

# MM3014 - Distribución de probabilidad

February 19, 2025

## 1 Distribución de probabilidad

Sea  $(S, \mathcal{S}, P)$  un espacio de probabilidad.

La función  $F : \mathbb{R} \rightarrow \mathbb{R}$  definida como  $F(x) = P(\{X \leq x\})$  se llama **función de distribución acumulada de la variable aleatoria  $X$**  (o solo **función de distribución de  $X$** ).

Para una variable aleatoria discreta  $X$ , la función  $p : \mathbb{R} \rightarrow \mathbb{R}$  definida como  $p(a) = P(\{X = a\})$  se llama **función de masa de probabilidad de  $X$** .

### 1.1 Ejemplo 4a.

Trace la función de masa de probabilidad del experimento del Ejemplo 1.

```
[1]: import itertools as it
import matplotlib.pyplot as plt
import numpy as np

S = set({'H', 'T'})

R = [(m, len([k for k in S if k.count('H') == m])) for m in range(0, 2)]
X = [k[0] for k in R] # Variable aleatoria
P = [round(k[1]/len(S), 3) for k in R] # Medida de probabilidad de cada evento

# Gráfico de barras
plt.bar(X, P, tick_label=[str(k) for k in X])
plt.xlabel("Resultado del lanzamiento")
plt.ylabel("Probabilidad")
plt.title("Función de Masa de Probabilidad (PMF)")
plt.ylim(0, max(P)+0.1)

# Mostrar valores en las barras
for i, prob in zip(X, P):
    plt.text(i, prob + 0.05, str(prob), ha='center', fontsize=12)

plt.show()

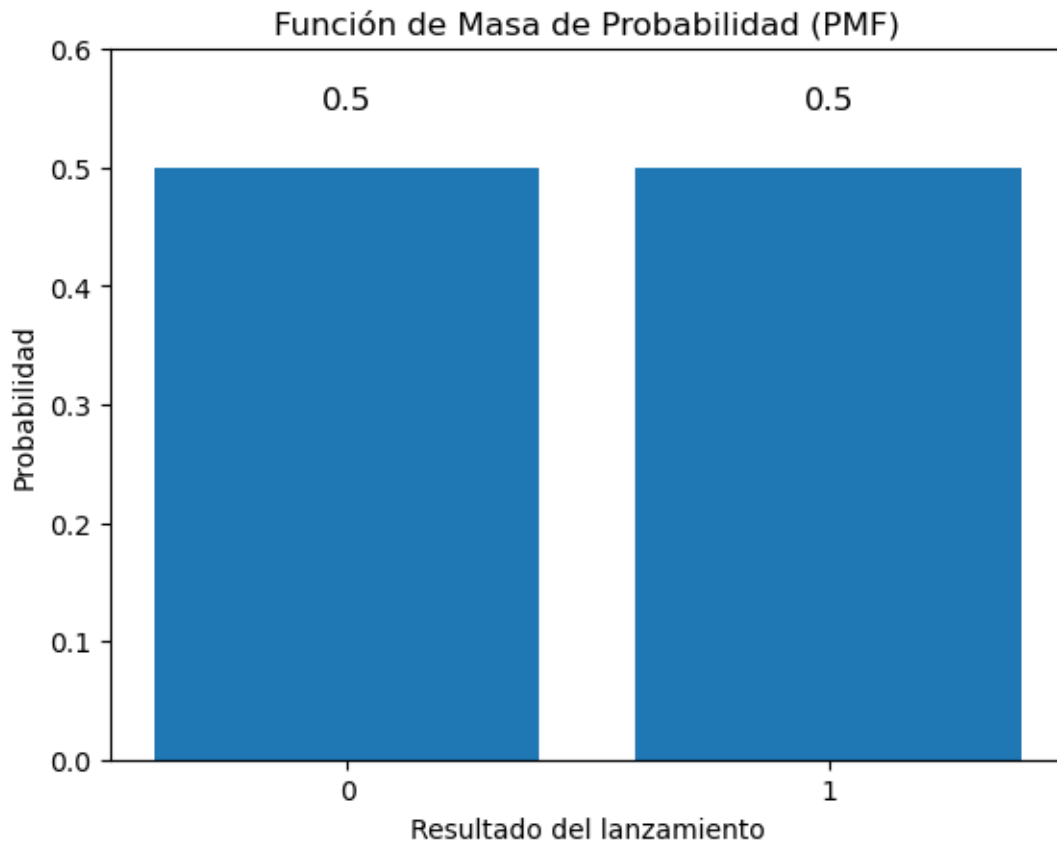
# Gráfico de escalón
acum = 0
```

