COVARIANZA Y CORRELACIÓN ENTRE VARIABLES ALEATORIAS.

Nexus-Probability

CURSO 2 (PROBABILIDAD II)

Lección 2

1.

Comencemos esta lección recordando el siguiente resultado.

Teorema 1 *Si X y Y son variables aleatorias independientes, entonces, para cualesquiera funciones h y g se cumple*

$$E[g(X)h(Y)] = E[g(X)]E[h(Y)] \tag{1}$$

Así como el valor presente y la varianza de una sola variable aleatoria nos da información sobre la variable aleatoria, también la covarianza de dos variables aleatrias nos da información de como se relacionan estas dos variables aleatorias.

Definición 1 (Covarianza de dos variables aleatorias.) *La covarianza entre X y Y, denotada como $Cov(X,Y)$, se define como*

$$Cov(X,Y) = E[(X - E[X])(Y - E[Y]] \tag{2}$$

Realizando algunas operaciones podemos llegar a esta otra equivalencia

$$Cov(X,Y) = E[XY] - E[X]E[Y] \tag{3}$$

Observar que si X y Y son independientes, por el teorema 1, $Cov(X,Y) = 0$. Sin embargo, si $Cov(X,Y) = 0$ no implica independencia entre las variables aleatorias.

La siguiente proposición enuncia las principales propiedades de la covarianza.

Teorema 2

- $Cov(X,Y) = Cov(Y,X)$
- $Cov(X,X) = Var(X,X)$
- $Cov(aX,Y) = aCov(X,Y)$
- $Cov\left(\sum_{i=1}^{n} X_i, \sum_{j=1}^{m} Y_j\right) = \sum_{i=1}^{n}\sum_{j=1}^{m} Cov(X_i, Y_j)$

Ejemplos

Ejercicio 1 *Supongamos que tienes un portafolio de 3 activos financieros. Los retornos de estos activos siguen distribuciones normales con medias* μ_1, μ_2, μ_3 *y desviaciones estándar* $\sigma_1, \sigma_2, \sigma_3$ *. La matriz de covarianza* $\sum$ *entre estos activos es conocida. Simula 1000 realizaciones de retornos para cada activo y calcula la covarianza y correlación entre el retorno total del portafolio y el retorno del activo 1.*
Datos:

- $\mu_1 = 0.02, \mu_2 = 0.012, \mu_3 = 0.01$
- $\sigma_1 = 0.05, \sigma_2 = 0.03, \sigma_3 = 0.04$
- *La matríz de covarianzas es:*

$$\sum = \begin{pmatrix} 0.0025 & 0.0012 & 0.0011 \\ 0.0012 & 0.0009 & 0.0008 \\ 0.0011 & 0.0008 & 0.0016 \end{pmatrix}$$

El retorno total del portafolio es la suma ponderada de los retornos de los tres activos, con pesos $w_1 = 0.05, w_2 = 0.3$ *y* $w_3 = 0.2$

1) Simulamos 1000 realizaciones de retornos para los tres activos usando sus distribuciones normales y la matriz de covarianza.
2) Calculamos el retorno total del portafolio usando los pesos dados.
3) Calculamos la covarianza y la correlación entre el retorno total del portafolio y el retorno del activo 1.

```
import numpy as np

# Par metros
mu = [0.02, 0.015, 0.01]
sigma = [0.05, 0.03, 0.04]
```

```python
cov_matrix = np.array([[0.0025, 0.0012, 0.0011],
                       [0.0012, 0.0009, 0.0008],
                       [0.0011, 0.0008, 0.0016]])

# Pesos del portafolio
weights = np.array([0.5, 0.3, 0.2])

# Generar 1000 simulaciones de retornos con distribuci n normal multivariada
returns = np.random.multivariate_normal(mu, cov_matrix, 1000)

# Retorno total del portafolio
portfolio_return = returns @ weights

# Retorno del activo 1
asset_1_return = returns[:, 0]

# Calcular covarianza y correlaci n
covarianza = np.cov(portfolio_return, asset_1_return)[0, 1]
correlacion = np.corrcoef(portfolio_return, asset_1_return)[0, 1]

print(f'Covarianza: {covarianza}')
print(f'Correlaci n: {correlacion}')

```

Ejercicio 2 *Sea $X \sim N(0,1)$ y define dos nuevas variables $Y = e^X$ y $Z = log(1+X^2)$. Calcula la covarianza entre Y y Z. ¿Qué interpretación se puede dar a la relación entre estas dos tranformaciones lineales?*

