KIBERNUM
IT Academy

Módulo 4 – Inferencia Estadística

Teorema Límite Central

Ciencia de Datos

Objetivos de Aprendizaje

- Utiliza los conceptos básicos de estadística Inferencial.
- Explicar ley de los grandes números.
- Explicar Teorema Límite Central.
- Realizar cálculos de probabilidad utilizando la distribución muestral para resolver un problema.

Contenido

1. Ley de los grandes números
2. Teorema del Límite Central.
3. Distribución muestral.

KIBERNUM

Ley de los Grandes Números

KIBERNUM

Ley de los Grandes Números

La ley de los grandes números es un teorema fundamental de la teoría de la probabilidad que establece que, **a medida que el tamaño de una muestra aleatoria aumenta, la media de la muestra se acerca cada vez más a la media poblacional**. En otras palabras, a medida que se realizan más observaciones, la probabilidad de que la media de la muestra difiera significativamente de la media poblacional disminuye.

Ley de los Grandes Números

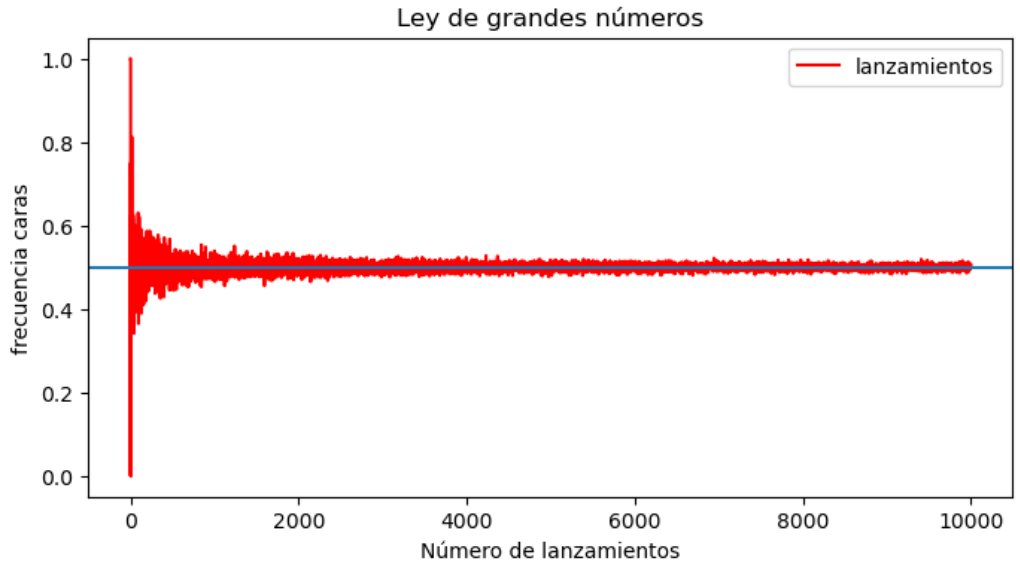
Un ejemplo de la ley de los grandes números es lanzar una moneda justa varias veces. Si lanzamos la moneda muchas veces, esperamos que la proporción de veces que la moneda cae cara se acerque a 0,5 a medida que aumenta el número de lanzamientos.

Por ejemplo, si lanzamos la moneda 10 veces, es posible que obtengamos 6 caras y 4 cruces, lo que equivale a una proporción del 60% de caras. Sin embargo, si lanzamos la moneda 100 veces, es más probable que obtengamos una proporción más cercana al 50% de caras, como 52 caras y 48 cruces. Si lanzamos la moneda 1000 veces, es aún más probable que la proporción se acerque al 50%, por ejemplo, 502 caras y 498 cruces.

A medida que se lanzan la moneda más veces, la proporción de caras se acerca cada vez más a 0,5, lo que ilustra la ley de los grandes números. En otras palabras, cuanto mayor sea el número de lanzamientos, más se acercará la proporción de caras a la probabilidad verdadera de obtener una cara al lanzar una moneda justa, que es del 50%.