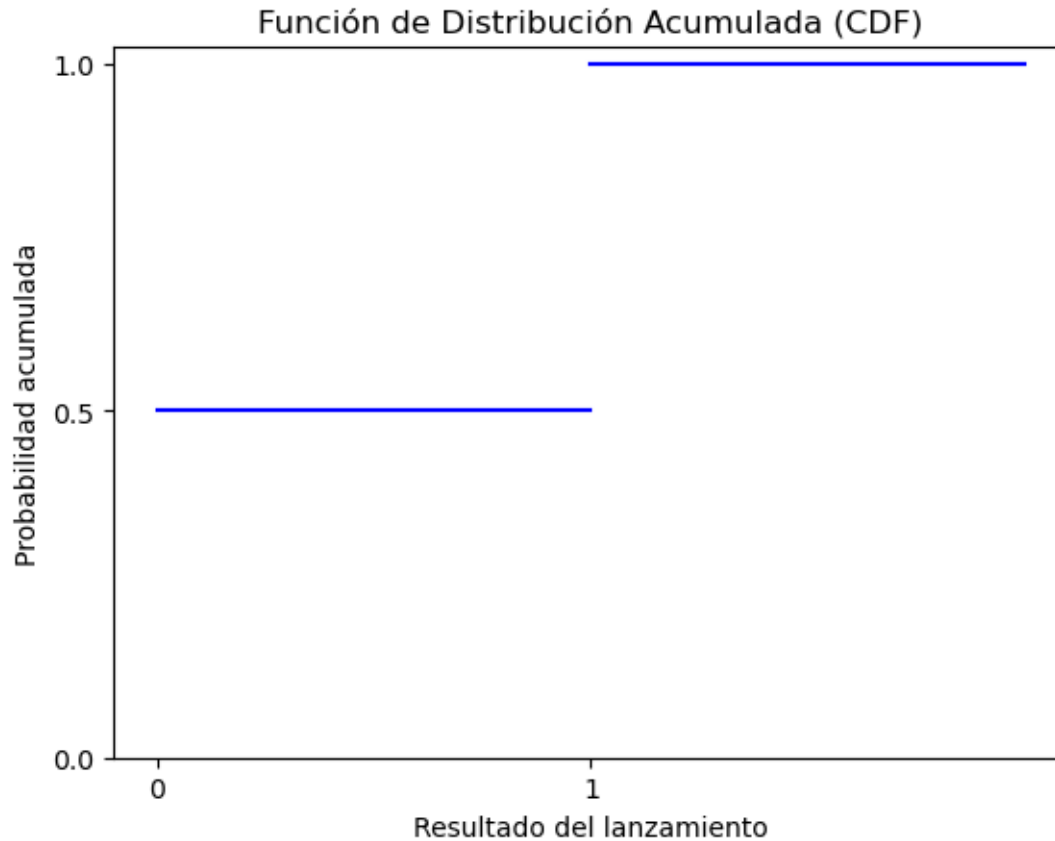
```

for i in range(len(X)-1):
    acum += P[i]
    plt.step([X[i],X[i+1]], [acum, acum], where='pre', color='blue')
plt.step([X[-1], X[-1]+1], [sum(P), sum(P)], where='pre', color='blue')

plt.xlabel("Resultado del lanzamiento")
plt.ylabel("Probabilidad acumulada")
plt.title("Función de Distribución Acumulada (CDF)")
plt.xticks(X, [str(k) for k in X])
plt.yticks(np.linspace(0,1,int(1/P[0])+1))
plt.show()

