



IIC 2433 Minería de Datos

<https://github.com/marcelomendoza/IIC2433>

Cierre de la clase 10 – Intervenciones



Intervenciones:

Si queremos hacer un deconfounding ¿Qué es lo que debemos hacer?

¿Cuál es la diferencia entre estimación y estimando?

Front-door y back-door:

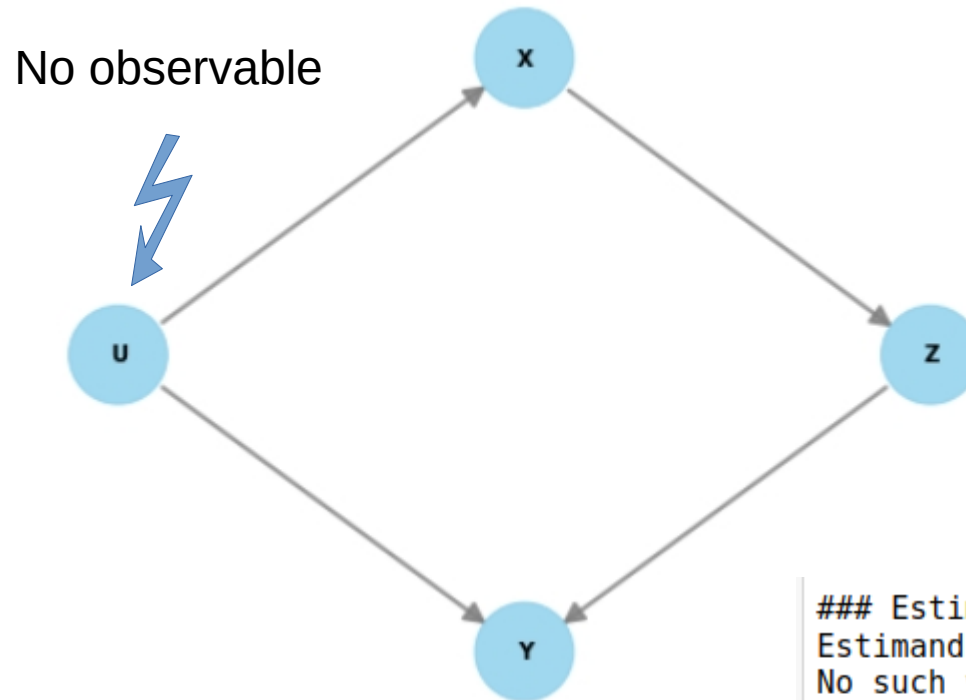
¿Qué establece el criterio back-door?

¿Qué es una variable mediadora?

¿Qué establece el criterio front-door?

Cierre de la clase 10 – Intervenciones

	X	Z	Y
0	4.112216	-2.145154	-0.416641
1	5.033208	-3.557968	-1.715015
2	2.072369	-0.980547	-0.156815
3	4.945853	-2.637407	-0.785717
4	6.507559	-3.719618	-1.959987
...
995	5.246228	-3.577730	-1.520773
996	6.546688	-4.235398	-1.465884
997	9.374855	-5.489424	-3.496549
998	9.176765	-4.474747	-2.941831
999	4.637124	-3.840523	-1.651353



Sólo front-door obtiene una expresión para el estimando

```
### Estimand : 1  
Estimand name: backdoor  
No such variable(s) found!
```

```
### Estimand : 2  
Estimand name: iv  
No such variable(s) found!
```

```
### Estimand : 3  
Estimand name: frontdoor  
Estimand expression:  

$$E \left[ \frac{d}{d[Z]}(Y) \cdot \frac{d}{d[X]}([Z]) \right]$$

```

Cierre de la clase 10 – Intervenciones

	X	Z1	Z2	Y
0	5.205841	0.244745	0.236655	0.812072
1	0.624142	1.050330	-0.599829	0.148622
2	7.295969	-0.527297	-0.091966	1.012046
3	3.802128	-0.328517	-0.569314	1.272041
4	2.384882	-0.364086	-0.572526	0.502528
...
995	9.861890	-2.081233	0.469809	0.645653
996	2.837282	0.815817	-0.406366	0.281442
997	4.261860	-0.055185	-0.268553	1.364996
998	5.739891	-2.075536	0.325745	0.737930
999	3.535107	-0.601190	1.153480	2.512154

Estimand type: EstimandType.NONPARAM

Estimand : 1
Estimand name: backdoor
No such variable(s) found!

Estimand : 2
Estimand name: iv
No such variable(s) found!

