

Hoja informativa: Implementación de una nueva funcionalidad

Práctica

```
# Encontrar el intervalo de confianza para la media
# alpha: nivel de significación;
# df: número de grados de libertad, = n - 1;
# loc: distribución media, igual a la estimación media.
# muestra: sample.mean();
# escala: error estándar de distribución, igual a la estimación del error estándar.
# Cálculo: sample.sem()

from scipy import stats as st

confidence_interval = st.t.interval(alpha = alpha, df=df,
                                   loc=sample.mean(), scale=sample.sem())
```

```
# Extracción de submuestra para bootstrap

from numpy.random import RandomState
state = RandomState(12345)

# sin reemplazo
print(example_data.sample(frac=1, replace=False, random_state=state))
# con reemplazo
print(example_data.sample(frac=1, replace=True, random_state=state))
```

```
# A/B test analysis using bootstrap

import pandas as pd
import numpy as np

# actual difference between the means in the groups
AB_difference = samples_B.mean() - samples_A.mean()

alpha = 0.05

state = np.random.RandomState(54321)
```

```

bootstrap_samples = 1000
count = 0
for i in range(bootstrap_samples):
    # calcula cuántas veces excederá la diferencia entre las medias
    # el valor actual, siempre que la hipótesis nula sea cierta
    united_samples = pd.concat([samples_A, samples_B])
    subsample = united_samples.sample(frac=1, replace=True, random_state=state)

    subsample_A = subsample[:len(samples_A)]
    subsample_B = subsample[len(samples_A):]
    bootstrap_difference = subsample_B.mean() - subsample_A.mean()

    if bootstrap_difference >= AB_difference:
        count += 1

pvalue = 1. * count / bootstrap_samples
print('p-value =', pvalue)

if pvalue < alpha:
    print("La hipótesis nula se rechaza, a saber, es probable que el importe promedio de las compras aumente")
else:
    print("La hipótesis nula no se rechaza, a saber, es poco probable que el importe medio de las compras aumente")

```

Teoría

Las **pruebas A/B** o **split testing** son una técnica de comprobación de hipótesis que ayuda a controlar el impacto que provocan los cambios de un servicio o producto sobre los usuarios. La técnica implica lo siguiente: la población se divide en el grupo de control que usa el servicio regular sin cambios y el grupo experimental que usa la nueva versión, la que necesitamos probar.

Peeking problem: el resultado general se distorsiona cuando se agregan nuevos datos al comienzo del experimento.

Error de tipo I: se produce cuando la hipótesis nula es correcta, pero se rechaza (resultado *falso positivo*. En este caso, la nueva funcionalidad se aprueba y, por lo tanto, es *positiva*)

Error de tipo II: se produce cuando la hipótesis nula es incorrecta, pero se acepta (resultado *falso negativo*)

Un **intervalo de confianza** representa un segmento del eje numérico dentro del que cae el parámetro poblacional de interés, con una probabilidad predeterminada.