

---

## **Clase 183 — Bootstrap y permutation tests**

Parte: 3 — Estadística Inferencial y Causal · Fuente: ISLP, cap. 5 Resampling Methods + Efron & Tibshirani, An Introduction to the Bootstrap. Duración estimada: 85 min.

## Clase 183 — Bootstrap y permutation tests

Parte: 3 — Estadística Inferencial y Causal · Fuente: ISLP, cap. 5 Resampling Methods + Efron & Tibshirani, *An Introduction to the Bootstrap*. Duración estimada: 85 min.

### Objetivo

Sustituir los supuestos paramétricos (normalidad, homocedasticidad, fórmulas cerradas) por resampling: el bootstrap estima la distribución muestral de cualquier estadístico re-muestreando con reemplazo, y los permutation tests calculan un p-value re-mezclando etiquetas de tratamiento. Aprender a usar las APIs modernas de scipy (bootstrap, permutation\_test,  $\geq 1.9$ ) y a interpretar las tres variantes de IC bootstrap (percentil, basic, BCa).

### Resultados de aprendizaje

Al finalizar, el estudiante podrá:

- Implementar bootstrap a mano: B resamples con reemplazo del mismo tamaño que la muestra original, calcular el estadístico en cada uno, sacar cuantiles  $\alpha/2$  y  $1-\alpha/2$ .
- Usar `scipy.stats.bootstrap(data, statistic, n_resamples=9_999, method='BCa')` y entender por qué BCa corrige sesgo y asimetría.
- Diseñar un permutation test bilateral con `scipy.stats.permutation_test((a, b), statistic, n_resamples=10_000, alternative='two-sided')`.
- Saber cuándo usar bootstrap (IC de estadísticos no estándar: mediana,  $R^2$ , AUC) vs permutación (p-value de comparación entre grupos sin supuestos).
- Reconocer las limitaciones del bootstrap (muestra muy chica  $n < 20$ , dependencia temporal — usar block bootstrap).

### Temas

- Intuición del bootstrap: "tratá la muestra como si fuera la población y resampleá".
- Tres intervalos bootstrap: percentile, basic (reflejado), BCa (bias-corrected + accelerated).
- ¿Cuántos resamples?  $B = 10_000$  para IC95 % (los percentiles 2.5 y 97.5 se estabilizan).
- Permutation test: intercambiar etiquetas de tratamiento bajo  $H_0$  de "no diferencia".
- Diferencia conceptual: bootstrap estima variabilidad del estadístico; permutación produce un p-value exacto condicional a los datos.
- Block bootstrap para series temporales (preserva autocorrelación).
- Complemento moderno: APIs `scipy.stats.bootstrap` y `permutation_test` desde scipy 1.9, vectorizadas y con BCa por default.

### Versión profundizada — 2026

El tema moderno que vivía como complemento dentro de esta clase ahora tiene clase propia dedicada con patrón completo, ejercicios y homework:

- Clase 153b — BCa bootstrap y APIs modernas de scipy

## Definiciones y características

- Bootstrap (Efron 1979): re-muestreo con reemplazo de la muestra original, del mismo tamaño  $n$ . Cada resample produce un valor del estadístico; el conjunto aproxima la distribución muestral.
- B (número de resamples): 1 000 alcanza para SE; 10 000 para IC95 %; 100 000 para colas o p-values pequeños.
- Percentile IC: cuantiles ( $\alpha/2$ ,  $1-\alpha/2$ ) de la distribución bootstrap.
- Basic IC:  $(2 \cdot \hat{\theta} - q_{\{1-\alpha/2\}}, 2 \cdot \hat{\theta} - q_{\{\alpha/2\}})$ . Refleja respecto al estimador puntual.
- BCa IC: percentile ajustado por sesgo ( $z$ ) y aceleración ( $a$ ). Default moderno.
- Permutation test: bajo  $H_0$  de no efecto, las etiquetas son intercambiables. Re-mezclar las etiquetas y recalculando el estadístico genera la distribución exacta bajo  $H_0$ .
- p-value de permutación:  $(\# \text{ permutations con } |\text{stat}| \geq |\text{stat}_{\text{obs}}| + 1) / (n_{\text{resamples}} + 1)$ . El +1 se llama corrección de continuidad y evita  $p=0$ .
- Block bootstrap: para series temporales, resampla bloques contiguos en vez de observaciones individuales. Preserva autocorrelación.

## Dataset / recursos

- `seaborn.load_dataset('diamonds')`: bootstrap de la mediana del precio por cut.
- Modelos: AUC de un clasificador entrenado — IC bootstrap sobre la AUC en test.
- Librerías: `scipy.stats` ( $\geq 1.9$ ), `numpy`, `sklearn`.

