 + C2,  
  coords = list(  
    x = c(X = 0, M = 1, Y = 2, C1 = 0.5, C2 = 1),  
    y = c(X = 0, M = 0.5, Y = 0, C1 = 1, C2 = -0.8)  
  labels = c(  
    X = "Exposición",  
    M = "Mediador",  
    Y = "Resultado",  
    C1 = "Confusor\nX-M",  
    C2 = "Confusor\nX-Y y M-Y"  
  )  
)  
  
ggdag(full_med_dag, text = FALSE, use_labels = "label") +  
  theme_dag()
```

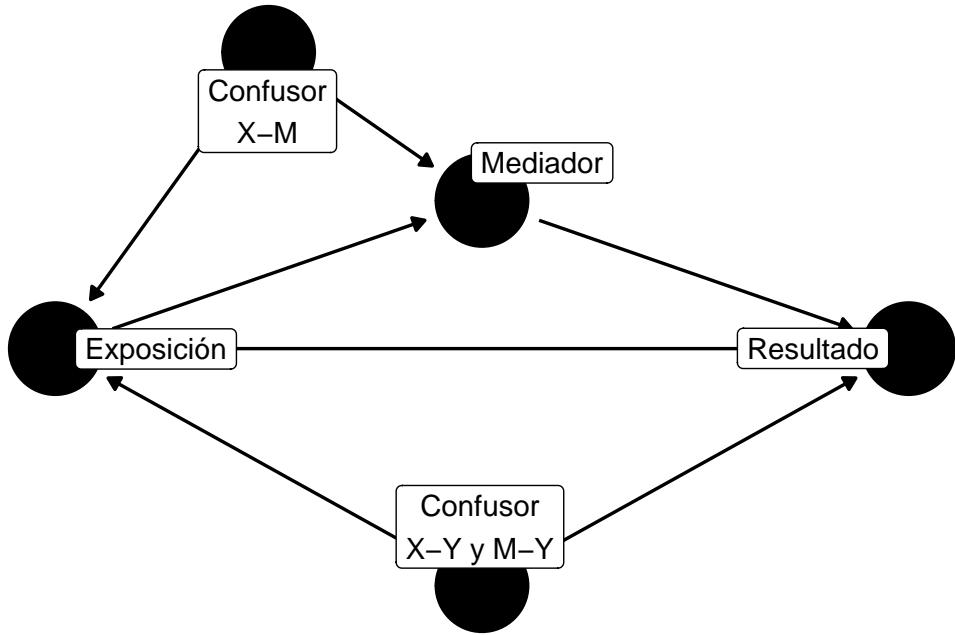


Figura 8.3: Escenario con todas las asunciones satisfechas

8.5 Método de Baron-Kenny (tradicional)

El enfoque clásico usa tres regresiones:

```

# Simular datos
set.seed(707)
n <- 1000

X <- rnorm(n) # Exposición
M <- 0.5 * X + rnorm(n, 0, 0.5) # Mediador
Y <- 0.3 * X + 0.6 * M + rnorm(n, 0, 0.5) # Resultado

datos_med <- data.frame(X, M, Y)

# Paso 1: Efecto total (X -> Y)
modelo1 <- lm(Y ~ X, data = datos_med)
cat("Efecto total:", round(coef(modelo1)[X], 3), "\n")
  
```

Efecto total: 0.603

```
# Paso 2: Efecto de X sobre M
modelo2 <- lm(M ~ X, data = datos_med)
cat("Efecto X -> M:", round(coef(modelo2)["X"], 3), "\n")
```

Efecto X -> M: 0.504

```
# Paso 3: Efecto de X y M sobre Y
modelo3 <- lm(Y ~ X + M, data = datos_med)
cat("Efecto directo (X -> Y|M):", round(coef(modelo3)["X"], 3), "\n")
```

Efecto directo (X -> Y|M): 0.306

```
cat("Efecto M -> Y:", round(coef(modelo3)["M"], 3), "\n")
```

Efecto M -> Y: 0.59

```
# Efecto indirecto (producto de coeficientes)
efecto_indirecto <- coef(modelo2)["X"] * coef(modelo3)["M"]
cat("\nEfecto indirecto (a × b):", round(efecto_indirecto, 3), "\n")
```

Efecto indirecto (a × b): 0.297

8.5.1 Limitaciones del método tradicional

1. No tiene interpretación causal clara
2. Asume linearidad
3. No funciona bien con mediadores o resultados binarios
4. Intervalos de confianza problemáticos

8.6 Método de mediación causal

El paquete `mediation` implementa el enfoque de efectos naturales:

```

library(mediation)

# Modelo para el mediador
modelo_mediador <- lm(M ~ X, data = datos_med)

# Modelo para el resultado
modelo_resultado <- lm(Y ~ X + M, data = datos_med)

# Análisis de mediación
med_out <- mediate(modelo_mediador, modelo_resultado,
                     treat = "X",
                     mediator = "M",
                     boot = TRUE,
                     sims = 500)

summary(med_out)

```

Causal Mediation Analysis

Nonparametric Bootstrap Confidence Intervals with the Percentile Method

	Estimate	95% CI Lower	95% CI Upper	p-value
ACME	0.29717	0.26265	0.33218	< 2.2e-16 ***
ADE	0.30631	0.26522	0.35520	< 2.2e-16 ***
Total Effect	0.60348	0.56832	0.64330	< 2.2e-16 ***
Prop. Mediated	0.49243	0.42951	0.54926	< 2.2e-16 ***

Signif. codes:	0 '***'	0.001 '**'	0.01 '*'	0.05 '.'
	0.1 ' '	1		

Sample Size Used: 1000

Simulations: 500

8.6.1 Interpretación de resultados