Ley de los Grandes Números

Este experimento lo podemos comprobar con Python:



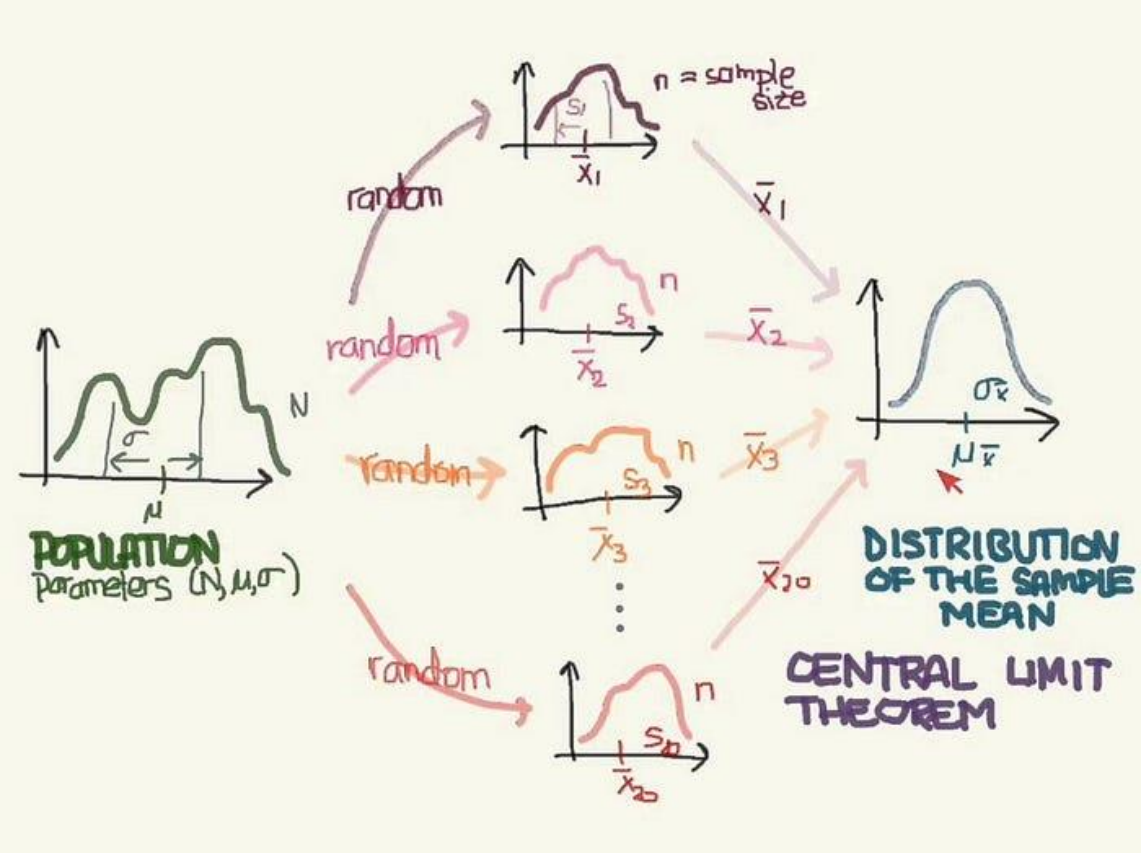

```python
# Ejemplo Ley de los Grandes Números
# moneda p=1/2 cara=1 sello=0
resultados = []

for n_lanzamientos in range(1, 10000):
    sample = np.random.choice([0, 1], size=n_lanzamientos)  # lanzamientos independientes
    caras = sample.mean()  # proporción de caras
    resultados.append(caras)

# Graficando
df = pd.DataFrame({'lanzamientos': resultados})
df.plot(title='Ley de los Grandes Números', color='r', figsize=(8, 4))
plt.axhline(0.5)  # valor esperado
plt.xlabel("Número de lanzamientos")
plt.ylabel("Frecuencia de caras")
plt.show()
```

Teorema del Límite Central

Teorema del Límite Central



Fuente:
https://towardsdatascience.com/central-limit-theorem-simulation-with-python-c80f8d3a6755

El teorema del límite central (TLC) es un teorema fundamental en estadística que establece que, bajo ciertas condiciones, **la distribución de las medias muestrales se aproxima a una distribución normal a medida que el tamaño de la muestra aumenta**.