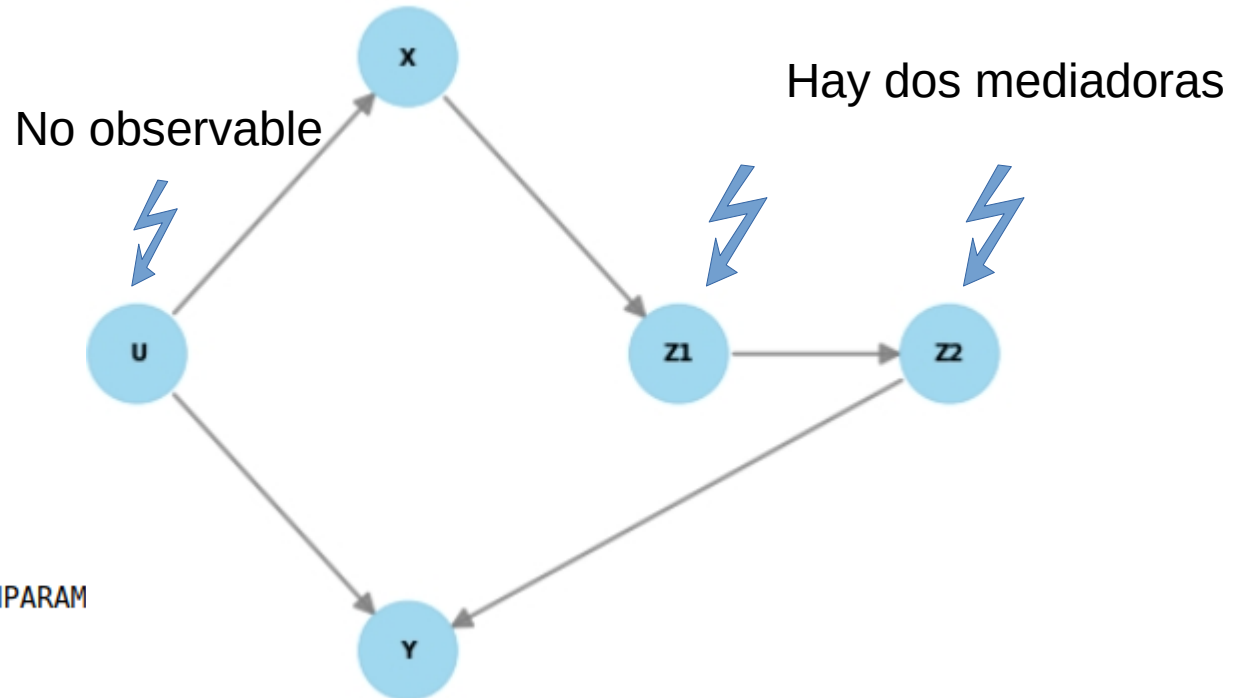
Estimand : 3
Estimand name: frontdoor
Estimand expression:

$$E \left[\frac{d}{d[Z_2, Z_1]}(Y) \cdot \frac{\partial}{\partial [X]}([Z_2, Z_1]) \right]$$

Estimand assumption 1, Full-mediation: Z2,Z1 intercepts (blocks) all directed paths from X to Y.

Estimand assumption 2, First-stage-unconfoundedness: If $U \rightarrow \{X\}$ and $U \rightarrow \{Z_2, Z_1\}$ then $P(Z_2, Z_1 | X, U) = P(Z_2, Z_1 | U)$

Estimand assumption 3, Second-stage-unconfoundedness: If $U \rightarrow \{Z_2, Z_1\}$ and $U \rightarrow Y$ then $P(Y | Z_2, Z_1, X, U) = P(Y | Z_2, Z_1, U)$



El criterio front-door
obtiene una expresión
para el estimando

Regresión lineal no va a funcionar aquí

- Matching, propensity scores y DR -

Estimar para medir el efecto de la intervención

Refutación

- Nos faltan algunos elementos aún para completar esta revisión de causalidad. En esta clase los veremos. Estos son **refutación** y **estimadores**.
- Para validar modelos causales usaremos **validación cruzada** y analizaremos la consistencia de los estimandos para cada setting.
- Al comparar los estimandos, estimaremos el **p-valor** de que ambos estimandos provengan de la misma distribución. Si los estimandos son similares, el p-valor será muy alto.

```
refute_subset = model.refute_estimate(  
    estimand=estimand,  
    estimate=estimate,  
    method_name="data_subset_refuter",  
    subset_fraction=0.4)
```

```
Refute: Use a subset of data  
Estimated effect:-0.4201729192182768  
New effect:-0.41971603098814647  
p value:0.98
```