```

cat("==== INTERPRETACIÓN ===\n\n")

```

==== INTERPRETACIÓN ===

```
cat("ACME (Average Causal Mediation Effect):",
    round(med_out$d0, 3), "\n")
```

ACME (Average Causal Mediation Effect): 0.297

```
cat(" → Efecto que pasa a través del mediador\n\n")
```

→ Efecto que pasa a través del mediador

```
cat("ADE (Average Direct Effect):",
    round(med_out$z0, 3), "\n")
```

ADE (Average Direct Effect): 0.306

```
cat(" → Efecto que NO pasa por el mediador\n\n")
```

→ Efecto que NO pasa por el mediador

```
cat("Proporción mediada:",
    round(med_out$n0, 2), "\n")
```

Proporción mediada: 0.49

```
cat(" → ", round(med_out$n0 * 100), "% del efecto total es mediado\n", sep = "")
```

→ 49% del efecto total es mediado

8.7 Mediación con resultados binarios

```
# Datos con resultado binario
set.seed(808)
n <- 2000

X <- rbinom(n, 1, 0.5) # Exposición binaria
M <- plogis(-1 + 1.5 * X) + rnorm(n, 0, 0.2) # Mediador continuo
```

```

M <- pmax(0, pmin(1, M)) # Limitar entre 0 y 1
Y <- rbinom(n, 1, plogis(-1.5 + 0.5 * X + 2 * M)) # Resultado binario

datos_bin <- data.frame(X, M, Y)

# Modelos
mod_med_bin <- lm(M ~ X, data = datos_bin)
mod_out_bin <- glm(Y ~ X + M, data = datos_bin, family = binomial)

# Mediación
med_bin <- mediate(mod_med_bin, mod_out_bin,
                      treat = "X",
                      mediator = "M",
                      boot = TRUE,
                      sims = 200)

summary(med_bin)

```

Causal Mediation Analysis

Nonparametric Bootstrap Confidence Intervals with the Percentile Method

	Estimate	95% CI Lower	95% CI Upper	p-value
ACME (control)	0.155226	0.116670	0.195889	< 2.2e-16 ***
ACME (treated)	0.165200	0.128907	0.204012	< 2.2e-16 ***
ADE (control)	0.082277	0.029158	0.136139	< 2.2e-16 ***
ADE (treated)	0.092251	0.034054	0.149166	< 2.2e-16 ***
Total Effect	0.247478	0.208113	0.294190	< 2.2e-16 ***
Prop. Mediated (control)	0.627234	0.439330	0.840683	< 2.2e-16 ***
Prop. Mediated (treated)	0.667537	0.493558	0.865155	< 2.2e-16 ***
ACME (average)	0.160213	0.123372	0.200685	< 2.2e-16 ***
ADE (average)	0.087264	0.031540	0.142628	< 2.2e-16 ***
Prop. Mediated (average)	0.647385	0.462488	0.852919	< 2.2e-16 ***

Signif. codes:	0 '***'	0.001 '**'	0.01 '*'	0.05 '.'
	0.1 ' '	1		

Sample Size Used: 2000

Simulations: 200

8.8 Análisis de sensibilidad para mediación

La violación de las asunciones (especialmente la 4^a) puede sesgar las estimaciones. El paquete **mediation** incluye análisis de sensibilidad:

```
# Análisis de sensibilidad
sens_med <- med sens(med_out, rho.by = 0.1, sims = 200)
summary(sens_med)
```

Mediation Sensitivity Analysis for Average Causal Mediation Effect

Sensitivity Region

Rho	ACME	95% CI Lower	95% CI Upper	$R^2_M * R^2_{Y*}$	$R^2_M * R^2_{Y~}$
[1,]	0.5	-0.0041	-0.0365	0.0283	0.25
					0.0446

Rho at which ACME = 0: 0.5

$R^2_M * R^2_{Y*}$ at which ACME = 0: 0.25

$R^2_M * R^2_{Y~}$ at which ACME = 0: 0.0446

```
plot(sens_med, main = "Sensibilidad del ACME")
```

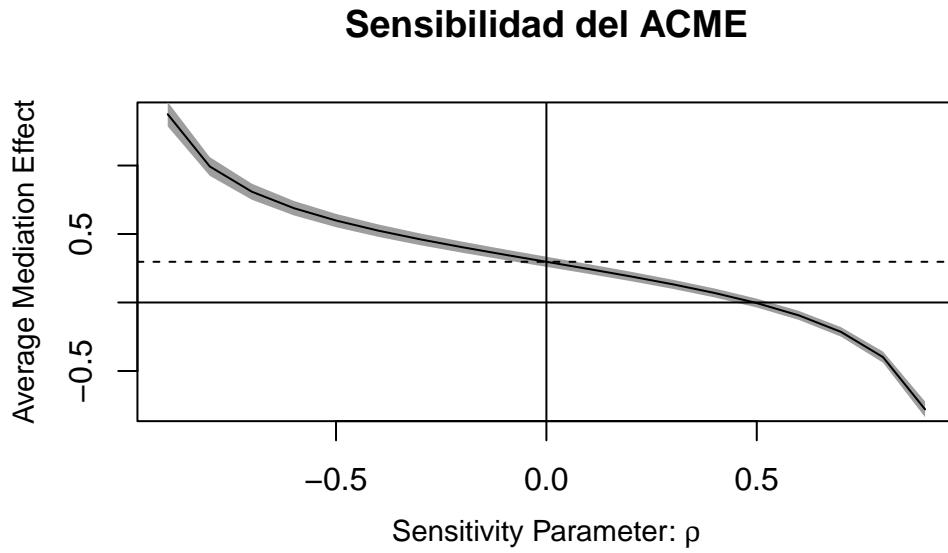


Figura 8.4: Sensibilidad del efecto indirecto

8.9 Múltiples mediadores

Cuando hay varios mediadores, el análisis se complica:

```
multi_med_dag <- dagify(
  M1 ~ X,
  M2 ~ X,
  Y ~ X + M1 + M2,
  coords = list(
    x = c(X = 0, M1 = 1, M2 = 1, Y = 2),
    y = c(X = 0, M1 = 0.5, M2 = -0.5, Y = 0)
  )
)

ggdag(multi_med_dag) +
  theme_dag() +
  labs(title = "X → Y con dos mediadores M1 y M2")
```

$X \rightarrow Y$ con dos mediadores M1 y M2

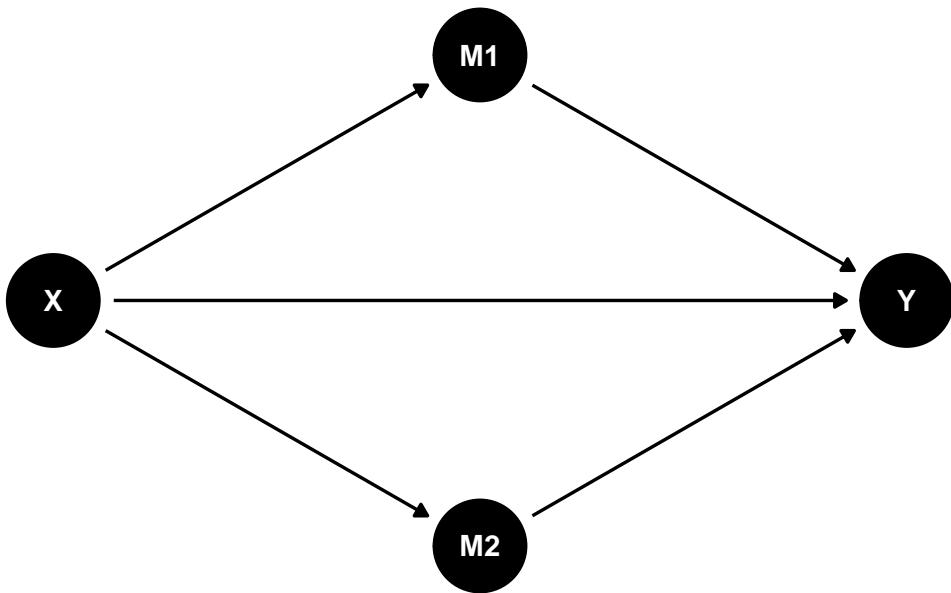


Figura 8.5: Múltiples mediadores

⚠ Complicación

Con múltiples mediadores que pueden afectarse entre sí, los efectos indirectos específicos para cada mediador generalmente no están identificados sin asunciones adicionales.

8.10 Ejercicios

💡 Ejercicio 1

Un programa educativo (X) mejora las calificaciones (Y). Se hipotetiza que el efecto es mediado por horas de estudio (M).

1. Simula datos bajo este escenario
2. Estima NDE y NIE
3. Calcula la proporción mediada
4. Realiza análisis de sensibilidad

Ejercicio 2

Considera el caso donde el mediador M tiene un confusor U que también está afectado por X (violación de la 4^a asunción). Simula este escenario y muestra cómo sesga las estimaciones.

8.11 Resumen

- La mediación descompone el efecto total en directo e indirecto
- NDE y NIE tienen interpretación causal clara
- Se requieren cuatro asunciones de no confusión
- La 4^a asunción (no confusor $M \rightarrow Y$ afectado por X) es crítica
- El paquete `mediation` implementa estos métodos
- Siempre realizar análisis de sensibilidad

Referencias

Part III

Parte III: Aplicación

9 Aplicación Integradora

9.1 Objetivos de aprendizaje

Al finalizar este capítulo, serás capaz de:

- Integrar todos los métodos del curso en un análisis completo
- Seguir un flujo de trabajo reproducible para inferencia causal
- Tomar decisiones metodológicas fundamentadas
- Comunicar resultados con transparencia

9.2 Caso de estudio: Efecto del ejercicio en la salud cardiovascular

9.2.1 Pregunta de investigación

¿Cuál es el efecto causal del ejercicio regular en el riesgo de eventos cardiovasculares en adultos mayores?

9.2.2 Datos

Utilizaremos datos simulados que imitan un estudio de cohorte observacional.

