Función Generadora de Momentos y Varianza

Nexus-Probability

CURSO 1 (PROBABILIDAD I)

PARTE 5 / LECCIÓN 3

1. Introducción

En teoría de probabilidad y procesos estocásticos, las funciones generadoras son herramientas matemáticas que permiten analizar y caracterizar distribuciones de probabilidad. Entre las funciones generadoras más utilizadas se encuentran la función generadora de momentos y la función generadora de probabilidad.

2. Función Generadora de Momentos

La función generadora de momentos (FGM) de una variable aleatoria X es definida como:

$$M_X(t) = E[e^{tX}]$$

siempre que la esperanza exista en un entorno de $t = 0$. Esta función es útil porque permite obtener los momentos de la variable aleatoria de la siguiente manera:

$$E[X^n] = M_X^{(n)}(0)$$

es decir, derivando la función generadora de momentos n veces y evaluando en $t = 0$, se obtienen los momentos de la variable aleatoria.

3. Función Generadora de Probabilidad

La función generadora de probabilidad (FGP) de una variable aleatoria discreta X se define como:

$$G_X(s) = E[s^X] = \sum_{k=0}^{\infty} P(X = k)s^k$$

para valores de s dentro de un radio de convergencia apropiado. Esta función es útil en el estudio de procesos estocásticos, especialmente en cadenas de Markov y procesos de ramificación.

4. Principales Variables Aleatorias y sus Funciones Generadoras

Distribución Bernoulli

- Definición: Una variable aleatoria X sigue una distribución Bernoulli con parámetro p si toma valores 1 con probabilidad p y 0 con probabilidad $1-p$.
- FGM:
$$M_X(t) = (1-p) + pe^t$$
- FGP:
$$G_X(s) = (1-p) + ps$$

Distribución Binomial

- Definición: Una variable aleatoria X sigue una distribución Binomial $B(n,p)$ si representa el número de éxitos en n ensayos de Bernoulli independientes.
- FGM:
$$M_X(t) = (1-p+pe^t)^n$$
- FGP:
$$G_X(s) = (1-p+ps)^n$$

Distribución Poisson

- Definición: Una variable aleatoria X sigue una distribución Poisson con parámetro λ si representa el número de eventos en un intervalo de tiempo fijo con tasa promedio λ.
- FGM:
$$M_X(t) = e^{\lambda(e^t-1)}$$
- FGP:
$$G_X(s) = e^{\lambda(s-1)}$$

Distribución Normal

- Definición: Una variable aleatoria X sigue una distribución Normal con media μ y varianza σ^2, denotada como $N(\mu, \sigma^2)$.

- FGM:
$$M_X(t) = e^{\mu t + \frac{1}{2}\sigma^2 t^2}$$

Distribución Exponencial

- Definición: Una variable aleatoria X sigue una distribución Exponencial con parámetro λ si modela el tiempo entre eventos en un proceso de Poisson.
- FGM:
$$M_X(t) = \frac{\lambda}{\lambda - t}, \quad \text{para } t < \lambda$$

Ejercicio 1 *Una fábrica recibe pedidos de productos a una tasa media de 5 pedidos por hora, siguiendo una distribución de Poisson. Usa la función generadora de momentos para determinar:*

- *La media y la varianza de los pedidos por hora.*
- *La probabilidad de recibir exactamente 7 pedidos en una hora.*
- *Simula 1000 muestras de la cantidad de pedidos en Python y compara los valores empíricos con los teóricos.*

Este ejercicio analiza un proceso de Poisson, que modela la llegada de eventos en un intervalo de tiempo.

- La media y la varianza de una distribución Poisson con parámetro λ son ambas iguales a λ.
- Se usa la función generadora de momentos (FGM) para derivar estos valores.
- Se calcula la probabilidad de recibir exactamente 7 pedidos usando la función de masa de probabilidad (PMF).
- Finalmente, se simulan 1000 muestras para comparar los resultados teóricos con valores empíricos.

```
 import numpy as np
import scipy.stats as stats
import matplotlib.pyplot as plt
import seaborn as sns
from scipy.special import factorial

# Ejercicio 1: Distribuci n Poisson y la FGM
lambda_poisson = 5
mean_poisson = lambda_poisson
```

```
var_poisson = lambda_poisson

prob_7_pedidos = (lambda_poisson**7 * np.exp(-lambda_poisson)) /
    factorial(7)

samples_poisson = np.random.poisson(lambda_poisson, 1000)
empirical_mean = np.mean(samples_poisson)
empirical_var = np.var(samples_poisson)
```