← Ambos son similares, por lo que no podemos refutar el efecto causal

Estimadores

- Algunas veces las dependencias causales no serán estimables usando modelos lineales, como por ejemplo, regresión lineal (AF9).
- Veremos métodos para estimar efectos causales en casos no lineales.
- Comenzaremos por matching. Matching es una familia de métodos para estimar efectos causales haciendo matching entre observaciones similares en los grupos de tratamiento y no tratamiento.
- Hay varios tipos de matching. El más usual es el **matching inexacto**, el que permite aparear observaciones que son similares. Se requiere que la relación entre el tratamiento y la salida esté **deconfounded**, es decir, que **todos los confounders sean observables y que no se controlen colliders**.
- Vamos a comparar salida con y sin intervención y mediremos la diferencia entre ambos grupos:

The diagram shows the formula $\tau_i = Y_i^1 - Y_i^0$ enclosed in a black rectangular box. Two blue arrows originate from text labels below the box. One arrow points from the label 'Salida con intervención' to the Y_i^1 term. The other arrow points from the label 'Salida sin intervención' to the Y_i^0 term.

$$\tau_i = Y_i^1 - Y_i^0$$

Salida con intervención

Salida sin intervención

- UC - M. Mendoza -

Matching

- A partir de las diferencias observadas entre intervenir y no intervenir, mediremos ATE (average treatment effect):

$$ATE = \frac{1}{N} \sum_i \tau_i$$

donde N es el número total de observaciones.

- También podemos restringir a observaciones intervenidas, calculando el ATT (average treatment over treated units):

$$ATT = \frac{1}{N_{T=1}} \sum_{i_{T=1}} \tau_i$$

- También se define ATC, esto es medición de efecto sobre unidades no intervenidas. ATE es el promedio de ATT y ATC.

Matching

- El estimador de matching para ATE se define por:

$$\hat{\tau}_{ATE} = \frac{1}{N} \sum_i (2T_i - 1) (Y_i - Y_{j(i)})$$

- Se parean observaciones entre el grupo con y sin tratamiento. Para cada observación tratada (i), hacemos match sobre la observación no tratada j(i) o viceversa.
- T_i representa el valor de tratamiento para la observación i. El tratamiento es un booleano, si $T_i=1$ indica tratamiento, 0 en otro caso. Lo que hace entonces $(2T_i-1)$ es invertir el signo para distinguir entre tratamiento y no tratamiento al hacer la diferencia.
- Este estimador sólo funciona para tratamiento binario (interviene/no interviene).

Estimador ATE

Unit	Treatment	Outcome
A	1	9
B	0	1.5

$N=2$, A y B pareados bajo algún criterio

$$\hat{\tau}_{ATE} = \frac{1}{N} \sum_i (2T_i - 1)(Y_i - Y_{j(i)})$$

Unit A: $(2 \times 1 - 1)(9 - 1.5) = 1 \times 7.5 = 7.5$

Unit B: $(2 \times 0 - 1)(1.5 - 9) = -1 \times -7.5 = 7.5$

$$\hat{\tau}_{ATE} = \frac{1}{2} (7.5 + 7.5) = \frac{1}{2} \times 15 = 7.5$$

Estimador ATE

- Podemos relajar la definición del estimador ATE incluyendo pareo con M muestras cercanas, en lugar de parear con una sola.
- Para cada unidad i , recolectamos todos los $j(i)$ que satisfacen algún criterio a definir, y tomamos el promedio sobre ellas para hacer la comparación:

$$\hat{\tau}_{ATE} = \frac{1}{N} \sum_i \left(2 T_i - 1 \right) \left(Y_i - \left(\frac{1}{M} \sum_m Y_{j_m(i)} \right) \right)$$

- Ejemplo: Tres variables, **edad**, **toma un curso** de finanzas, y **ganancias** en la bolsa. Queremos saber si tomar el curso tiene un efecto causal en las ganancias de la persona. Toma el curso es booleano (toma/no toma).

	age	took_a_course	earnings
0	19	False	110579.0
1	28	False	142577.0
2	22	True	130520.0
3	25	True	142687.0
4	24	False	127832.0

Estimador ATE

- Vamos a agrupar por edad (criterio de pareo).

```
earnings_data.groupby(['age', 'took_a_course']).mean()
```

- Separamos en grupo de tratamiento (toma el curso) y grupo de control (no lo toma).

```
treatment_avg = earnings_data.query('took_a_course==1')['earnings'].mean()
cntrl_avg = earnings_data.query('took_a_course==0')['earnings'].mean()

treatment_avg - cntrl_avg
```

- Si medimos la diferencia directa desde los datos nos da 6695, que sería la ganancia a favor de quienes cursan la asignatura.