Solución:
1) Generamos 1000 realizaciones de $X \sim N(0,1)$.
2) Calculamos $Y = e^X y$Z=log(1+X^2).
3)Calculamos la covarianza de Y y Z.

```python
        # Generar realizaciones de X
X = np.random.normal(0, 1, 1000)

# Definir Y = e^X y Z = log(1 + X^2)
Y = np.exp(X)
Z = np.log(1 + X**2)

# Calcular la covarianza
covarianza = np.cov(Y, Z)[0, 1]
```

```python
print(f'Covarianza entre Y y Z: {covarianza}')


```

Ejercicio 3 *Sea $X \sim N(0,1), Z \sim N(0,1)$ y definamos $Y = X + Z$. Calcule la correlación entre X y Y, y luego entre Z e Y. ¿Qué puedes concluir sobre la influencia de una tercera variable en la correlación?*

1) Generamos realizaciones de X y Z, ambos independientes.

2) Definimos $Y = X + Z$.

3) Calculemos las correlaciones $Corr(X, Y)$ y $Corr(Z, Y)$

```python
# Generar X y Z independientes
X = np.random.normal(0, 1, 1000)
Z = np.random.normal(0, 1, 1000)

# Definir Y = X + Z
Y = X + Z

# Calcular correlaciones
corr_X_Y = np.corrcoef(X, Y)[0, 1]
corr_Z_Y = np.corrcoef(Z, Y)[0, 1]

print(f'Correlacion entre X e Y: {corr_X_Y}')
print(f'Correlacion entre Z e Y: {corr_Z_Y}')



```

Ejercicio 4 *Genera dos series temporales X_t e Y_t donde*

- $X_t = 0.07X_{t-1} + \epsilon_t, \;\; \epsilon{\sim}N(0,1),$
- $Y_t = 0.5Y_{t-1} + n_t, \;\; n_t \sim N(0,1),$ *y calcula la covarianza entre X_t e Y_t en cada paso del tiempo t.*

Solución:
1) Simulamos las series temporales X_t e Y_t con las relaciones de autocorrelaciones dadas tiempo t.

2) Calculamos la covarianza en cada paso del tiempo.

```python
# Par metros
n = 1000
X = np.zeros(n)
Y = np.zeros(n)
epsilon = np.random.normal(0, 1, n)
eta = np.random.normal(0, 1, n)

# Inicializaci n
X[0] = epsilon[0]
Y[0] = eta[0]

# Generar las series temporales
for t in range(1, n):
    X[t] = 0.7 * X[t-1] + epsilon[t]
    Y[t] = 0.5 * Y[t-1] + eta[t]

# Calcular covarianza entre X e Y en cada paso
covarianzas = [np.cov(X[:t], Y[:t])[0, 1] for t in range(2, n)]
print(f'Covarianza en el  ltimo  paso: {covarianzas[-1]}')


```

Ejercicio 5 *Sea $X \sim N(0,1)$ y define $Y = 3X + \epsilon$ donde $\epsilon \sim N(0,2)$ es ruido independiente. Calcule la correlación entre X y Y. Luego, repite el experimento con $\epsilon N(0,10)$ y compara los resultados. Explica cómo el ruido afecta la correlación.*

Solución:

- Se general X y $Y = 3X + \epsilon$.
- Se calcula la correlación entre X y Y con dos niveles de ruido diferentes.
- Se compara el impacto del ruido en la correlación.

```python
import numpy as np
from scipy.stats import pearsonr

# Generar X
X = np.random.normal(0, 1, 1000)

# Caso 1: Ruido con varianza 2
epsilon_1 = np.random.normal(0, np.sqrt(2), 1000)
Y1 = 3*X + epsilon_1
correlacion_1 = pearsonr(X, Y1)[0]

```

```python
# Caso 2: Ruido con varianza 10
epsilon_2 = np.random.normal(0, np.sqrt(10), 1000)
Y2 = 3*X + epsilon_2
correlacion_2 = pearsonr(X, Y2)[0]

print(f'Correlaci n con ruido de varianza 2: {correlacion_1:.3f}')
print(f'Correlaci n con ruido de varianza 10: {correlacion_2:.3f}')

```

- En el primer caso, como Y es mayormente una función de X, la correlación debería ser alta.
- En el segundo caso, al aumentar la varianza de ϵ, la relación entre X y Y se desvanece debido al ruido agregado, lo que reduce la correlación.
- El ruido disminuye la correlación, ya que introduce variabilidad en Y que no está explicada por X.