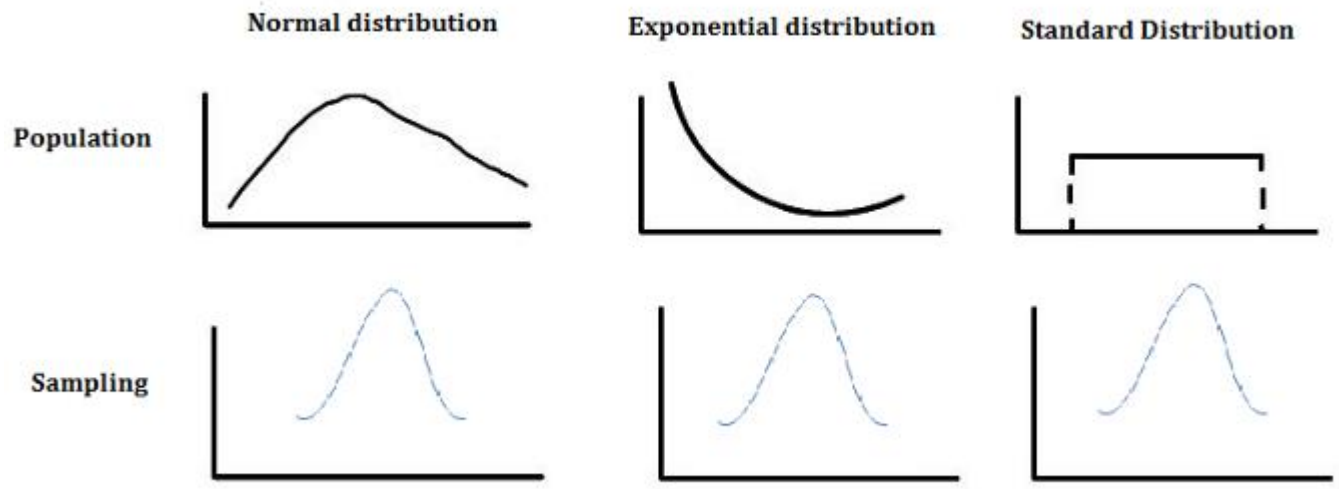
Teorema del Límite Central

El teorema del límite central es importante en estadística porque **permite hacer inferencias sobre la media o la proporción de una población a partir de una muestra aleatoria**. Debido a que la distribución normal es bien conocida y fácilmente interpretable, el TLC permite hacer estimaciones y cálculos precisos sobre una población a partir de una muestra.

Es importante tener en cuenta que el teorema del límite central se aplica solo si se cumplen ciertas condiciones, como la aleatoriedad de la muestra y la independencia de las observaciones. Además, en algunos casos, puede ser necesario usar una corrección si el tamaño de la muestra es pequeño o la distribución de probabilidad de la población es muy asimétrica.

Teorema del Límite Central

Independiente de la distribución original, la distribución de los promedios de todas las posibles muestras de tamaño n será siempre de tipo normal.



Teorema del Límite Central

TEOREMA:

Sea X_1, X_2,…, X_n un conjunto de variables aleatorias, independientes e idénticamente distribuidas de una distribución con medias μ y varianza $\sigma^2 \neq 0$

Entonces, si n es suficientemente grande, la variable aleatoria:

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

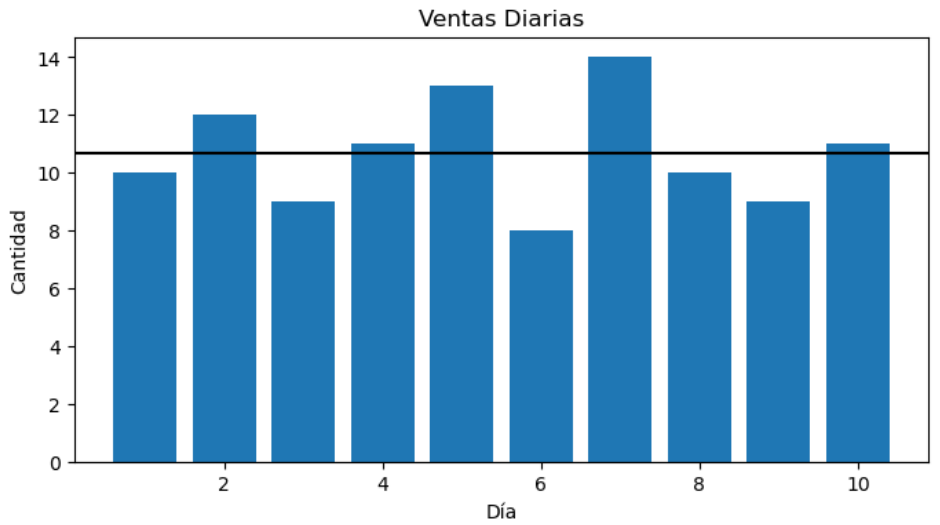
Tiene aproximadamente una distribución normal con:

$$\mu_{\bar{X}} = \mu \quad \text{y} \quad \sigma^2_{\bar{X}} = \frac{\sigma^2}{n}$$