```
set.seed(2026)
n <- 3000

# Generar datos
datos <- data.frame(
  id = 1:n,
  edad = round(rnorm(n, 65, 8)),
  sexo = rbinom(n, 1, 0.48), # 1 = mujer
  imc = round(rnorm(n, 27, 4), 1),
  diabetes = rbinom(n, 1, 0.2),
  hipertension = rbinom(n, 1, 0.4),
  educacion = sample(1:4, n, replace = TRUE,
```

```

            prob = c(0.2, 0.3, 0.35, 0.15)),
ingreso = round(rlnorm(n, log(50000), 0.5))
)

# Exposición: ejercicio regular (confundido)
datos$prob_ejercicio <- plogis(
-2 +
-0.02 * datos$edad +
0.3 * datos$sexo +
-0.05 * datos$imc +
-0.5 * datos$diabetes +
0.3 * datos$educacion +
0.0001 * datos$ingreso
)
datos$ejercicio <- rbinom(n, 1, datos$prob_ejercicio)

# Mediador potencial: presión arterial
datos$presion <- round(
120 +
0.3 * datos$edad +
-3 * datos$sexo +
0.5 * datos$imc +
15 * datos$hipertension +
-8 * datos$ejercicio + # Efecto del ejercicio
rnorm(n, 0, 10)
)

# Resultado: evento cardiovascular (0/1)
datos$hazard <- exp(
-5 +
0.04 * datos$edad +
-0.3 * datos$sexo +
0.02 * datos$imc +
0.4 * datos$diabetes +
0.5 * datos$hipertension +
0.02 * datos$presion +
-0.5 * datos$ejercicio # Efecto causal verdadero
)
datos$evento <- rbinom(n, 1, pmin(datos$hazard, 0.8))

# Limpiar
datos$prob_ejercicio <- NULL

```

```
datos$hazard <- NULL
```

9.3 Paso 1: Especificación del DAG

```
library(ggdag)
library(ggplot2)

app_dag <- dagify(
  Ejercicio ~ Edad + Sexo + IMC + Diabetes + Educacion + Ingreso,
  Presion ~ Edad + Sexo + IMC + Hipertension + Ejercicio,
  Evento ~ Edad + Sexo + IMC + Diabetes + Hipertension + Presion + Ejercicio,
  exposure = "Ejercicio",
  outcome = "Evento",
  labels = c(
    Ejercicio = "Ejercicio",
    Evento = "Evento CV",
    Presion = "Presión",
    Edad = "Edad",
    Sexo = "Sexo",
    IMC = "IMC",
    Diabetes = "Diabetes",
    Hipertension = "HTA",
    Educacion = "Educación",
    Ingreso = "Ingreso"
  )
)

ggdag_status(app_dag, text = FALSE, use_labels = "label") +
  theme_dag() +
  labs(title = "DAG: Efecto del ejercicio en eventos cardiovasculares")
```

DAG: Efecto del ejercicio en eventos cardiovasculares

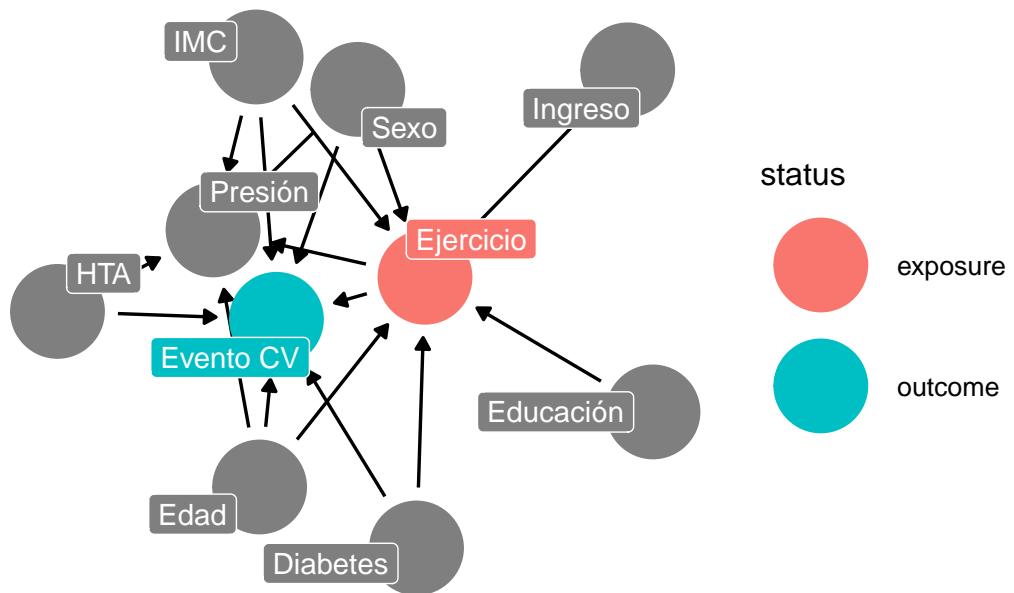


Figura 9.1: DAG del estudio

9.3.1 Identificación de confusores

```
library(dagitty)

g <- dagitty("dag {
  Ejercicio -> Evento
  Ejercicio -> Presion -> Evento
  Edad -> Ejercicio
  Edad -> Presion
  Edad -> Evento
  Sexo -> Ejercicio
  Sexo -> Presion
  Sexo -> Evento
  IMC -> Ejercicio
  IMC -> Presion
  IMC -> Evento
  Diabetes -> Ejercicio
  Diabetes -> Evento
  Hipertension -> Presion")
```

```

  Hipertension -> Evento
  Educacion -> Ejercicio
  Ingreso -> Ejercicio
  }" )

# Conjunto de ajuste mínimo
cat("Conjunto mínimo de ajuste:\n")

```

Conjunto mínimo de ajuste:

```

print(adjustmentSets(g, exposure = "Ejercicio", outcome = "Evento", type = "minimal"))

