

Modelación y Simulación 2024

Lab 09

31.octubre.2024

En las clases anteriores hemos aprendido cómo simular y generar una muestra de una variable aleatoria X , a partir de una variable uniforme. En este lab, investigaremos como construir un generador de una variable uniforme $Y \sim Unif(0, 1)$.

1. (Pruebas para comparar dos distribuciones)

Vamos a usar pruebas de hipótesis para comparar dos distribuciones. Existen muchas pruebas para esto. En este lab, vamos a estudiar dos de ellas

- Explicar cómo funciona y cómo aplicar la prueba de Chi Cuadrado en Python, para comparar dos muestras.
- Explicar cómo funciona y cómo aplicar la prueba de Kolmogorov-Smirnov en Python, para comparar dos muestras.

Para ello es posible que necesite usar librerías especiales de estadística, como `scipy.stats` o `statsmodels`.

2. (Comparar muestras de una Geométrica)

Para una distribución geométrica $Geom(p)$, generar una muestra aleatoria de tamaño N usando la librería `scipy.stats`, a la cual llamaremos la muestra teórica. Luego, generar una muestra del mismo tamaño, pero usando el algoritmo de la transformada integral que aprendimos en la clase anterior. A esta le llamaremos la muestra empírica.

Comparar ambas muestras usando las prueba de Chi Cuadrado y la prueba de Kolmogorov-Smirnov. ¿Qué concluye? ¿Se pueden considerar como muestras de una misma distribución? Explique sus conclusiones a partir de las pruebas de hipótesis. Use un nivel de confianza de $\alpha = 0.05$.

3. (Comparar muestras de una Normal)

Para una distribución geométrica $N(\mu, \sigma^2)$, generar una muestra aleatoria de tamaño N usando la librería `scipy.stats`, a la cual llamaremos la muestra teórica. Luego, generar una muestra del mismo tamaño, pero usando el algoritmo de la transformada integral que aprendimos en la clase anterior. A esta le llamaremos la muestra empírica.

Comparar ambas muestras usando las prueba de Chi Cuadrado y la prueba de Kolmogorov-Smirnov. ¿Qué concluye? ¿Se pueden considerar como muestras de una misma distribución? Explique sus conclusiones a partir de las pruebas de hipótesis. Use un nivel de confianza de $\alpha = 0.05$.

4. (Generador Pseudoaleatorio Uniforme)

Investigar qué es un **generador pseudoaleatorio de números** (*pseudo-random number generator*). En particular investigar el generador lineal congruencial (GLC).

Implementar un generador aleatorio GLC para generar una distribución uniforme $Unif(0, 1)$. Generar una muestra de tamaño N , y aplicarle las pruebas de hipótesis para comparar los resultados generados contra una distribución uniforme teórica. Escriba sus conclusiones.

Defina usted los parámetros: el módulo m , las constantes $0 \leq a, c < m$, y el tamaño N de la muestra generada, y repita sus experimentos para 3 conjuntos diferentes de parámetros.

5. **(Otro Generador Pseudoaleatorio)**

Investigar e implementar en Python un generador aleatorio *Mersenne Twister* para generar una distribución uniforme $Unif(0, 1)$. Generar una muestra de tamaño N , y aplicarle las pruebas de hipótesis para comparar los resultados generados contra una distribución uniforme teórica. Escriba sus conclusiones.

Defina usted los parámetros y el tamaño N de la muestra generada, y repita sus experimentos para 3 conjuntos diferentes de parámetros. ¿Es mejor este generador que el GLC?