```





## 1.2 Ejemplo 4b.

Trace la función de masa de probabilidad del experimento del Ejemplo 2.

```
[2]: A = set({'H', 'T'})
S = set(it.product(A, repeat=2))

R = [(m, len([k for k in S if k.count('T') == m])) for m in range(0,3)]
X = [k[0] for k in R] # Variable aleatoria
P = [round(k[1]/len(S),3) for k in R] # Medida de probabilidad de cada evento

# Gráfico de barras
plt.bar(X, P, tick_label=[str(k) for k in X])
plt.xlabel("Resultado del lanzamiento")
plt.ylabel("Probabilidad")
plt.title("Función de Masa de Probabilidad (PMF)")
plt.ylim(0, max(P)+0.1)

# Mostrar valores en las barras
for i, prob in zip(X, P):
```

```

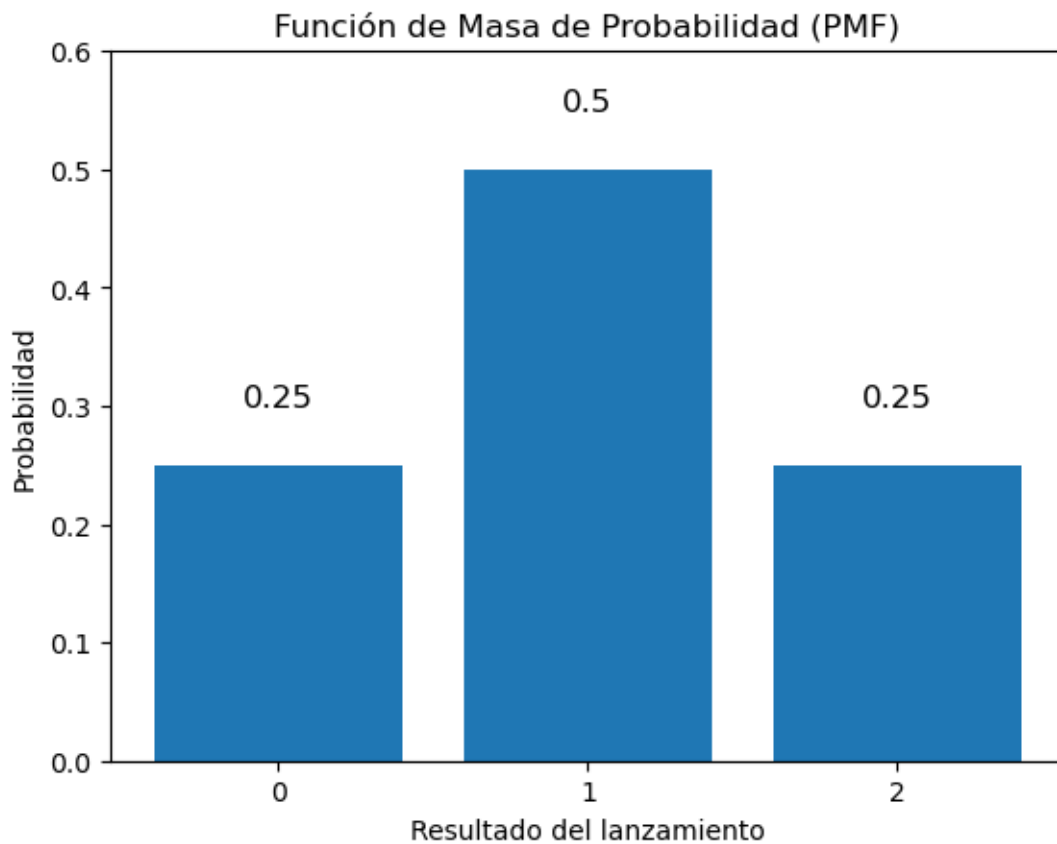
plt.text(i, prob + 0.05, str(prob), ha='center', fontsize=12)

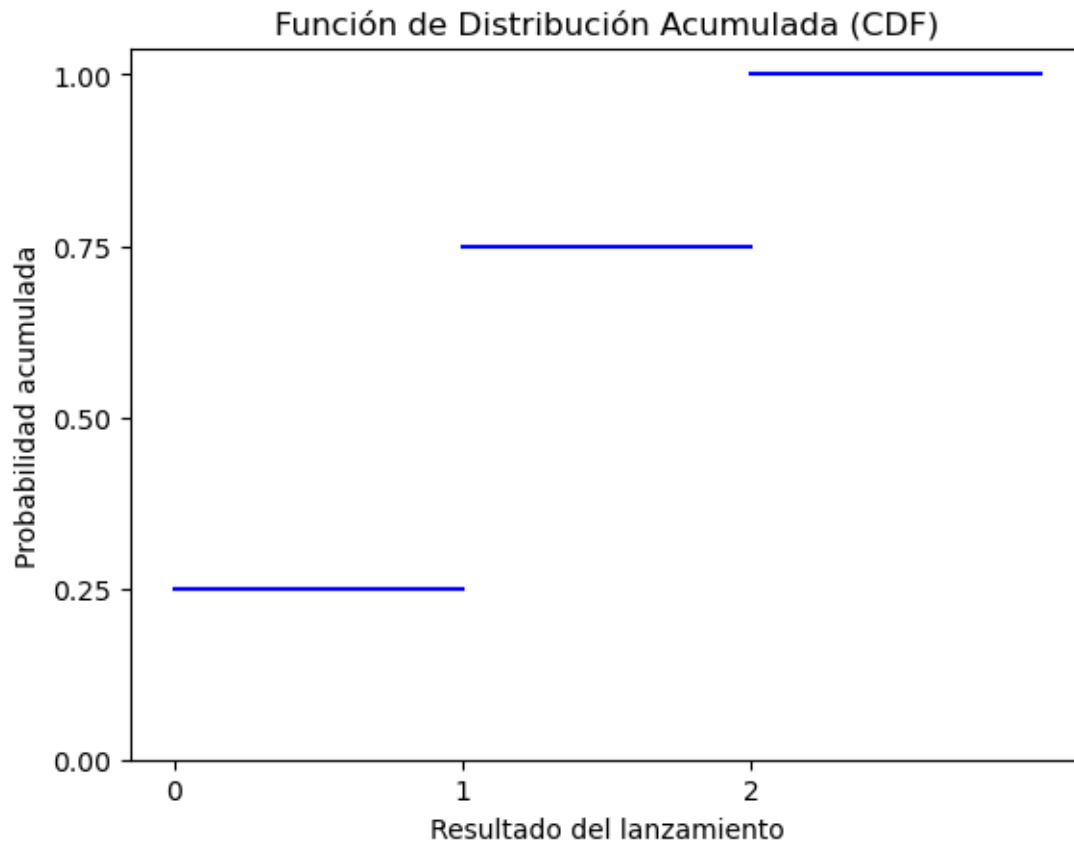
plt.show()

# Gráfico de escalón
acum = 0
for i in range(len(X)-1):
    acum += P[i]
    plt.step([X[i],X[i+1]], [acum, acum], where='pre', color='blue')
plt.step([X[-1], X[-1]+1], [sum(P), sum(P)], where='pre', color='blue')

plt.xlabel("Resultado del lanzamiento")
plt.ylabel("Probabilidad acumulada")
plt.title("Función de Distribución Acumulada (CDF)")
plt.xticks(X, [str(k) for k in X])
plt.yticks(np.linspace(0,1,int(1/P[0])+1))
plt.show()