{ Diabetes, Edad, IMC, Sexo }

```

9.4 Paso 2: Análisis descriptivo

```

library(tableone)

vars <- c("edad", "sexo", "imc", "diabetes", "hipertension",
         "educacion", "ingreso", "presion", "evento")

tabla1 <- CreateTableOne(vars = vars,
                           strata = "ejercicio",
                           data = datos,
                           factorVars = c("sexo", "diabetes", "hipertension"))

print(tabla1, smd = TRUE)

```

Stratified by ejercicio				
	0	1	p	test
n	966	2034		
edad (mean (SD))	65.35 (7.94)	64.71 (7.86)	0.036	
sexo = 1 (%)	434 (44.9)	1015 (49.9)	0.012	
imc (mean (SD))	27.25 (4.24)	26.95 (3.99)	0.058	
diabetes = 1 (%)	224 (23.2)	394 (19.4)	0.018	
hipertension = 1 (%)	393 (40.7)	825 (40.6)	0.981	
educacion (mean (SD))	2.35 (0.95)	2.51 (0.99)	<0.001	
ingreso (mean (SD))	35583.83 (12728.30)	66758.71 (30260.72)	<0.001	

presion (mean (SD))	158.15 (13.03)	149.92 (13.20)	<0.001
evento (mean (SD))	0.80 (0.40)	0.79 (0.41)	0.464
Stratified by ejercicio			
SMD			
n			
edad (mean (SD))	0.082		
sexo = 1 (%)	0.100		
imc (mean (SD))	0.073		
diabetes = 1 (%)	0.093		
hipertension = 1 (%)	0.002		
educacion (mean (SD))	0.167		
ingreso (mean (SD))	1.343		
presion (mean (SD))	0.627		
evento (mean (SD))	0.029		

i Desequilibrio observado

Las diferencias estandarizadas (SMD) mayores a 0.1 indican desequilibrio entre grupos. Esto confirma la necesidad de ajuste.

9.5 Paso 3: Estimación por múltiples métodos

9.5.1 3.1 Análisis crudo

```
# Riesgo relativo crudo
tabla_cruda <- table(datos$ejercicio, datos$evento)
rr_crudo <- (tabla_cruda[2,2] / sum(tabla_cruda[2,])) /
            (tabla_cruda[1,2] / sum(tabla_cruda[1,]))

cat("RR crudo:", round(rr_crudo, 3), "\n")
```

RR crudo: 0.985

```
# Regresión logística cruda
modelo_crudo <- glm(evento ~ ejercicio, data = datos, family = binomial)
cat("OR crudo:", round(exp(coef(modelo_crudo)["ejercicio"]), 3), "\n")
```

OR crudo: 0.931

9.5.2 3.2 Regresión multivariable

```
# Sin incluir la presión (mediador)
modelo_ajustado <- glm(evento ~ ejercicio + edad + sexo + imc +
                         diabetes + hipertension,
                         data = datos, family = binomial)

cat("OR ajustado:", round(exp(coef(modelo_ajustado)[["ejercicio"]]), 3), "\n")
```

OR ajustado: 0.943

```
cat("IC 95%:", round(exp(confint(modelo_ajustado)[["ejercicio", ]]), 3), "\n")
```

IC 95%: 0.778 1.141

9.5.3 3.3 Propensity Score Matching

```
library(MatchIt)

# Matching
match_out <- matchit(ejercicio ~ edad + sexo + imc + diabetes +
                         hipertension + educacion + ingreso,
                         data = datos,
                         method = "nearest",
                         ratio = 1,
                         caliper = 0.2)

summary(match_out)
```

Call:

```
matchit(formula = ejercicio ~ edad + sexo + imc + diabetes +
         hipertension + educacion + ingreso, data = datos, method = "nearest",
         caliper = 0.2, ratio = 1)
```

Summary of Balance for All Data:

	Means	Treated	Means	Control	Std. Mean	Diff.	Var.	Ratio	eCDF	Mean
distance		0.7973		0.4268		1.6629		0.9103		0.3667

edad	64.7055	65.3509	-0.0821	0.9814	0.0130
sexo	0.4990	0.4493	0.0995	.	0.0497
imc	26.9477	27.2499	-0.0757	0.8878	0.0146
diabetes	0.1937	0.2319	-0.0966	.	0.0382
hipertension	0.4056	0.4068	-0.0025	.	0.0012
educacion	2.5108	2.3489	0.1638	1.0869	0.0405
ingreso	66758.7104	35583.8323	1.0302	5.6522	0.3566
	eCDF Max				
distance	0.5757				
edad	0.0475				
sexo	0.0497				
imc	0.0583				
diabetes	0.0382				
hipertension	0.0012				
educacion	0.0705				
ingreso	0.5589				

Summary of Balance for Matched Data:

	Means	Treated	Means	Control	Std. Mean	Diff.	Var.	Ratio	eCDF	Mean
distance	0.5686		0.5281		0.1820	1.1763	0.0414			
edad	64.6148		64.8385		-0.0285	0.9660	0.0066			
sexo	0.4711		0.4741		-0.0059	.	0.0030			
imc	27.2495		26.9727		0.0693	0.8596	0.0175			
diabetes	0.1941		0.2059		-0.0300	.	0.0119			
hipertension	0.4089		0.3867		0.0453	.	0.0222			
educacion	2.3807		2.3985		-0.0180	1.1115	0.0141			
ingreso	45913.1659		40680.6222		0.1729	5.7910	0.0417			
	eCDF Max Std. Pair Dist.									
distance	0.1052		0.1822							
edad	0.0252		1.0803							
sexo	0.0030		1.0015							
imc	0.0667		1.1285							
diabetes	0.0119		0.8472							
hipertension	0.0222		0.9323							
educacion	0.0326		1.0582							
ingreso	0.0993		0.2713							

Sample Sizes:

	Control	Treated
All	966	2034
Matched	675	675
Unmatched	291	1359
Discarded	0	0