## Ejercicios

1. Bootstrap a mano: para `tips.tip`, hacé `B=10_000` resamples con `rng.choice(x, size=len(x), replace=True)`, calculá la media, sacá los cuantiles 2.5 y 97.5. Verificá contra `scipy.stats.bootstrap(..., method='percentile')`.
2. BCa vs percentile: con datos lognormales `rng.lognormal(0, 1, 50)`, calculá IC de la mediana con `method='percentile'` y con `method='BCa'`. Comprubá que BCa es asimétrico hacia la cola derecha (refleja la asimetría real).
3. IC para AUC: entrenar un `LogisticRegression` en `breast cancer`, calcular AUC en test. Bootstrap `n_resamples=2_000` sobre `(y_true_test, y_proba_test)` con un `statistic` que devuelva `roc_auc_score`. Reportar IC95 % BCa.
4. Permutation test bilateral: con `tips.tip` por `sex`, ejecutá `scipy.stats.permutation_test` y comparalo con el `mannwhitneyu` de la Clase 150.
5. Cobertura: simulá 1 000 datasets de `Exp(1)` con `n=25`. Para cada uno, calculá IC95 % de la mediana con BCa y con percentile. Contá la cobertura empírica. BCa debería estar más cerca de 95 % que percentile.

## Homework verifiable

Sobre `diamonds`:

1. Bootstrap BCa (`B=10 000`) para la mediana de `price` global.
2. Bootstrap BCa para la diferencia de medianas entre `cut='Ideal'` y `cut='Fair'`.
3. Permutation test sobre la misma diferencia, p-value.
4. Reportar mediana  $\pm$  IC95 % BCa por categoría de `cut` (un gráfico con 5 puntos + bigotes).
5. Conclusión en 4 líneas: relacionar el p-value de permutación con el hecho de que el IC de la

diferencia no incluye 0.

Criterio de aceptación: la diferencia de medianas tiene IC95 % BCa positivo (Ideal < Fair en precio mediano, contraintuitivo — los diamantes Fair son más grandes), p por permutación < 0.001, y la conclusión menciona que IC y p coinciden cualitativamente.

## Errores comunes

Síntoma / mensaje	Causa y cómo arreglar
Bootstrap con n_resamples=100 y conclusion	Demasiado poco para IC. Fix: 10 000 mínimo
Aplico bootstrap a una serie temporal sin	Asume independencia → IC demasiado angosto
Reporto p=0 en un permutation test	Significa que ninguna permutación produjo
Uso bootstrap para n=8	Sesga el SE hacia abajo. Fix: bootstrap fu
Bootstrap del R <sup>2</sup> da IC negativo	Pasa cuando hay un mal fit; significa que

## Preguntas frecuentes

¿Bootstrap o cross-validation?

Distintos objetivos. CV estima el error de generalización de un modelo. Bootstrap estima la variabilidad de un estimador (cualquiera). Para "qué tan bueno es mi modelo en datos nuevos", CV. Para "qué IC tiene el AUC reportado", bootstrap.

¿Por qué BCa no es siempre el default?

Porque requiere jackknife (n recálculos del estadístico), lo cual es caro para modelos costosos. Para estadísticos baratos (media, mediana), siempre BCa. Para estadísticos caros (AUC de un modelo), percentile o basic puede ser un compromiso aceptable.

¿Permutation test es exacto?

Es exacto condicional a los datos observados: si hicieras todas las permutaciones (no una muestra de 10 000), el p-value sería exacto. Con 10 000 permutaciones, el SE del p-value es pequeño ( $\approx \sqrt{p(1-p)/n}$ ).

¿Bootstrap funciona para cualquier estadístico?

No para todos. Falla con estadísticos no suaves (máximo, percentiles extremos en datasets chicos). En esos casos, subsampling o jackknife son alternativas. Para estadísticos suaves (media, mediana, percentiles centrales, regresión), funciona excelente.

¿Y el out-of-bag de Random Forest es bootstrap?

Sí — Random Forest hace bootstrap sobre filas para cada árbol, y las muestras no incluidas (out-of-bag,  $\approx 37\%$  por árbol) se usan para estimar error sin necesidad de CV. Es bootstrap aplicado a ensembles.

## Referencias

- ISLP, cap. 5 — Resampling Methods.
- Efron, B. & Tibshirani, R. (1993), An Introduction to the Bootstrap, Chapman & Hall.
- Efron, B. (1987), Better Bootstrap Confidence Intervals, JASA — paper original de BCa.
- DiCiccio & Efron (1996), Bootstrap Confidence Intervals, Statistical Science.

- `scipy.stats.bootstrap` y `permutation_test`.
- `arch.bootstrap` — block bootstrap para series temporales.

## Siguiente clase

Clase 184 — BCa bootstrap y APIs modernas de `scipy`

## Apéndice: notebook (primer bloque)

Primera celda ejecutable del notebook de la clase.

```
# Imports y configuración inicial
```

## Archivos complementarios

- `notebook.ipynb`