```





### 1.3 Ejemplo 4c.

Trace la función de masa de probabilidad del experimento del Ejemplo 3.

```
[3]: A = set({1,2,3,4,5,6})
S = set(it.product(A,repeat=2))

R = [(m,len([k for k in S if sum(k) == m])) for m in range(3,13)]
X = [k[0] for k in R] # Variable aleatoria
P = [round(k[1]/len(S),3) for k in R] # Medida de probabilidad de cada evento

# Gráfico de barras
plt.bar(X, P, tick_label=[str(k) for k in X])
plt.xlabel("Resultado del lanzamiento")
plt.ylabel("Probabilidad")
plt.title("Función de Masa de Probabilidad (PMF)")
plt.ylim(0, max(P)+0.1)

# Mostrar valores en las barras
for i, prob in zip(X, P):
```

```

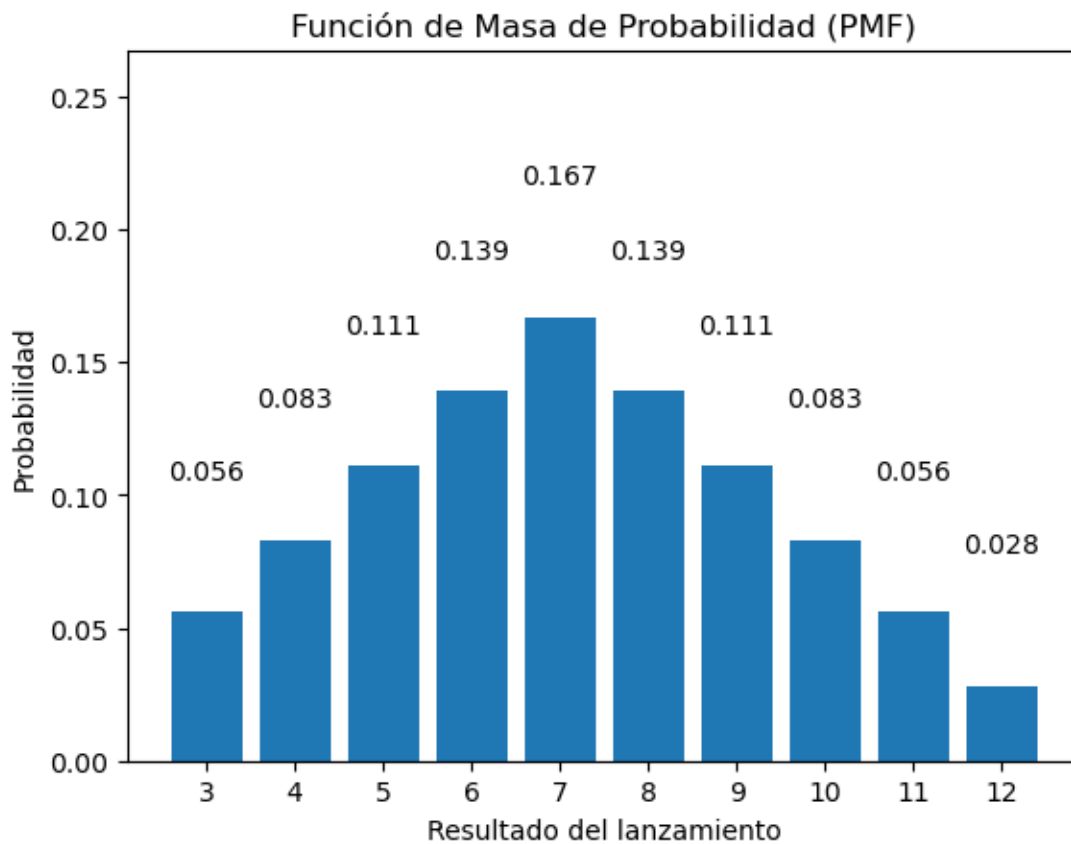
plt.text(i, prob + 0.05, str(prob), ha='center', fontsize=10)

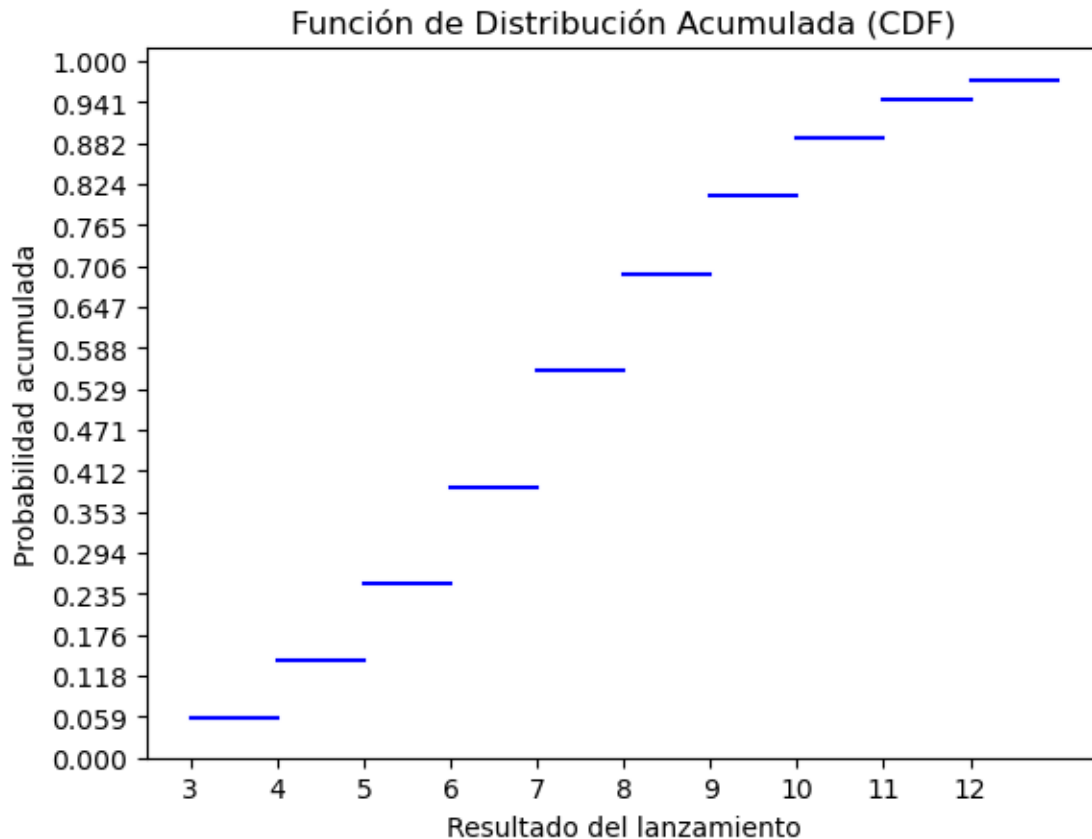
plt.show()

# Gráfico de escalón
acum = 0
for i in range(len(X)-1):
    acum += P[i]
    plt.step([X[i],X[i+1]], [acum, acum], where='pre', color='blue')
plt.step([X[-1], X[-1]+1], [sum(P), sum(P)], where='pre', color='blue')

plt.xlabel("Resultado del lanzamiento")
plt.ylabel("Probabilidad acumulada")
plt.title("Función de Distribución Acumulada (CDF)")
plt.xticks(X, [str(k) for k in X])
plt.yticks(np.linspace(0,1,int(1/P[0])+1))
plt.show()