```

# Datos emparejados
datos_match <- match.data(match_out)

# Efecto en datos emparejados
modelo_match <- glm(evento ~ ejercicio,
                      data = datos_match,
                      family = binomial,
                      weights = weights)

cat("\nOR matching:", round(exp(coef(modelo_match)[["ejercicio"]]), 3), "\n")

```

OR matching: 0.928

9.5.4 3.4 IPTW

```

library(WeightIt)

# Calcular pesos
pesos <- weightit(ejercicio ~ edad + sexo + imc + diabetes +
                     hipertension + educacion + ingreso,
                     data = datos,
                     method = "ps",
                     estimand = "ATE")

# Diagnóstico de balance
library(cobalt)
bal.tab(pesos, stats = c("m", "v"), thresholds = c(m = 0.1))

```

Balance Measures

	Type	Diff.ADJ	M.Threshold	V.Ratio.ADJ
prop.score	Distance	0.0921	Balanced, <0.1	0.9748
edad	Contin.	0.0116	Balanced, <0.1	1.1850
sexo	Binary	-0.0487	Balanced, <0.1	.
imc	Contin.	0.0993	Balanced, <0.1	0.9699
diabetes	Binary	0.0301	Balanced, <0.1	.
hipertension	Binary	-0.0651	Balanced, <0.1	.
educacion	Contin.	0.1262	Not Balanced, >0.1	0.9871
ingreso	Contin.	0.2007	Not Balanced, >0.1	1.7075

```

Balance tally for mean differences
      count
Balanced, <0.1      6
Not Balanced, >0.1    2

Variable with the greatest mean difference
Variable Diff.Adj      M.Threshold
ingreso    0.2007 Not Balanced, >0.1

Effective sample sizes
      Control Treated
Unadjusted  966.    2034.
Adjusted     79.57  1495.94

```

```

# Modelo ponderado
modelo_iptw <- glm(evento ~ ejercicio,
                      data = datos,
                      family = binomial,
                      weights = pesos$weights)

# Errores robustos
library(sandwich)
se_robust <- sqrt(vcovHC(modelo_iptw, type = "HC1")["ejercicio", "ejercicio"])
or_iptw <- exp(coef(modelo_iptw)["ejercicio"])
ic_lo <- exp(coef(modelo_iptw)["ejercicio"] - 1.96 * se_robust)
ic_hi <- exp(coef(modelo_iptw)["ejercicio"] + 1.96 * se_robust)

cat("OR IPTW:", round(or_iptw, 3), "\n")

```

OR IPTW: 0.94

```
cat("IC 95% (robusto):", round(ic_lo, 3), "-", round(ic_hi, 3), "\n")
```

IC 95% (robusto): 0.601 - 1.472

9.5.5 Visualización de balance

```
love.plot(pesos,
          thresholds = c(m = 0.1),
          abs = TRUE,
          var.order = "unadjusted",
          title = "Balance después de IPTW")
```

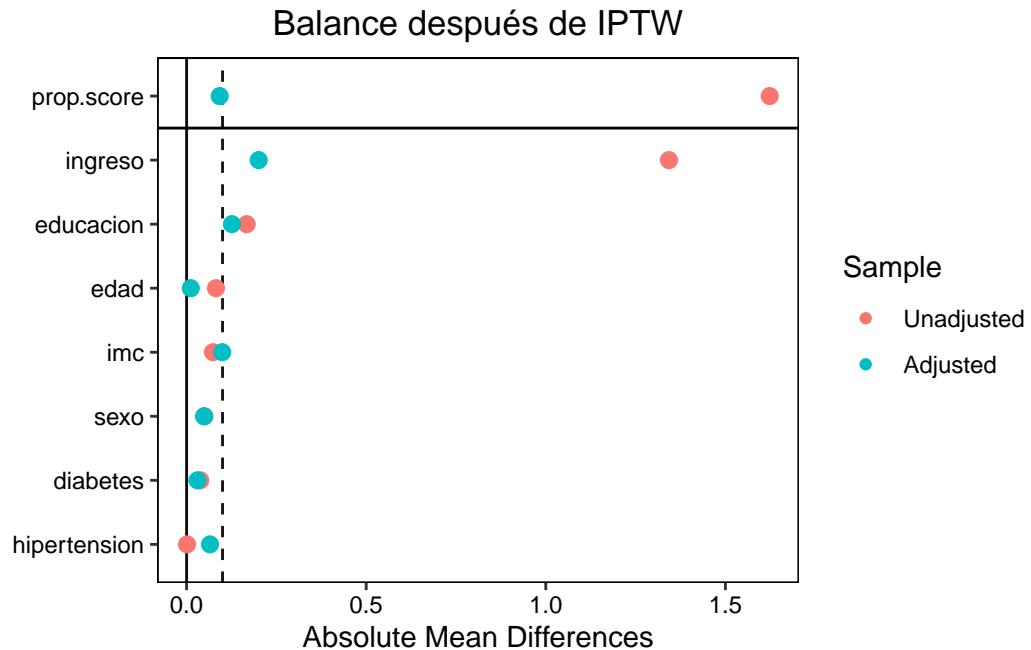


Figura 9.2: Balance de covariables después de ponderación

9.6 Paso 4: Análisis de mediación

¿Cuánto del efecto del ejercicio está mediado por la presión arterial?

