GENERADORES DE NÚMEROS PSEUDOALETORIOS DE DISTINTAS DISTRIBUCIONES DE PROBABILIDAD - SIMULACION

Fadua Dora Sicardi **Franco Giangiordano** **Gonzalo Turconi** **Ignacio Curti**

20 de mayo de 2022

ABSTRACT

Los generadores de números pseudoaleatorios uniformes en informática son los más populares (e importantes) a la hora de trabajar con variables aleatorias, pero ésta no es la única distribución que estos números pueden tener. El trabajo de investigación actual incluye la investigación sobre generadores de números pseudoaleatorios. Estudiamos un conjunto de métodos analíticos para reproducir secuencias de números con cierta aleatoriedad, por lo que se denominan pseudoaleatorios. Los tipos de distribuciones que desarrollaremos son: Binomial, Empírica, Exponencial, Gamma, Hipergeométrica, Normal, Pascal, Poisson y Uniforme. Aclaramos que sean cuales sean las distribuciones de probabilidad que necesitemos que tengan los números pseudoaleatorios, se pueden generar utilizando programas basados en estadísticas computacionales básicas que se traducen en algoritmos simples para cada una de ellas.

***K*eywords**
Simulación, Números Pseudoaleatorios, Generadores, Distribuciones.

1. Introducción

La simulación depende fuertemente en la aleatoriedad de las variables que conforman el modelo. Esta aleatoriedad está dada por los generadores de números aleatorios que producen una sucesión de valores distribuidos uniformemente entre 0 y 1. Sin embargo, los modelos a simular no siempre poseen variables aleatorias distribuidas de esta manera, sino que se transforman para poder simular las distintas distribuciones que se requieran.

2. Marco teórico

Un número pseudoaleatorio es un número generado en un proceso determinístico, pero que se comporta como si hubiese sido generado al azar, aunque realmente no sea así. Una buena secuencia de números pseudoaleatorios no presenta ningún patrón o regularidad aparente desde el análisis estadístico, y debe ser capaz de superar Tests de rendimiento según sea la distribución que se busca que tengan éstas secuencias. Cabe aclarar que no importa cual sea la distribución de probabilidad que necesitamos que tengan los números pseudoaleatorios, es posible generarlos utilizando procedimientos basados puramente en estadística computacional básica que se traducen a un sencillo algoritmo para cada una de ellas. En teoría de la probabilidad y estadística, la distribución de probabilidad de una variable aleatoria es una función que asigna a cada suceso definido sobre la variable la probabilidad de que dicho suceso ocurra. La distribución de probabilidad está definida sobre el conjunto de todos los sucesos y cada uno de los sucesos es el rango de valores de la variable aleatoria.
Una distribución de probabilidades puede comprenderse como una distribución de frecuencia teórica, ya que describe cómo se espera que varíen los resultados.
La distribución de probabilidad está completamente especificada por la función de distribución, cuyo valor en cada x real es la probabilidad de que la variable aleatoria sea menor o igual que x.

2.1. Variables aleatorias

Una **variable aleatoria** es una función que asigna a cada elemento del espacio muestral un número real. Los valores posibles de una variable aleatoria pueden representar los posibles resultados de un experimento aún no realizado, o los posibles valores de una cantidad cuyo valor actualmente existente es incierto.

Como nuestro estudio se centra en valores numéricos los tipos de **variables** que nos interesan son:

- **Variable aleatoria discreta:** Es aquella que solo toma ciertos valores (frecuentemente enteros) y que resulta principalmente del conteo realizado.

- **Variable aleatoria continua:** Es aquella que resulta generalmente de la medición y puede tomar cualquier valor dentro de un intervalo dado

Algunas variables aleatorias con distribución de probabilidad de tipo discreta:

- Binomial
- Pascal
- Poisson
- Hipergeométrica
- Empírica

Algunas variables aleatorias con distribución de probabilidad de tipo continua:

- Normal
- Uniforme
- Exponencial
- Gamma

Para una correcta ejecución de la investigación recurrimos a definiciones cruciales como lo son:

El **promedio** μ es una medida de posición. No coincide necesariamente con un valor de la variable. El cálculo del promedio de n observaciones de la variable X (x_i con i = 1, 2,. . . ,n), resulta:

$$\mu = \frac{\sum_{i=1}^{n} x_i}{n} \tag{1}$$

La **varianza** σ^2 se define como el promedio de los cuadrados de las diferencias de los datos con promedio:

$$\sigma^2 = \frac{\sum\limits_{i=1}^{N} (x_i - \mu)^2}{N} \tag{2}$$

La **desviación estándar** σ es la raíz cuadrada positiva de la varianza:

$$\sigma = \sqrt{\sigma^2} \tag{3}$$

2.2. Funcion de Probabilidad Acumulada

En teoría de la probabilidad y en estadística, la función de distribución acumulada asociada a una variable aleatoria real X sujeta a cierta ley de distribución de probabilidad, es una función matemática de la variable real x quedescribe la probabilidad de que X tenga un valor menor o igual que x. Es decir: $F(x) = P\{X \leq x\}$

Para una variable aleatoria continua: $F(x) = \int_{-\infty}^{x} f(t)\, dt$ es la función de densidad de probabilidad de la distribución (la cual sera detallada en la subseccion siguiente).