Ejemplo

Imagina que tienes una empresa que vende productos en línea y quieres analizar el número de ventas por día. Durante un mes, has estado registrando el número de ventas diarias y has obtenido **los siguientes resultados:**

```
Día 1: 10 ventas
Día 2: 12 ventas
Día 3: 9 ventas
Día 4: 11 ventas
Día 5: 13 ventas
Día 6: 8 ventas
Día 7: 14 ventas
Día 8: 10 ventas
Día 9: 9 ventas
Día 10: 11 ventas
```

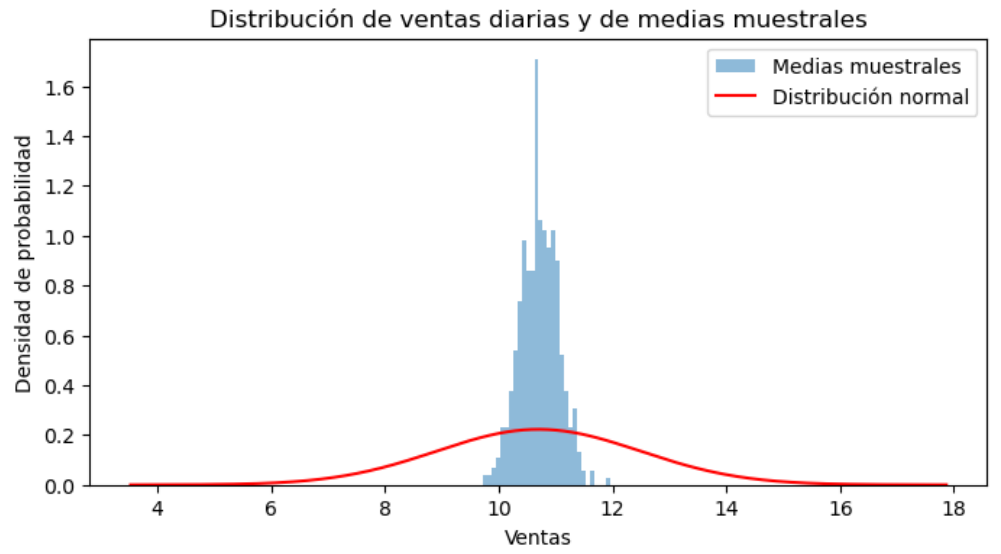


Puedes calcular la media de ventas diarias para este período de 10 días, que es:
(10 + 12 + 9 + 11 + 13 + 8 + 14 + 10 + 9 + 11) / 10 = 11 ventas por día.

Ejemplo

El teorema del límite central establece que, en una muestra grande de observaciones independientes de una misma variable aleatoria, la distribución de la media muestral se aproxima a una distribución normal, independientemente de la distribución original. En otras palabras, si tomamos muestras de tamaño suficientemente grande de las ventas diarias, la distribución de las medias muestrales se parecerá cada vez más a una distribución normal, incluso si la distribución original de las ventas diarias no es normal.

Así que, si tomamos muchas muestras de tamaño suficientemente grande de las ventas diarias, la distribución de las medias muestrales se aproximará a una distribución normal.



Ejemplo

Podemos entonces utilizar esta distribución normal para hacer inferencias estadísticas y para responder preguntas como "¿Cuál es la probabilidad de tener menos de 9 ventas en un día determinado?"

Por lo tanto, el teorema del límite central es una herramienta valiosa para hacer inferencias estadísticas y para entender mejor la distribución de una variable aleatoria.

Ejemplo

Código Python.

```python
import numpy as np
import matplotlib.pyplot as plt
```

```python
# Datos de ventas diarias
ventas_diarias = [10, 12, 9, 11, 13, 8, 14, 10, 9, 11]

# Media y desviación estándar de las ventas diarias
media = np.mean(ventas_diarias)
desviacion_estandar = np.std(ventas_diarias)

# Generar 1000 muestras de tamaño 30 de las ventas diarias
muestras = []
for i in range(1000):
    muestra = np.random.choice(ventas_diarias, size=30)
    muestras.append(np.mean(muestra))

plt.figure(figsize=(8,4))

# Graficar la distribución de las medias muestrales
plt.hist(muestras, bins=30, density=True, alpha=0.5, label='Medias muestrales')

# Calcular la media y la desviación estándar de las medias muestrales
media_muestral = np.mean(muestras)
desviacion_estandar_muestral = np.std(muestras)

# Graficar la distribución normal aproximada
x = np.linspace(media - 4*desviacion_estandar, media + 4*desviacion_estandar, 1000)
y = 1/(desviacion_estandar*np.sqrt(2*np.pi)) * np.exp(-(x-media)**2/(2*desviacion_estandar**2))


plt.plot(x, y, color='r', label='Distribución normal')

plt.title('Distribución de ventas diarias y de medias muestrales')
plt.xlabel('Ventas')
plt.ylabel('Densidad de probabilidad')
plt.legend()
plt.show()
```