Ejercicio 2 *Un casino ofrece un juego donde un jugador tiene una probabilidad del 40 % de ganar en cada intento. Si juega 10 veces:*

- *Usa la función generadora de probabilidad para determinar la media y la varianza de las ganancias.*
- *¿Cuál es la probabilidad de que gane exactamente 6 veces?*
- *Simula 5000 repeticiones del juego y compara con la distribución teórica.*

Este ejercicio trata sobre un juego de azar con ensayos de Bernoulli repetidos.

- Se usa la función generadora de probabilidad (FGP) para encontrar la media y varianza de una distribución binomial.
- Se calcula la probabilidad de ganar exactamente 6 veces utilizando la PMF de la binomial.
- Se simulan 5000 juegos para comparar los resultados empíricos con los teóricos.

```

# Ejercicio 2: Distribuci n Binomial y la FGP
n, p = 10, 0.4
mean_binomial, var_binomial = n * p, n * p * (1 - p)
prob_6_wins = stats.binom.pmf(6, n, p)
samples_binomial = np.random.binomial(n, p, 5000)
empirical_mean_binomial, empirical_var_binomial = np.mean(
   samples_binomial), np.var(samples_binomial)
```

Ejercicio 3 *El tiempo entre llegadas de clientes a un restaurante sigue una distribución exponencial con una media de 2 minutos.*

- *Usa la función generadora de momentos para encontrar la varianza.*
- *Determina la probabilidad de que el siguiente cliente tarde más de 5 minutos en llegar.*
- *Simula 10000 llegadas y verifica la distribución empírica.*

Este ejercicio modela tiempos de espera entre llegadas de clientes, siguiendo una distribución exponencial.

- Se usa la FGM para calcular la varianza de la distribución exponencial.
- Se determina la probabilidad de esperar más de 5 minutos usando la función de distribución acumulada (CDF).
- Se simulan 10000 muestras para comparar la distribución empírica con la teórica.

```
# Ejercicio 3: Distribuci n Exponencial y la FGM
lambda_exp = 1/2
var_exp = 1 / lambda_exp**2
prob_more_5 = 1 - stats.expon.cdf(5, scale=1/lambda_exp)
samples_exp = np.random.exponential(1/lambda_exp, 10000)
empirical_var_exp = np.var(samples_exp)
```

Ejercicio 4 *Supón que los tiempos de espera de dos cajeros siguen distribuciones exponenciales independientes con parámetros $\lambda_1 = 1$ y $\lambda_2 = 2$.*

- *Usa la función generadora de momentos para encontrar la distribución de la suma de los tiempos de espera.*
- *Simula 5000 sumas de estos tiempos de espera y compáralos con la distribución teórica.*

Aquí se estudia la suma de dos variables aleatorias exponenciales independientes.

- Se usa la FGM para analizar la distribución de la suma de tiempos de espera.
- Se simulan 5000 valores de la suma de dos tiempos de espera y se comparan con la distribución teórica.

```python
# Ejercicio 4: Suma de Variables Exponenciales
lambda_1, lambda_2 = 1, 2
samples_sum = np.random.exponential(1/lambda_1, 5000) + np.random.exponential(1/lambda_2, 5000)
```

Ejercicio 5 *Si una variable aleatoria X sigue una distribución normal con media 10 y varianza 4:*

- *Usa la función generadora de momentos para verificar que la media y la varianza son correctas.*
- *Genera 10000 valores aleatorios en Python y compara los momentos empíricos con los teóricos.*

Este ejercicio verifica propiedades de la distribución normal mediante su FGM.

- Se usa la FGM para verificar la media y varianza teóricas.
- Se generan 10000 valores aleatorios y se comparan los momentos empíricos con los teóricos.

```python
# Ejercicio 5: Distribuci n Normal y la FGM
mu_normal, sigma_normal = 10, 2
samples_normal = np.random.normal(mu_normal, sigma_normal, 10000)
empirical_mean_normal, empirical_var_normal = np.mean(samples_normal), np.var(samples_normal)
```