Para una variable aleatoria discreta: $F(x) = \sum_{x_i \leq x} f(x_i)$, donde $f(x_i)$ es la función de probabilidad de la distribución.

2.3. Funcion de densidad de Probabilidad

En teoría de la probabilidad, la función de densidad de probabilidad continua describe la probabilidad relativa según la cual una variable aleatoria tomará determinado valor. La probabilidad de que la variable aleatoria caiga en una región específica del espacio de posibilidades estará dada por la integral de la densidad de esta variable entre uno y otro límite de dicha región. Esta función es positiva a lo largo de todo su dominio, y su integral sobre todo el espacio es de valor unitario.
Según la definición que dimos en función de densidad acumulada, se puede despejar la funcion de densidad de probabilidad de la siguiente expresión: $f(x) = \frac{d}{dx}F(x)$, y decimos que una variable aleatoria X tiene densidad f, siendo f una función no-negativa, continua e integrable, si: $P\{a \leq X \leq b\} = \int_a^b f(x)\,dx$

2.4. El metodo de la transformada inversa

Si deseamos generar los valores xi de las variables aleatorias a partir de cierta estadística de población cuya función de densidad este dada por f(x) debemos en primer lugar obtener la función de distribución acumulativa F(x). Puesto que F(x) se define sobre el rango de 0 a 1, podemos generar números aleatorios distribuidos uniformemente y además hacer F(x) = r. Resulta claro cómo queda x determinada unívocamente por r = F(x). Luego, para cualquier valor particular de r que generemos aleatoriamente, por ejemplo r_0 siempre es posible encontrar el valor de x que corresponde a r_0 debido a la función inversa de F si ésta es conocida. Esto es:

$$x_0 = F^{-1}{}_{r0} \tag{4}$$

donde $F^{-1}(x)$ es la transformación inversa de r sobre el intervalo unitario en el dominio de x. Si generamos números aleatorios uniformes correspondientes a una F(x) dada, podemos resumir matemáticamente este método como sigue:

$$r = F(x) = \int_{-\infty}^{x} f(t)\,dt \tag{5}$$

entonces:

$$P(X \leq x) = F(x) = P[r \leq F(x)] = P[F^{-1}{}_r \leq x] \tag{6}$$

y consecuentemente F_r^{-1} es una variable que tiene a f(x) como función de densidad de probabilidad.

3. Distribuciones de probabilidad

3.1. Distribuciones discretas

Se denomina distribución de variable discreta a aquella cuya función de probabilidad solo toma valores positivos en un conjunto de valores de X finito o infinito numerable. A dicha función se le llama función de masa de probabilidad. En este caso la distribución de probabilidad es la suma de la función de masa, por lo que tenemos entonces que:

$$F(x) = P(X \leq x) = \sum_{k=-\infty}^{x} f(k) \tag{7}$$

Y, tal como corresponde a la definición de distribución de probabilidad, esta expresión representa la suma de todas las probabilidades desde -∞ hasta el valor x.

Las distribuciones discretas que analizaremos son:

3.1.1. Distribución Binomial

La distribución binomial es una distribución de probabilidad discreta que cuenta el número de éxitos en una secuencia de n ensayos de Bernoulli independientes entre sí con una probabilidad fija p de ocurrencia de éxito entre los ensayos. Un experimento de Bernoulli se caracteriza por ser dicotómico, es decir, solo dos resultados son posibles: a uno de estos se le denomina "éxito" y tiene una probabilidad de ocurrencia p y al otro se le denomina "fracaso" y tiene una probabilidad $q = 1 - p$.

Su función de probabilidad está dada por:

$$P[X = x] = \binom{n}{x} p^x (1-p)^{n-x} \tag{8}$$

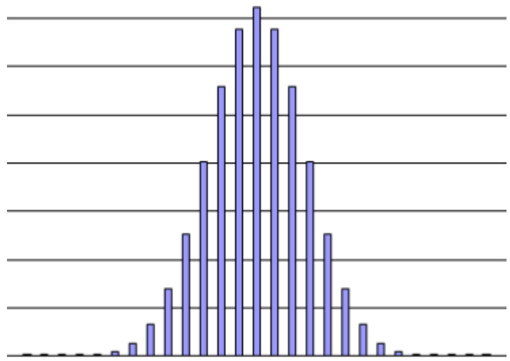

Figura 1: *Gráfica Teórica de la distribución Binomial*

3.1.2. Distribución Hipergeométrica

La distribución hipergeométrica es una distribución de probabilidad discreta relacionada con muestreos aleatorios y sin reemplazo. Suponga que se tiene una población de N elementos de los cuales, K pertenecen a la categoría A y $N-K$ pertenecen a la categoría B. La distribución hipergeométrica mide la probabilidad de obtener x $(0 \leq x \leq K)$ elementos de la categoría A en una muestra sin reemplazo de n elementos de la población original.
Su función de probabilidad está dada por:

$$P[X=x] = \frac{\binom{K}{x}\binom{N-K}{n-x}}{\binom{N}{n}} \tag{9}$$

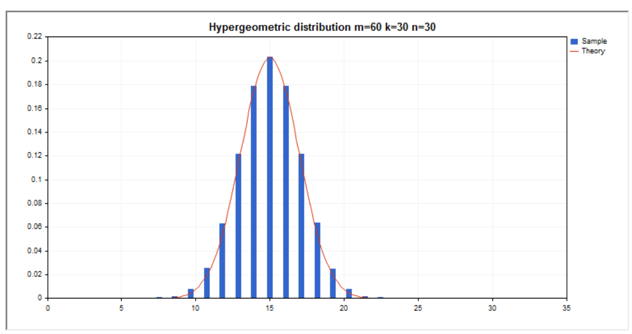

Figura 2: *Gráfica Teórica de la distribución Hipergeometrica*

3.1.3. Distribución de Pascal

La distribución de Pascal está incluida en la Distribución Binomial Negativa que es una distribución de probabilidad discreta. Es una ampliación de las distribuciones geométricas, utilizada en procesos en los cuales se ve necesaria la repetición de ensayos hasta conseguir un número de casos favorables. Una variable aleatoria geométrica corresponde al número de ensayos Bernoulli necesarios para obtener el primer éxito. Si deseamos conocer el número de estos para conseguir n éxitos, la variable aleatoria es binomial negativa. Su función de probabilidad está dada por:

$$f_x(x) = \binom{r+x-1}{x} p^r (1-p)^x \tag{10}$$

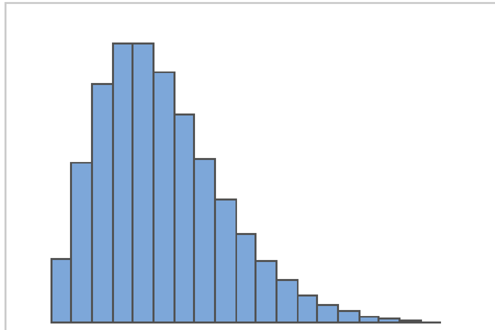

Figura 3: *Gráfica Teórica de la distribución Pascal*

3.1.4. Distribución Empírica

Es aquella asociada con las medidas empíricas de una muestra, es decir, las observaciones. Esta función de distribución acumulativa es una función escalonada que salta $\frac{1}{n}$ en cada uno de los n puntos de datos. Su valor en cualquier valor especificado de la variable medida es la fracción de observaciones de la variable medida que son menores o iguales al valor especificado.
Sean X_1, X_2, ..., X_n independientes, idénticamente distribuidas variables aleatorias reales con el común de la función de distribución acumulada F(t) . Entonces, la función de distribución empírica se define como:

$$F(t) = \frac{\text{número de elementos en la muestra} \leq t}{n} = \frac{1}{n}\sum_{i=1}^{n} 1x_i \leq t \tag{11}$$

3.2. Distribuciones continuas

Se denomina variable continua a aquella que puede tomar cualquiera de los infinitos valores existentes dentro de un intervalo. En el caso de variable continua la distribución de probabilidad es la integral de la función de densidad, por lo que tenemos entonces que:

$$F(x) = P(X \leq x) = \int_{-\infty}^{x} f(t)dt \tag{12}$$

3.2.1. Distribución Uniforme

En teoría de probabilidad y estadística, la distribución uniforme continua es una familia de distribuciones de probabilidad para variables aleatorias continuas, tales que para cada miembro de la familia, todos los intervalos de igual longitud en la distribución en su rango son igualmente probables. El dominio está definido por dos parámetros, a y b, que son sus valores mínimo y máximo. Decimos que una variable aleatoria X se distribuye uniformemente en el intervalo [a,b] y notamos $X \sim \mathcal{U}(a, b)$ cuando su densidad de probabilidad es:

$$fx(x) = \begin{cases} \dfrac{1}{b-a} & si \quad x \in [a, b], \\ 0 \\ caso & en \quad otro \end{cases} \tag{13}$$

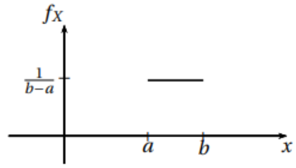

Figura 4: *Gráfica Teórica de la distribución Uniforme*

3.2.2. Distribución de probabilidad Exponencial

En estadística la distribución exponencial es una distribución de probabilidad continua con un parámetro > 0 Su función de densidad es:

$$fx(x) = \begin{cases} \lambda e^{\lambda x} & si \quad x > 0, \\ 0 & en \quad otro \\ caso & \end{cases} \tag{14}$$

La distribución exponencial es un caso particular de distribución gamma con k = 1. Además la suma de variables aleatorias que siguen una misma distribución exponencial es una variable aleatoria expresable en términos de la distribución gamma.