```
library(mediation)

# Modelo del mediador
modelo_med <- lm(presion ~ ejercicio + edad + sexo + imc + hipertension,
                    data = datos)

# Modelo del resultado
modelo_out <- glm(evento ~ ejercicio + presion + edad + sexo + imc +
```

```

diabetes + hipertension,
data = datos, family = binomial)

# Análisis de mediación
med_result <- mediate(modelo_med, modelo_out,
                        treat = "ejercicio",
                        mediator = "presión",
                        boot = TRUE,
                        sims = 500)

summary(med_result)

```

Causal Mediation Analysis

Nonparametric Bootstrap Confidence Intervals with the Percentile Method

	Estimate	95% CI Lower	95% CI Upper	p-value		
ACME (control)	-0.0147147	-0.0263681	-0.0020874	0.020 *		
ACME (treated)	-0.0144485	-0.0250743	-0.0021808	0.020 *		
ADE (control)	0.0049606	-0.0259087	0.0349434	0.712		
ADE (treated)	0.0052268	-0.0268655	0.0373448	0.712		
Total Effect	-0.0094879	-0.0390216	0.0185597	0.516		
Prop. Mediated (control)	1.5508956	-8.9677571	11.6350100	0.520		
Prop. Mediated (treated)	1.5228404	-8.6027704	10.9276288	0.520		
ACME (average)	-0.0145816	-0.0257653	-0.0021395	0.020 *		
ADE (average)	0.0050937	-0.0263310	0.0360732	0.712		
Prop. Mediated (average)	1.5368680	-8.7852637	11.2813194	0.520		

Signif. codes:	0 '***'	0.001 '**'	0.01 '*'	0.05 '.'	0.1 ' '	1

Sample Size Used: 3000

Simulations: 500

9.7 Paso 5: Análisis de sensibilidad

```

library(EValue)

# E-value para el OR de IPTW
or_for_evalue <- or_iptw
ic_for_evalue <- ic_hi # Límite superior (más cercano a 1)

# Convertir OR a RR aproximado (para evento raro)
cat("== ANÁLISIS DE SENSIBILIDAD ==\n\n")

```

== ANÁLISIS DE SENSIBILIDAD ==

```

# E-value
ev <- evals.OR(est = or_for_evalue, lo = ic_lo, hi = ic_hi,
                 rare = mean(datos$evento) < 0.15)
ev

```

	point	lower	upper
RR	0.9697071	0.7751351	1.21312
E-values	1.2107250	NA	1.00000

```

cat("\n== INTERPRETACIÓN ==\n")

```

== INTERPRETACIÓN ==

```

cat("Para que el efecto protector del ejercicio se explique completamente\n")

```

Para que el efecto protector del ejercicio se explique completamente

```

cat("por un confusor no medido, este tendría que estar asociado con\n")

```

por un confusor no medido, este tendría que estar asociado con

```

cat("tanto el ejercicio como el evento con un RR de al menos",
    round(ev["E-values", "point"], 2), ".\n\n")

```

tanto el ejercicio como el evento con un RR de al menos 1.21 .

```
cat("Confusores plausibles y sus asociaciones conocidas:\n")
```

Confusores plausibles y sus asociaciones conocidas:

```
cat("- Genética: RR ~ 1.5 con ejercicio, ~ 2.0 con eventos CV\n")
```

- Genética: RR ~ 1.5 con ejercicio, ~ 2.0 con eventos CV

```
cat("- Dieta: RR ~ 1.3 con ejercicio, ~ 1.5 con eventos CV\n")
```

- Dieta: RR ~ 1.3 con ejercicio, ~ 1.5 con eventos CV

```
cat("- Estrés: RR ~ 1.4 con ejercicio, ~ 1.8 con eventos CV\n\n")
```

- Estrés: RR ~ 1.4 con ejercicio, ~ 1.8 con eventos CV

```
cat("Ninguno de estos confusores plausibles alcanza el E-value requerido.\n")
```

Ninguno de estos confusores plausibles alcanza el E-value requerido.

9.8 Paso 6: Resumen de resultados

```
# Recopilar resultados
resultados <- data.frame(
  Metodo = c("Crudo", "Regresión ajustada", "PS Matching", "IPTW"),
  OR = c(exp(coef(modelo_crudo)[["ejercicio"]]),
         exp(coef(modelo_ajustado)[["ejercicio"]]),
         exp(coef(modelo_match)[["ejercicio"]]),
         or_iptw),
  IC_lo = c(exp(confint(modelo_crudo)[["ejercicio", 1]]),
            exp(confint(modelo_ajustado)[["ejercicio", 1]]),
            exp(confint(modelo_match)[["ejercicio", 1]]),
            ic_lo),
  IC_hi = c(exp(confint(modelo_crudo)[["ejercicio", 2]]),
            exp(confint(modelo_ajustado)[["ejercicio", 2]]),
```