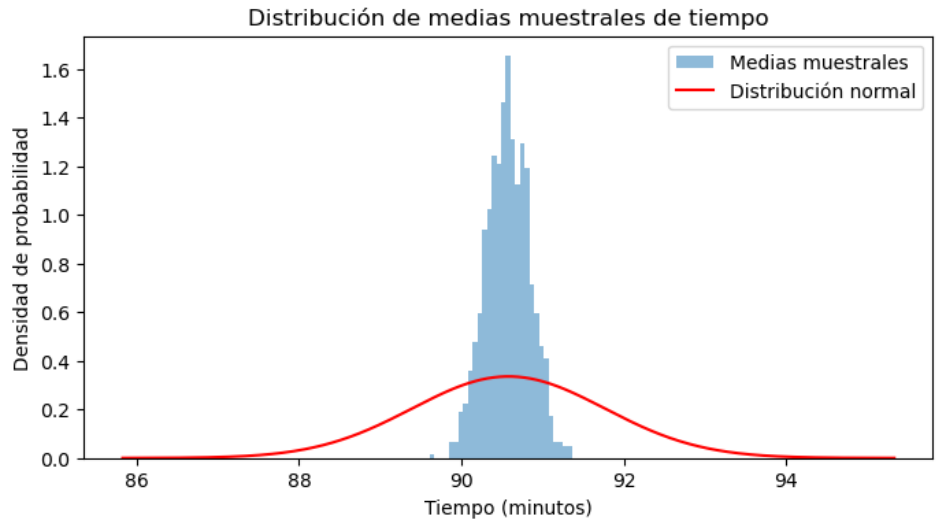
Ejemplo 2

Supongamos que estás interesado en estudiar el tiempo que tarda una persona promedio en completar una prueba de matemáticas en línea. Realizas un experimento en el que haces que 100 personas diferentes tomen la prueba y registras el tiempo que tardan en completarla. Supongamos que los tiempos que obtuviste son los siguientes:

[90.2, 89.5, 92.1, 88.6, 91.3, 89.8, 90.5, 91.9, 89.2, 92.5, 88.7,
90.1, 91.8, 89.9, 91.2, 90.8, 88.3, 92. , 90.7, 89.1, 91.6, 89.7,
90.3, 91.1, 90.6, 89.8, 92.2, 89.6, 90.4, 91.7, 90. , 88.9, 92.3,
89.4, 91. , 90.9, 88.5, 92.4, 89.3, 91.4, 91.5, 88.8, 90.6, 92.6,
91.2, 90.2, 89.1, 92.1, 89.5, 91.9, 90.3, 88.7, 92.5, 90.8, 89.6,
91.6, 89.8, 91.1, 90.5, 89.9, 91.8, 90.4, 89.7, 91. , 92.2, 88.9,
91.7, 90. , 89.2, 92.3, 90.7, 91.4, 91.5, 88.6, 92.4, 89.4, 90.1,
91.2, 90.6, 88.3, 92. , 89.3, 91.3, 91.6, 88.8, 90.6, 92.6, 91.1,
89.8, 90.2, 92.5, 91.9, 90.5, 89.9, 91.8, 88.7, 91.6, 90.3, 90.]