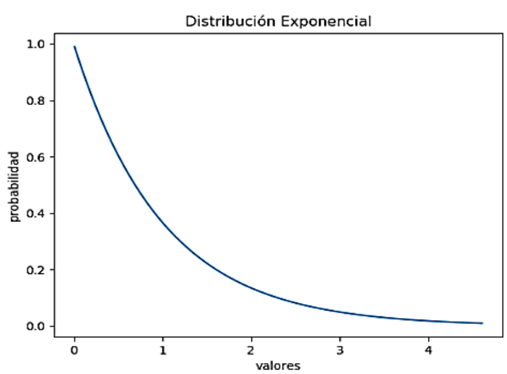

Figura 4: *Gráfica Teórica de la distribución Exponencial*

3.2.3. Distribución de probabilidad Gamma

En estadística la distribución gamma es una distribución de probabilidad continua con dos parámetros k>0 y > 0. La función de densidad para valores x>0 es:

$$f(x) = \lambda e^{-\lambda x} \frac{(\lambda x)^{k-1}}{\Gamma} \tag{15}$$

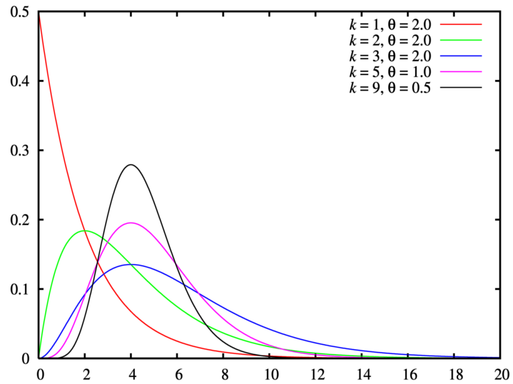

Figura 5: *Gráfica Teórica de la distribución Gamma*

3.2.4. Distribución de probabilidad Normal

En estadística y probabilidad se llama distribución normal, distribución de Gauss, distribución gaussiana o distribución de Laplace-Gauss, a una de las distribuciones de probabilidad de variable continua que con más frecuencia aparece en estadística y en la teoría de probabilidades.

La gráfica de su función de densidad tiene una forma acampanada y es simétrica respecto de un determinado parámetro estadístico. Esta curva se conoce como campana de Gauss y es el gráfico de una función gaussiana.

La importancia de esta distribución radica en que permite modelar numerosos fenómenos naturales, sociales y psi- cológicos.Mientras que los mecanismos que subyacen a gran parte de este tipo de fenómenos son desconocidos, por la enorme cantidad de variables incontrolables que en ellos intervienen, el uso del modelo normal puede justificarse asumiendo que cada observación se obtiene como la suma de unas pocas causas independientes.

De hecho, la estadística descriptiva sólo permite describir un fenómeno, sin explicación alguna. Para la explicación causal es preciso el diseño experimental, de ahí que al uso de la estadística en psicología y sociología sea conocido como método correlacional.

La distribución normal también es importante por su relación con la estimación por mínimos cuadrados, uno de los métodos de estimación más simples y antiguos

En probabilidad, la distribución normal aparece como el límite de varias distribuciones de probabilidad continuas y discretas.

La función de distribución de la distribución normal está definida como sigue:

$\phi_{\mu,\sigma^2} = \int_{-\infty}^{x} \varphi_{\mu,\sigma^2}(u)du = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{x} e^{\frac{(u-\mu)^2}{2\sigma^2}} du, x \in R$

donde:

- μ es la media
- σ es la desviación típica
- σ^2 es la varianza
- φ representa la función de densidad de probabilidad

También podemos definir la normal a través de la función de densidad:

$\phi_{\mu,\sigma^2} = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{(u-\mu)^2}{2\sigma^2}}, x \in R$

La función de distribución normal estándar es un caso especial de la función donde μ =0 y σ =1:

$\Phi(x) = \Phi_{0,1}(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{\frac{u^2}{2}} du, x \in R$

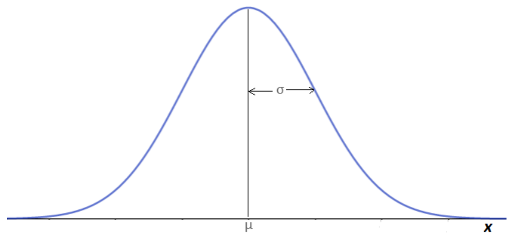

Figura 6: *Gráfica Teórica de la distribución Normal*

4. Prueba de Kolmogórov-Smirnov

La prueba de Kolmogórov-Smirnov es una propia perteneciente a la estadística, concretamente a la estadística inferencial. La estadística inferencial pretende extraer información sobre las poblaciones. Se trata de una prueba no paramétrica que determina la bondad de ajuste de dos distribuciones de probabilidad entre sí. Su objetivo es señalar si los datos provienen de una población que tiene la distribución teórica especificada, es decir, lo que hace es contrastar si las observaciones podrían razonablemente proceder de la distribución especificada. Principalmente, se usa cuando en una investigación tenemos dos muestras procedentes de dos poblaciones que son diferentes.

4.1. Supuestos

Para poder aplicar la prueba de Kolmogórov-Smirnov correctamente, se deben asumir una serie de supuestos. Primeramente, la prueba asume que los parámetros de la distribución de prueba se han especificado previamente. Este procedimiento estima los parámetros a partir de la muestra.

4.2. Ventajas

Algunas de las ventajas de la prueba de Kolmogórov-Smirnov son:

- Es más poderosa que la prueba Chi cuadrado (x2) (también prueba de bondad de ajuste).
- Es fácil de calcular y usar, y no requiere agrupación de los datos.
- El estadístico es independiente de la distribución de frecuencias esperada, solo depende del tamaño de la muestra

4.3. La hipótesis

H0: La distribución de frecuencia observada es consistente con la distribución de la frecuencia teórica (Buen ajuste).
H1: La distribución de frecuencia observada no es coherente con la distribución de la frecuencia teórica (Mal ajuste).
α = Nivel de significación de la prueba.

