

---

# Clase 184 — BCa bootstrap y APIs modernas de scipy

Parte: 3 — Estadística Inferencial y Causal · Fuente: Efron (1987) BCa + DiCiccio & Efron (1996) + `scipy.stats.bootstrap` docs. Duración estimada: 75 min.

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

Parte: 3 — Estadística Inferencial y Causal · Fuente: Efron (1987) BCa + DiCiccio & Efron (1996) + `scipy.stats.bootstrap docs`. Duración estimada: 75 min.

### Objetivo

Profundizar el BCa (Bias-Corrected and accelerated) bootstrap —el default moderno (Efron 1987)— y las APIs modernas de `scipy` (`scipy.stats.bootstrap`  $\geq 1.9$ , `scipy.stats.permutation_test`  $\geq 1.8$ ). Cubrir las correcciones que BCa hace sobre percentile clásico: bias correction (z) y acceleration (a) vía jackknife.

### Resultados de aprendizaje

Al finalizar, el estudiante podrá:

- Diferenciar bootstrap percentile vs basic vs BCa.
- Calcular z (bias correction) y a (acceleration) manualmente.
- Aplicar `scipy.stats.bootstrap((data,), statistic, method='BCa', n_resamples=10_000)`.
- Aplicar `permutation_test` para p-value de comparación de 2 grupos sin paramétrica.
- Reconocer cuándo BCa importa: estadísticos no lineales, distribuciones asimétricas.

### Temas

- Percentile bootstrap: cuantiles directos. Sub-cubre con asimetría.
- Basic bootstrap: reflexión  $2\theta - q_{\{1-\alpha/2\}}$ .
- BCa: corrige bias (z) y aceleración (a vía jackknife).
- Studentized bootstrap: estandariza con SE bootstrap del SE.
- `Scipy.stats.bootstrap` API.
- Permutation test exacto.

### Definiciones y características

- z:  $\Phi^{-1}(P(\theta^*_b \leq \theta))$  — fracción de bootstraps por debajo del estimador.
- a (acceleration): estima cómo SE depende del parámetro. Calculado con jackknife.
- BCa interval: percentiles ajustados  $\alpha$ ,  $\alpha$  función de z, a,  $z_{\{\alpha/2\}}$ .
- `scipy.stats.bootstrap` desde 1.9: vectorizado, BCa default, IC + SE + bias estimate.
- `permutation_test`: re-mezcla labels bajo H, calcula stat distribution.

### Dataset / recursos

- Datos lognormales sintéticos.
- `seaborn.load_dataset('diamonds')` para mediana de price.
- Librerías: `scipy.stats`, `numpy`, `matplotlib`.

### Ejercicios

1. Tres ICs: para mediana de  $x = \text{rng.lognormal}(0, 1, 100)$ , calcular IC con percentile, basic, BCa. Comparar.
2.  $z$  a mano: implementar  $z = \text{ppf}((B\_below\_theta) / B)$ . Verificar contra scipy.
3.  $a$  con jackknife: implementar leave-one-out para cada  $\theta_{(i)}$ . Calcular  $a$ .
4. Cobertura empírica: 1000 datasets  $\text{Exp}(1)$ ,  $n=25$ ; cobertura percentile vs BCa. BCa más cerca de 95 %.
5. `Permutation_test`: comparar dos lognormales con tamaño efecto chico. P-value exacto.

### Homework verificable

IC del AUC de un clasificador binario:

1. LogisticRegression en breast cancer. AUC en test.
2. Bootstrap BCa de  $(y\_test, y\_proba)$ : 5000 resamples.
3. Reportar AUC [BCa 95% CI].
4. Comparar con percentile bootstrap (más estrecho, sub-cubre).

Criterio de aceptación: BCa CI asimétrico (refleja asimetría de AUC cerca de 1.0); más amplio que percentile.

### Errores comunes

Síntoma / mensaje	Causa y cómo arreglar
BCa con <code>n_resamples=100</code>	Inestable. Fix: $\geq 5000$ , idealmente 10000.
Estadístico no vectorizable lento	Bootstrap es $O(B)$ . Fix: <code>vectorized=False</code> e
Bootstrap sobre serie temporal	Asume independencia. Fix: <code>block bootstrap</code>
<code>permutation_test n_resamples=999</code>	Resolución del p-value $1/(n+1)$ . Fix: <code>10_00</code>
Reportar percentile vs BCa indistintamente	BCa tiene cobertura nominal. Fix: <code>document</code>

### Preguntas frecuentes

Cuándo BCa importa?

Con estadísticos sesgados (mediana en asimetría) o  $n$  chico. Para media +  $n$  grande, percentile basta.

Studentized bootstrap mejor que BCa?

A veces. Requiere SE del SE  $\rightarrow$  bootstrap doble  $\rightarrow$  costoso. BCa es el compromiso pragmático.

Para IC de proporciones?

Wilson o Clopper-Pearson son específicos y mejores que bootstrap genérico.

`vectorized=True` en scipy?

Si tu statistic acepta `axis=`, sí — 100× más rápido.

Block bootstrap para series?

scipy no lo tiene; `arch.bootstrap.MovingBlockBootstrap` sí.

### Referencias

- Efron (1987), Better Bootstrap Confidence Intervals, JASA.
- DiCiccio & Efron (1996), Bootstrap Confidence Intervals, Statistical Science.
- `scipy.stats.bootstrap`.
- `scipy.stats.permutation_test`.

## Siguiente clase

Clase 185 — A/B testing: tamaño de muestra, poder estadístico

## Apéndice: notebook (primer bloque)

BCa (bias-corrected accelerated) corrige sesgo y asimetría del bootstrap percentil. Permutation test = inferencia exacta sin asumir distribución. Requiere: `pip install numpy scipy`.

```
import numpy as np
from scipy import stats

rng = np.random.default_rng(42)
# Distribución sesgada (lognormal) — donde percentil bootstrap subperforma
data = rng.lognormal(mean=1.0, sigma=0.6, size=80)
true_mean = np.exp(1.0 + 0.6**2/2) # E[lognormal]
print(f'sample mean = {data.mean():.3f} | true mean = {true_mean:.3f}')
```

## Archivos complementarios

- `notebook.ipynb`