

**Centro de Enseñanza Técnica Industrial**

**Desarrollo de Software**

**Actividad 2 - Clase 8**

**Jesús Alberto Aréchiga Carrillo**

**22310439 6N**

**Profesor**

**Clara Margarita Fernández Riveron**

**Abril de 2025**

**Guadalajara, Jalisco**

## Introducción

En probabilidad discreta, el estudio de las distribuciones conjuntas de dos (o más) variables aleatorias permite describir cómo se comportan en conjunto. Para variables aleatorias discretas XXX e YYY, la función de probabilidad conjunta

se suele presentar en forma de tabla o matriz. A partir de ella podemos obtener las distribuciones marginales (sumando sobre una de las variables) y, de ahí, las distribuciones condicionales

## Ejercicio:

**Caso de estudio:** Analizar el coeficiente de correlación entre ingresos y gastos en datos económicos (usar datos simulados).

import numpy as np

import matplotlib.pyplot as plt

from scipy import stats

def generar\_datos(n, media\_ing=50000, std\_ing=10000, pendiente=0.6, intercepto=10000, ruido\_std=5000, semilla=42):

    """

    Genera datos simulados de ingresos y gastos.

    Parámetros:

    - n: número de observaciones

    - media\_ing, std\_ing: media y desviación estándar de la distribución de ingresos

    - pendiente, intercepto: parámetros de la relación lineal gastos = pendiente\*ingresos + intercepto

    - ruido\_std: desviación estándar del ruido aleatorio en gastos

    - semilla: semilla para reproducibilidad

    Retorna:

    - ingresos: array de forma (n,)

    - gastos:   array de forma (n,)

    """

    np.random.seed(semilla)

    ingresos = np.random.normal(loc=media\_ing, scale=std\_ing, size=n)

    # Generamos gastos como función lineal de los ingresos más ruido

    gastos = pendiente \* ingresos + intercepto + np.random.normal(scale=ruido\_std, size=n)

    return ingresos, gastos

def analizar\_correlacion(ingresos, gastos):

    """

    Calcula y devuelve el coeficiente de correlación de Pearson

    y su p-valor asociado.

    """

    corr\_coef, p\_valor = stats.pearsonr(ingresos, gastos)

    return corr\_coef, p\_valor

def graficar(ingresos, gastos):

    """

    Dibuja un scatter plot de ingresos vs. gastos

    y traza la línea de regresión lineal.

    """

    # Ajuste de regresión lineal

    pendiente, intercepto, r\_val, p\_val, std\_err = stats.linregress(ingresos, gastos)

    linea = pendiente \* ingresos + intercepto

    plt.figure()

    plt.scatter(ingresos, gastos, label='Datos simulados')

    plt.plot(ingresos, linea, label=f'Ajuste lineal\ny={pendiente:.2f}·x+{intercepto:.0f}')

    plt.xlabel('Ingresos')

    plt.ylabel('Gastos')

    plt.title('Dispersión Ingresos vs. Gastos con línea de regresión')

    plt.legend()

    plt.grid(True)

    plt.show()

def main():

    # 1. Generar datos

    ingresos, gastos = generar\_datos(n=100)

    # 2. Calcular coeficiente de correlación

    corr, p = analizar\_correlacion(ingresos, gastos)

    print(f"Coeficiente de correlación de Pearson: {corr:.3f}")

    print(f"P-valor asociado: {p:.3e}")

    # 3. Graficar resultados

    graficar(ingresos, gastos)

if \_\_name\_\_ == "\_\_main\_\_":

    main()

A graph with blue dots and white text

AI-generated content may be incorrect.

## Conclusiones:

El análisis de distribuciones conjuntas y condicionales es una herramienta fundamental en estadística y probabilidad, ya que nos permite cuantificar la dependencia entre variables y actualizar nuestras creencias ante nueva información (principio de Bayes). Estas técnicas se aplican en campos tan diversos como la ingeniería, la economía, el aprendizaje automático y la bioestadística. Dominar el paso de la distribución conjunta a las marginales y condicionales abre la puerta a modelos más sofisticados (por ejemplo, cadenas de Markov o redes bayesianas) y a la toma de decisiones basadas en evidencia probabilística.