Para la siguiente prueba se utilizó un nivel de confianza del 95 %, lo cual, indica un riesgo de 5 % de concluir que los datos no se corresponden con la distribución buscada a pesar de que los datos sí siguen dicha distribución.

5. Exposición del código realizado en Python y explicación del mismo

En esta sección mostraremos el código realizado en Python para la obtención de las distintas distribuciones a partir de la generación de números pseudoaleatorios

5.1. Código de distribución Uniforme

```python
import numpy as np

def uniform (a,b):
    r = random()
    x= a+(b-a)*r
    return x
```

Listing 1: Algoritmo para la generación de numeros pseudoaleatorios con distribución Uniforme

En el código anterior ingresamos como parámetro dos valores, a y b, que expresan los limites de la distribución uniforme. El valor se generara a través del método de la transformación inversa con una variable r cuyo valor es obtenido por el generador pseudoleatorio de Python

5.2. Código de distribución Exponencial

```
import numpy as np

def exponential(ex):
    r = random()
    x = -ex * ln(r)
    return x
```

Listing 2: Algoritmo para la generación de numeros pseudoaleatorios con distribución Exponencial

En el código anterior ingresamos como parámetro el valor ex, que es la media de la distribución exponencial. El valor se generara a través del método de la transformación inversa con una variable r cuyo valor es obtenido por el generador pseudoleatorio de Python

5.3. Código de distribución Gamma

```
import numpy as np

def gamma(k, alpha):
    tr = 1
    for i in range(k):
        r = random()
        tr = tr * r
    x = -ln(tr) / alpha
    return x
```

Listing 3: Algoritmo para la generación de numeros pseudoaleatorios con distribución Gamma

En el código anterior ingresamos como parámetro k y alpha. El parámetro k representa la cantidad de eventos sucesivos y alpha representa el parámetro lambda a utilizar. El valor se generara a través del método de la transformación inversa con una variable r cuyo valor es obtenido por el generador pseudoleatorio de Python

5.4. Código de distribución Normal

```
import numpy as np

def normal(ex,stdx):
    sum = 0
    for i in range (12):
        r=random()
        sum += r
    x = stdx*(sum-6.0)+ex
    return x
```

Listing 4: Algoritmo para la generación de numeros pseudoaleatorios con distribución Normal

En el código anterior ingresamos como parámetro ex y stdx. El primero representa a la media y el segundo representa a la desviación estándar. El valor se generara a través del método de la transformación inversa con una variable r cuyo valor es obtenido por el generador pseudoleatorio de Python

5.5. Código de distribución Pascal

```
import numpy as np

def pascal(k,q):
    tr = 1
    qr = ln(q)
    for i in range(k):
        r = random()
        tr = tr * r
    x = floor(ln(tr)/qr)
    return x
```

Listing 5: Algoritmo para la generación de numeros pseudoaleatorios con distribución Pascal

En el código anterior ingresamos como parámetro k y q. El parámetro k es la cantidad de repeticiones hasta el éxito y el parámetro q es la probabilidad de no tener éxito. El valor se generara a través del método de la transformación inversa con una variable r cuyo valor es obtenido por el generador pseudoleatorio de Python

5.6. Código de distribución Binomial

```
import numpy as np

def binomial(n, p):
    x = 0
    for i in range(1, n):
        r = random()
        if (r - p) <= 0:
            x += 1
    return x
```

Listing 6: Algoritmo para la generación de numeros pseudoaleatorios con distribución Binomial

En el código anterior ingresamos como parámetro n y p. El parámetro n representa la cantidad de ensayos independientes de Bernoulli y p representa la probabilidad de éxito. El valor se generara a través del método de la transformación inversa con una variable r cuyo valor es obtenido por el generador pseudoleatorio de Python

5.7. Código de distribución Hipergeométrica

```
import numpy as np

def hypergeometric(tn, ns, p):
    x = 0
    for i in range(1, ns):
        r = random()
        if (r - p )<= 0:
            s = 1
            x += 1
        else:
            s = 0
        p = (tn * p - s) / (tn - 1)
        tn = tn - 1
    return x
```

Listing 7: Algoritmo para la generación de numeros pseudoaleatorios con distribución hipergeométrica

En el código anterior ingresamos como parámetro tn, ns y p. El parámetro tn representa la población total, ns representa la muestra seleccionada y p representa al índice de proporción. El valor se generara a través del método de la transformación inversa con una variable r cuyo valor es obtenido por el generador pseudoleatorio de Python

5.8. Código de distribución Poisson

```
import numpy as np

def poisson(p):
    x = 0
    b = exp(-p)
    tr = 1
    while (tr-b) >= 0:
        r = random()
        tr = tr * r
        x += 1
    return x
```

Listing 8: Algoritmo para la generación de numeros pseudoaleatorios con distribución Poisson

En el código anterior ingresamos como parámetro p que representa a lambda, el número promedio de eventos esperados por unidad de tiempo o de espacio. El valor se generara a través del método de la transformación inversa con una variable r cuyo valor es obtenido por el generador pseudoleatorio de Python

5.9. Código de distribución Empirica

```
import numpy as np

def empirica(sample):
    x=[]
    for i in range(100):
        r = random()
        a=0
        z=1
        for j in sample:
            a+=j
            if(r<=a):
                break
            else:
                z+=1
        x.append(z)
    return x
```