Estimador ATE

- Hagamoslo con modelos causales.

1) Modelamos el GCM. Edad puede influir en ambas variables (confounding):

```
nodes = ['took_a_course', 'earnings', 'age']
edges = [
    ('took_a_course', 'earnings'),
    ('age', 'took_a_course'),
    ('age', 'earnings')
]
```

...

```
model = CausalModel(
    data=earnings_data,
    treatment='took_a_course',
    outcome='earnings',
    graph=gml_string
)
```

Estimador ATE

2) Identificamos el efecto causal:

```
estimand = model.identify_effect()
```

```
### Estimand : 1
Estimand name: backdoor
Estimand expression:
      d
      ----- (E[earnings | age])
d[took_a_course]
Estimand assumption 1, Unconfoundedness: If  $U \rightarrow \{\text{took\_a\_course}\}$  and  $U \rightarrow \text{earnings}$  then  $P(\text{earnings} | \text{took\_a\_course}, \text{age}, U) = P(\text{earnings} | \text{took\_a\_course}, \text{age})$ 
```

Nos dice que el back-door es suficiente para estimar el efecto causal. Debemos controlar por edad.

Estimador ATE

3) Calculamos el estimando usando ATE:

```
estimate = model.estimate_effect(  
    identified_estimand=estimand,  
    method_name='backdoor.distance_matching',  
    target_units='ate',  
    method_params={'distance_metric': 'minkowski', 'p': 2})
```

Observemos que no hemos estandarizado el matching a edad.
Estamos usando una función de distancia.

El estimado es de 10118, mucho mayor que el mostrado al hacer la diferencia directa.

4) Refutamos:

```
refutation = model.refute_estimate(  
    estimand=estimand,  
    estimate=estimate,  
    method_name='random_common_cause')
```

```
Refute: Add a random common cause  
Estimated effect:10118.445  
New effect:10311.769699999999  
p value:0.42
```



Propensity scores

- Al usar match inexacto, se introduce un bias que depende del criterio usado para formar las unidades de comparación. Si tenemos muchas variables, hacer match sufrirá del efecto de la maldición de la dimensionalidad (lo vimos en las primeras clases).
- Para abordar este problema, usaremos **propensity scores**, que es la probabilidad de que una unidad dada sea asignada a un grupo de tratamiento basada en sus características.

$$\hat{e}(X) = P(T = 1 | X = x)$$

- Una propiedad de los propensity scores es que si tenemos unconfoundness dado X, también lo tenemos dado su propensity score.
- Los propensity scores son unidimensionales es decir, abordamos el problema de matching reduciendo la dimensionalidad a una sola dimensión.



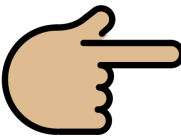
El propensity score será mayor en la medida que un tipo de dato sea más frecuente en las observaciones.

Propensity scores

- Los propensity scores tienen un problema. No los podemos usar directamente para matching ya que reducir información a una dimensión introduce mucha pérdida. Es decir, no son una buena alternativa para hacer matching directo. Sin embargo, los podemos usar en un estimador.
- Vamos a redefinir ATE usando propensity scores. A este nuevo estimador lo denominamos **Inverse Probability Weighting (IPW)**:

$$\hat{\tau}_{ATE_{IPW}} = \frac{1}{N_{T=1}} \sum_{i_{T=1}}^{N_{T=1}} \frac{y_i}{\hat{e}(X_i)} - \frac{1}{N_{T=0}} \sum_{j_{T=0}}^{N_{T=0}} \frac{y_j}{1 - \hat{e}(X_j)}$$

Al introducir el propensity score, hacemos upweighting de datos subrepresentados y downweighting de datos sobrerrepresentados. Esto introduce un debias por frecuencia de casos en el estimador.



En IPW, las unidades más propensas a ser intervenidas pesan menos en el estimador, mientras que las menos propensas pesan más.

Propensity scores

- Volvamos al ejemplo anterior, en el paso 3).