Ejemplo 2

Obtenemos el siguiente resultado en Python:




```python
# Media y desviación estándar de las ventas diarias
media = np.mean(tiempo)
desviacion_estandar = np.std(tiempo)

muestras = []
for i in range(1000):
    muestra = np.random.choice(tiempo, size=20)
    muestras.append(np.mean(muestra))


plt.figure(figsize=(8,4))

# Graficar la distribución de las medias muestrales
plt.hist(muestras, bins=30, density=True, alpha=0.5, label='Medias muestrales')

# Calcular la media y la desviación estándar de las medias muestrales
media_muestral = np.mean(muestras)
desviacion_estandar_muestral = np.std(muestras)

# Graficar la distribución normal aproximada
x = np.linspace(media - 4*desviacion_estandar, media + 4*desviacion_estandar, 1000)
y = 1/(desviacion_estandar*np.sqrt(2*np.pi)) * np.exp(-(x-media)**2/(2*desviacion_estandar**2))


plt.plot(x, y, color='r', label='Distribución normal')

plt.title('Distribución de medias muestrales de tiempo')
plt.xlabel('Tiempo (minutos)')
plt.ylabel('Densidad de probabilidad')
plt.legend()
plt.show()
```

Teorema del Límite Central

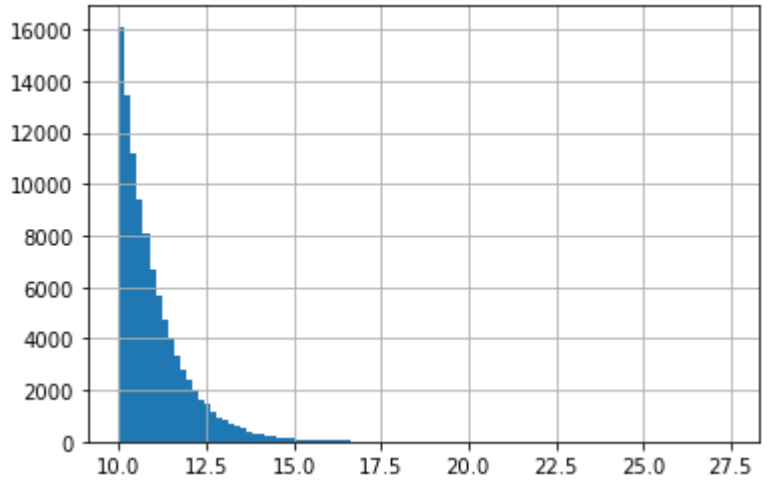
¿Y esto funciona si tomo muestras que se distribuyen con otras distribuciones?
por ejemplo, una exponencial

Teorema del Límite Central

Tomemos números aleatorios de una distribución exponencial.

```
poblacion_expo = pd.DataFrame()
poblacion_expo['number'] = expon.rvs(10, size = 100000)

poblacion_expo['number'].hist(bins=100)
```



Teorema del Límite Central

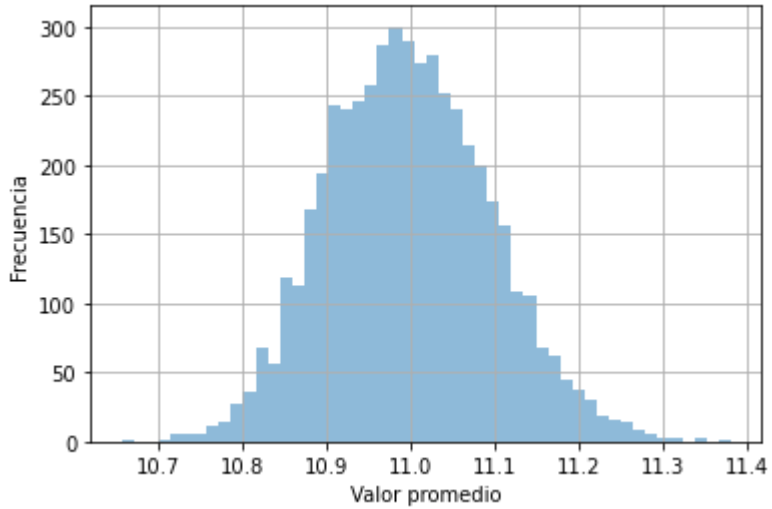
Ahora tomemos una muestra aleatoria de tamaño 100 y calculemos el promedio. Repitamos esto 5000 veces y almacenémoslo en un listado con todos los promedios muestrales. Por último, visualicemos un histograma de los promedios muestrales.

```
muestra_promedio_dis_expo = []

tamano = 5000

for i in range(0,tamano):
  muestra_promedio_dis_expo.append(poblacion_expo.sample(n=100).mean().values[0])

fig, ax = plt.subplots()
ax.hist(muestra_promedio_dis_expo, bins=50, alpha = 0.5)
ax.set_xlabel('Valor promedio')
ax.set_ylabel('Frecuencia')
ax.grid()
```



Dudas y consultas

KIBERNUM

Fin presentación

KIBERNUM