Listing 9: Algoritmo para la generación de numeros pseudoaleatorios con distribución Empirica

En el código anterior ingresamos como parámetro sample que representa las probabilidades de cada uno de los posibles sucesos = [0.273, 0.037, 0.195, 0.009, 0.124, 0.058, 0.062, 0.151, 0.047, 0.044]. El valor se generara a través del método de la transformación inversa con una variable r cuyo valor es obtenido por el generador pseudoleatorio de Python

6. Exposición de las gráficas de las distribuciones

6.1. Graficas de las Distribuciones Discretas

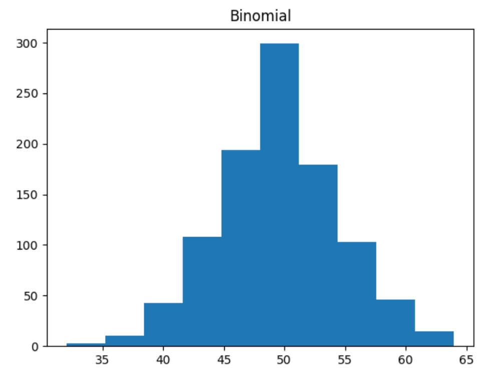

Figura 1: *Gráfica de barras de la distribución Binomial*

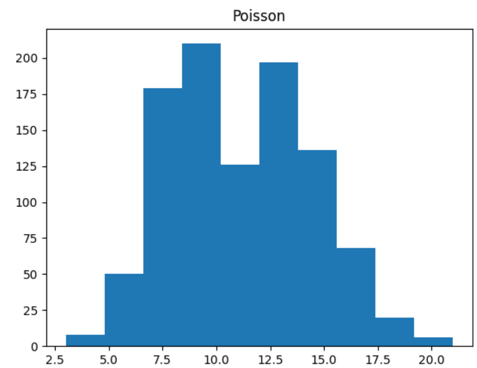

Figura 2: *Gráfica de barras de la distribución Poisson*

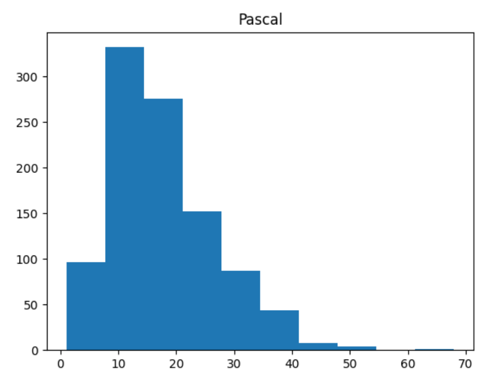

Figura 3: *Gráfica de barras de la distribución Pascal*

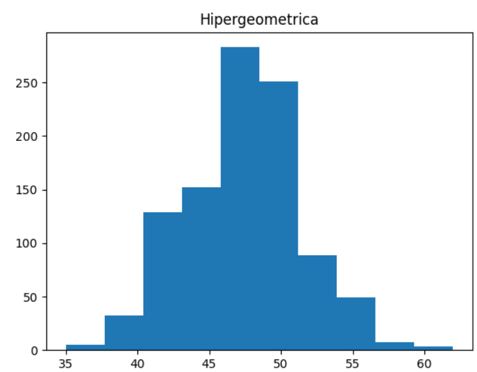

Figura 4: *Gráfica de barras de la distribución Hipergeométrica*

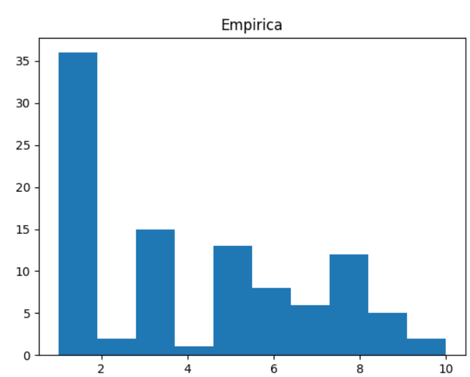

Figura 5: *Gráfica de barras de la distribución Empirica*

6.2. Graficas de las Distribuciones Continuas

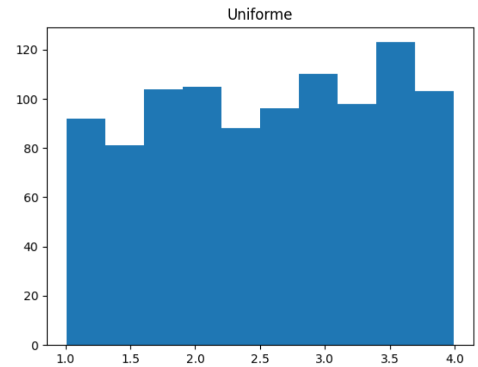

Figura 1: *Histograma de la distribución Uniforme*

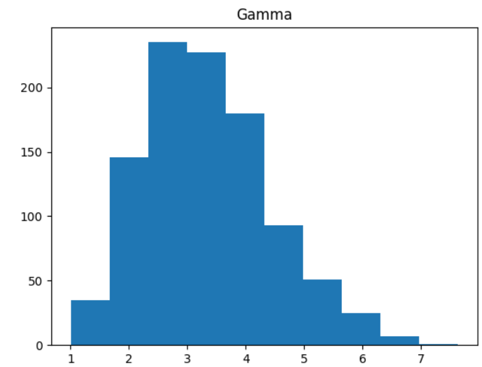

Figura 3: *Histograma de la distribución Gamma*

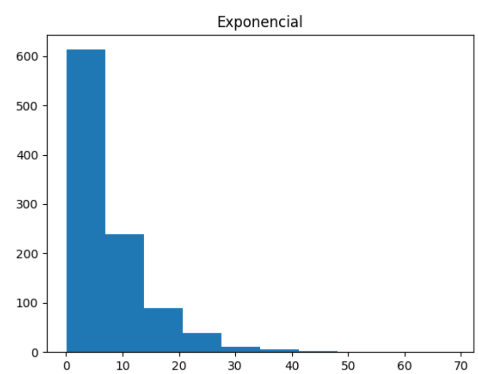

Figura 2: *Histograma de la distribución Exponencial*

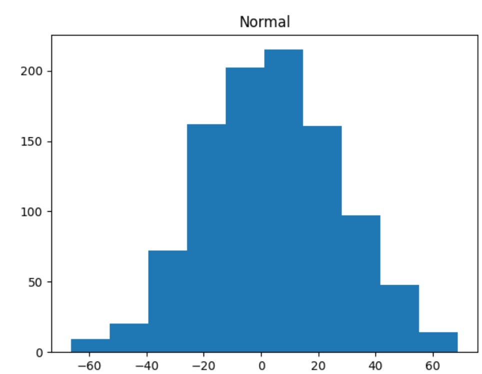

Figura 4: *Histograma de la distribución Normal*

7. Exposición de los resultados del Test de Kolmogorov Smirnov

Distribucion Binomial	
Cantidad de corridas	**Resultados**
Corrida 1	0.9942356257694902
Corrida 2	0.8933338427537486
Corrida 3	0.7089555196244846
Corrida 4	0.8755489553756126
Corrida 5	0.7029135792285938
Corrida 6	0.5967624253270457
Corrida 7	0.6539671261080773
Corrida 8	0.6539671261080773
Corrida 9	0.223090952631604
Corrida 10	0.3920251114650495

Cuadro 1: Tabla con los resultados del test de Kolmogorov Smirnov para 10 corridas de 100 numeros pseudoaleatorios con distribucion binomial

Distribucion Hipergeométrica	