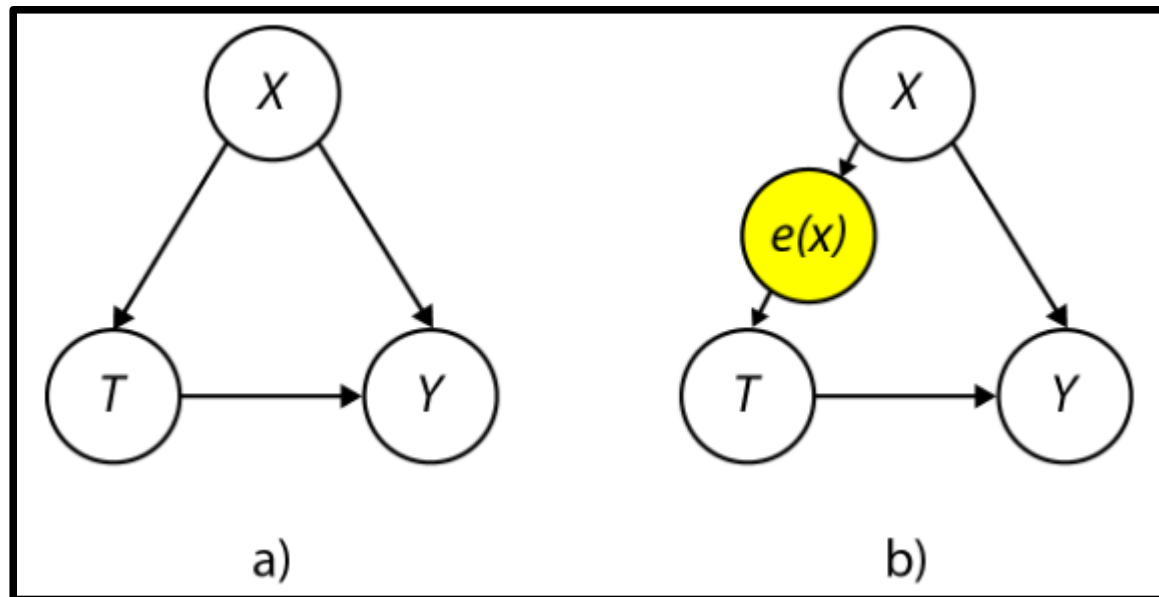
```
estimate = model.estimate_effect(  
    identified_estimand=estimand,  
    method_name='backdoor.propensity_score_weighting',  
    target_units='ate')
```

- El resultado es 10313, en línea con lo obtenido al usar ATE (consistente).

Doubly Robust (DR)

- IPW servirá para hacer deconfounding de data sólo si el confounder es observable. Lo mismo ocurriría con regresión, donde bastan con controlar la variable observada y bloquear caminos hacia o desde la variable de tratamiento. Al primero se le denomina **treatment model** y al segundo **outcome model**.

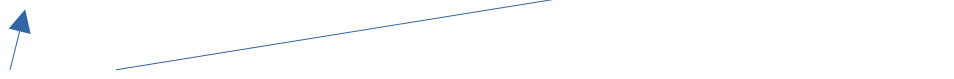
- Lo que estamos haciendo al incorporar el propensity score es los siguiente:



- Notemos que hay tres formas de cerrar el back-door de T. i) Controlar X, ii) Controlar $e(X)$, y iii) Controlar ambos. i) es ATE, ii) es IPW.

Doubly Robust (DR)

- ¿Por qué no combinamos i) y ii) de manera que elijamos el mejor de ambos *a posteriori*?
- DR hace esto. Nos entrega un estimador insesgado si es que alguno de los dos modelos (salida o tratamiento) está correctamente especificado:

$$\hat{\tau}_{ATE} = \frac{1}{N} \sum_i \left(\frac{T_i(Y_i - \mu_1(X_i))}{\hat{e}(X_i)} + \mu_1(X_i) \right) - \frac{1}{N} \sum_i \left(\frac{(1 - T_i)(Y_i - \mu_0(X_i))}{1 - \hat{e}(X_i)} + \mu_0(X_i) \right)$$


Modelos de salida que estiman el efecto causal

- Es decir, vamos a trabajar con modelos que estiman:

$$E[Y|X, T = t]$$

- Son modelos de salida basados en machine learning. Para propensity podemos usar regresión logística por ejemplo.

Doubly Robust (DR)

Averaging

$$\hat{t}_{ATE} = \frac{1}{N} \sum_i \left(\underbrace{\frac{T_i(Y_i - \mu_1(X_i))}{\hat{e}(X_i)} + \mu_1(X_i)}_{\mu_1 = \text{model with treatment}} \right) - \frac{1}{N} \sum_i \left(\underbrace{\frac{(1 - T_i)(Y_i - \mu_0(X_i))}{1 - \hat{e}(X_i)} + \mu_0(X_i)}_{\mu_0 = \text{model without treatment}} \right)$$

- ¿Por qué es robusto?

Combina tanto el modelo de salida como el de tratamiento en ATE. Veamos la parte tratada (naranja). Si el propensity score es 1, el efecto medido por el modelo de salida se cancela. Es decir, el modelo de salida tendrá peso en ATE en la medida que X_i tenga un propensity score bajo.