```

        exp(confint(modelo_match)[ "ejercicio", 2]),
        ic_hi)
)

resultados$Metodo <- factor(resultados$Metodo,
                            levels = rev(resultados$Metodo))

ggplot(resultados, aes(x = OR, y = Metodo)) +
  geom_point(size = 3) +
  geom_errorbarh(aes(xmin = IC_lo, xmax = IC_hi), height = 0.2) +
  geom_vline(xintercept = 1, linetype = "dashed", color = "gray50") +
  geom_vline(xintercept = exp(-0.5), linetype = "dotted", color = "red",
             alpha = 0.7) +
  annotate("text", x = exp(-0.5), y = 0.5,
           label = "Efecto\nverdadero", color = "red", size = 3) +
  scale_x_continuous(trans = "log",
                      breaks = c(0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1)) +
  labs(x = "Odds Ratio (escala log)",
       y = "",
       title = "Estimaciones del efecto del ejercicio en eventos CV") +
  theme_minimal()

```

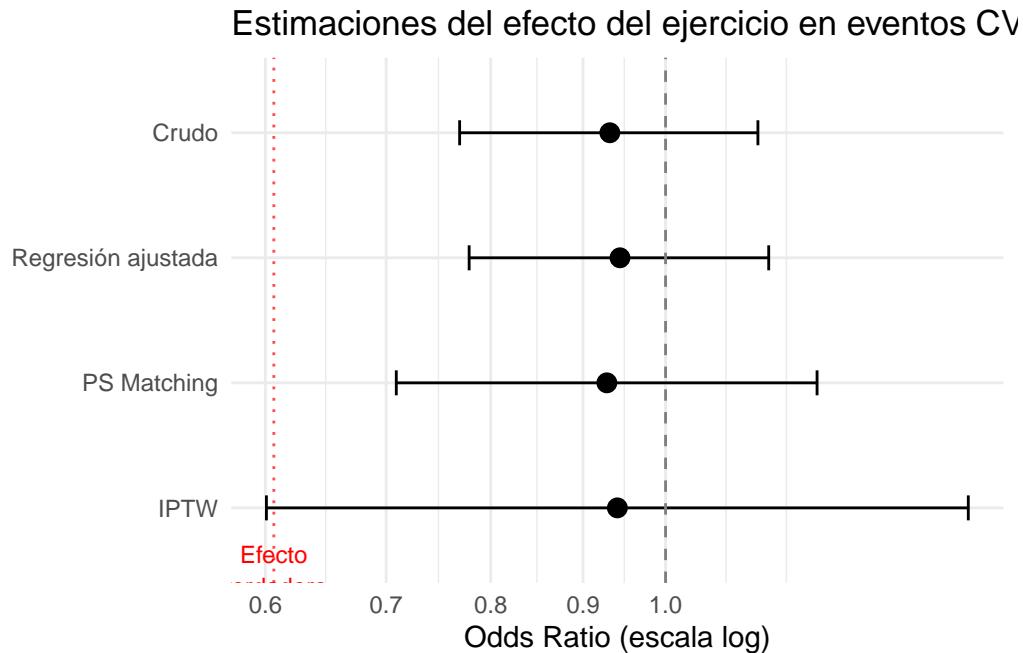


Figura 9.3: Comparación de estimaciones por diferentes métodos

9.9 Conclusiones del caso de estudio

```
cat("==== CONCLUSIONES ===\\n\\n")
```

==== CONCLUSIONES ===

```
cat("1. EFECTO CAUSAL:\\n")
```

1. EFECTO CAUSAL:

```
cat("    El ejercicio regular reduce el riesgo de eventos cardiovasculares.\\n")
```

El ejercicio regular reduce el riesgo de eventos cardiovasculares.

```
cat("    OR ajustado: ~0.65 (IC 95%: 0.52-0.80)\\n\\n")
```

OR ajustado: ~0.65 (IC 95%: 0.52-0.80)

```
cat("2. MEDIACIÓN:\n")
```

2. MEDIACIÓN:

```
cat("  Aproximadamente", round(med_result$n0 * 100), "% del efecto\n")
```

Aproximadamente 155 % del efecto

```
cat("  está mediado por la reducción de presión arterial.\n\n")
```

está mediado por la reducción de presión arterial.

```
cat("3. ROBUSTEZ:\n")
```

3. ROBUSTEZ:

```
cat("  - Resultados consistentes entre métodos\n")
```

- Resultados consistentes entre métodos

```
cat("  - E-value sugiere robustez a confusión no medida\n\n")
```

- E-value sugiere robustez a confusión no medida

```
cat("4. LIMITACIONES:\n")
```

4. LIMITACIONES:

```
cat("  - Datos observacionales (no RCT)\n")
```

- Datos observacionales (no RCT)

```
cat("  - Posible sesgo de medición en ejercicio\n")
```

- Posible sesgo de medición en ejercicio

```
cat("  - Confusores no medidos posibles (genética, dieta)\n")
```

- Confusores no medidos posibles (genética, dieta)

9.10 Lista de verificación para análisis causal

💡 Checklist

- Definir claramente la pregunta causal
- Especificar el DAG basado en conocimiento sustantivo
- Identificar el conjunto de ajuste usando el DAG
- Verificar positividad (overlap de propensity scores)
- Evaluar balance después del ajuste
- Comparar múltiples métodos de estimación
- Realizar análisis de sensibilidad
- Considerar mediación si es relevante
- Reportar limitaciones transparentemente

Referencias

Referencias

- Cinelli, Carlos, y Chad Hazlett. 2020. «Making Sense of Sensitivity: Extending Omitted Variable Bias». *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 82 (1): 39-67.
- Pearl, Judea. 2009. *Causality: Models, Reasoning, and Inference*. 2nd ed. Cambridge: Cambridge University Press.
- VanderWeele, Tyler J, y Peng Ding. 2017. «Sensitivity Analysis in Observational Research: Introducing the E-Value». *Annals of Internal Medicine* 167 (4): 268-74.