Cantidad de corridas	**Resultados**
Corrida 1	2.2087606931995054e-59
Corrida 2	4.809713933059553e-82
Corrida 3	8.921184402908388e-97
Corrida 4	9.795786091743983e-108
Corrida 5	1.7995139563917767e-116
Corrida 6	9.753023778342561e-124
Corrida 7	5.8631465067451905e-130
Corrida 8	2.13128586279506e-135
Corrida 9	3.132316311431234e-140
Corrida 10	1.4055138003704013e-144

Cuadro 2: Tabla con los resultados del test de Kolmogorov Smirnov para 10 corridas de 100 numeros pseudoaleatorios con distribucion hipergeometrica

Distribucion Pascal	
Cantidad de corridas	**Resultados**
Corrida 1	1.4287357500157535e-51
Corrida 2	2.2670896492182174e-71
Corrida 3	1.5669425152930818e-87
Corrida 4	1.301638971092967e-87
Corrida 5	4.350350490275825e-98
Corrida 6	4.808353183911301e-102
Corrida 7	1.7478706911545398e-102
Corrida 8	9.739547718142809e-105
Corrida 9	5.443029307096649e-121
Corrida 10	3.151728633862023e-107

Cuadro 3: Tabla con los resultados del test de Kolmogorov Smirnov para 10 corridas de 100 numeros pseudoaleatorios con distribucion pascal

Distribucion Poisson	
Cantidad de corridas	**Resultados**
Corrida 1	0.9084105017744525
Corrida 2	0.17099477872745325
Corrida 3	0.0291063318902525
Corrida 4	0.045314127793102996
Corrida 5	0.12300392451700602
Corrida 6	0.12923301331713263
Corrida 7	0.04498403953147768
Corrida 8	0.10993303925958987
Corrida 9	0.5377883223791866
Corrida 10	0.01826550946571282

Cuadro 4: Tabla con los resultados del test de Kolmogorov Smirnov para 10 corridas de 100 numeros pseudoaleatorios con distribucion de poisson

Dsitribución Empírica Discreta	
Cantidad de corridas	**Resultados**
Corrida 1	2.7084430241713094e-18
Corrida 2	1.070075869779788e-12
Corrida 3	3.0406585087050305e-14
Corrida 4	1.7047967801006258e-09
Corrida 5	1.424536470142474e-19
Corrida 6	3.35710076793659e-13
Corrida 7	8.448372017533173e-11
Corrida 8	2.359168035084735e-10
Corrida 9	4.528308394643339e-17
Corrida 10	3.319419024623016e-12

Cuadro 5: Tabla con los resultados del test de Kolmogorov Smirnov para 10 corridas de 100 numeros pseudoaleatorios con distribucion empirica

Distribucion Uniforme	
Cantidad de corridas	**Resultados**
Corrida 1	0.015577131622877688
Corrida 2	0.3870644102581624
Corrida 3	0.5166740537529734
Corrida 4	0.2758282785102079
Corrida 5	0.1458258918637001
Corrida 6	0.9930655784580933
Corrida 7	0.36368041284747477
Corrida 8	0.26638152939186516
Corrida 9	0.10932320299120467
Corrida 10	0.5055028519339378

Cuadro 6: Tabla con los resultados del test de Kolmogorov Smirnov para 10 corridas de 100 numeros pseudoaleatorios con distribucion uniforme

Distribucion Exponencial	
Cantidad de corridas	**Resultados**
Corrida 1	0.05390207893129876
Corrida 2	0.5082964140925988
Corrida 3	0.04052802495213534
Corrida 4	0.7426633810170453
Corrida 5	0.8458839028922771
Corrida 6	0.035806086299017846
Corrida 7	0.5865203760593989
Corrida 8	0.5591479628658687
Corrida 9	0.6964281228010804
Corrida 10	0.41891337467915346

Cuadro 7: Tabla con los resultados del test de Kolmogorov Smirnov para 10 corridas de 100 numeros pseudoaleatorios con distribucion exponencial

Distribucion Normal	
Cantidad de corridas	**Resultados**
Corrida 1	0.8154147124661313
Corrida 2	0.7783973524742297
Corrida 3	0.3874916970869029
Corrida 4	0.7056414896982005
Corrida 5	0.07143654944636102
Corrida 6	0.77650763459057
Corrida 7	0.6765323484769824
Corrida 8	0.9180750405258453
Corrida 9	0.678697709140195
Corrida 10	0.40533687259832335

Cuadro 8: Tabla con los resultados del test de Kolmogorov Smirnov para 10 corridas de 100 numeros pseudoaleatorios con distribucion normal

Distribucion Gamma	
Cantidad de corridas	**Resultados**
Corrida 1	2.2087606931995054e-59
Corrida 2	4.809713933059553e-82
Corrida 3	8.921184402908388e-97
Corrida 4	9.795786091743983e-108
Corrida 5	1.7995139563917767e-116
Corrida 6	9.753023778342561e-124
Corrida 7	5.8631465067451905e-130
Corrida 8	2.13128586279506e-135
Corrida 9	3.132316311431234e-140
Corrida 10	1.4055138003704013e-144

Cuadro 9: Tabla con los resultados del test de Kolmogorov Smirnov para 10 corridas de 100 numeros pseudoaleatorios con distribucion gamma

Distribucion	Promedio	Resultado
Binomial	0.699985742574267	PASA
Hipergeometrica	2.2087606931995054e-60	NO PASA
Pascal	1.4287357500157536e-52	NO PASA
Poisson	0.21170335886553668	PASA
Empirica	2.0299529634746952e-10	NO PASA
Uniforme	0.3578923341630497	PASA
Exponencial	0.44880897245898743	PASA
Normal	0.6213531406503742	PASA
Gamma	2.2087606931995054e-60	NO PASA

Cuadro 10: Tabla con los promedios de los resultados del test de Kolmogorov Smirnov para 10 corridas de 100 numeros pseudoaleatorios en cada distribucion

Referencias a los resultados obtenidos en la tabla:

- Pasaron la prueba
- No pasaron la prueba

8. Conclusión

Como se puede observar, es posible generar distintas distribuciones aleatorias de probabilidad siguiendo algoritmos sencillos y rápidos de implementar. En su mayoría, los generadores que hemos presentado son contundentes y confiables. Comparando las gráficas de las distribuciones generadas con las distribuciones teóricas correspondientes podemos observar una gran similaridad en la mayoría de las distribuciones.
Sin embargo, hemos hallado ciertas distribuciones cuyo valor esperado p se encuentra muy lejos del necesario para pasar la prueba ($p = 0{,}05$). Esto se puede deber tanto como a un error a la hora de la generación de la distribuciones como a un error a la hora de la aplicación del test Kolmogorov-Smirnov. Notamos que de todas las distribuciones que no consideramos confiables (por no pasar la prueba), tres son discretas con una grafica es campanular.
En síntesis, dados los positivos resultados de las pruebas, consideramos que los aquellos algoritmos para generar distribuciones aleatorias que han superado la prueba que hemos presentado pueden llegar a ser utilizados en diferentes modelos de simulación en los que se necesiten diferentes distribuciones, aquellas que se adapten mejor a la realidad del problema en cuestión.

Referencias

[1] Python
https://python-para-impacientes.blogspot.com/2014/08/graficos-en-ipython.html
[2] Scipy
https://www.scipy.org/scipylib/index.html
[3] Probabilidad y Estadistica
https://relopezbriega.github.io/blog/2015/06/27/probabilidad-y-estadistica-con-python/
[4] Kolmogorov-Smirnov test
https://www.statology.org/kolmogorov-smirnov-test-python/
[5] Generación de distribuciones de probabilidad
https://numpy.org/doc/1.16/reference/routines.random.html
[6]Distribucion de probabilidad
https://es.wikipedia.org/wiki/Distribuci%C3%B3n$_{d}e_{p}robabilidad$
[7]Naylor, T.H. Técnicas de simulación en computadoras, 1982.