```





#### 1.4 Ejercicio 5.

Supongamos que se lanza una moneda cuatro veces. Sea  $X$  el número de caras en la secuencia observada una variable aleatoria. Trace las representaciones geométricas de:

- la función de masa de probabilidad de  $X$
- la función de densidad acumulada de probabilidad de  $X$

```
[4]: A = set({'H', 'T'})
S = set(it.product(A,repeat=4))

R = [(m,len([k for k in S if k.count('H') == m])) for m in range(0,5)]
X = [k[0] for k in R] # Variable aleatoria
P = [round(k[1]/len(S),3) for k in R] # Medida de probabilidad de cada evento

# Gráfico de barras
plt.bar(X, P, tick_label=[str(k) for k in X])
plt.xlabel("Resultado del lanzamiento")
plt.ylabel("Probabilidad")
plt.title("Función de Masa de Probabilidad (PMF)")
```

```

plt.ylim(0, max(P)+0.1)

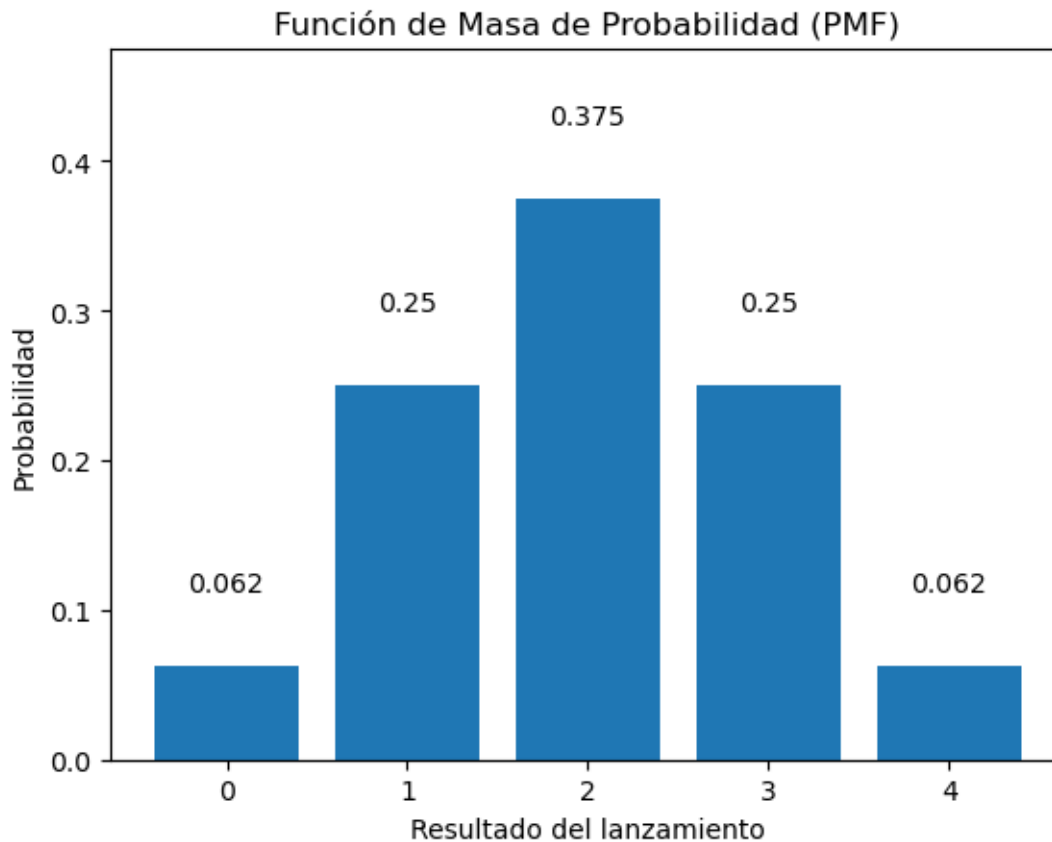
# Mostrar valores en las barras
for i, prob in zip(X, P):
    plt.text(i, prob + 0.05, str(prob), ha='center', fontsize=10)

plt.show()

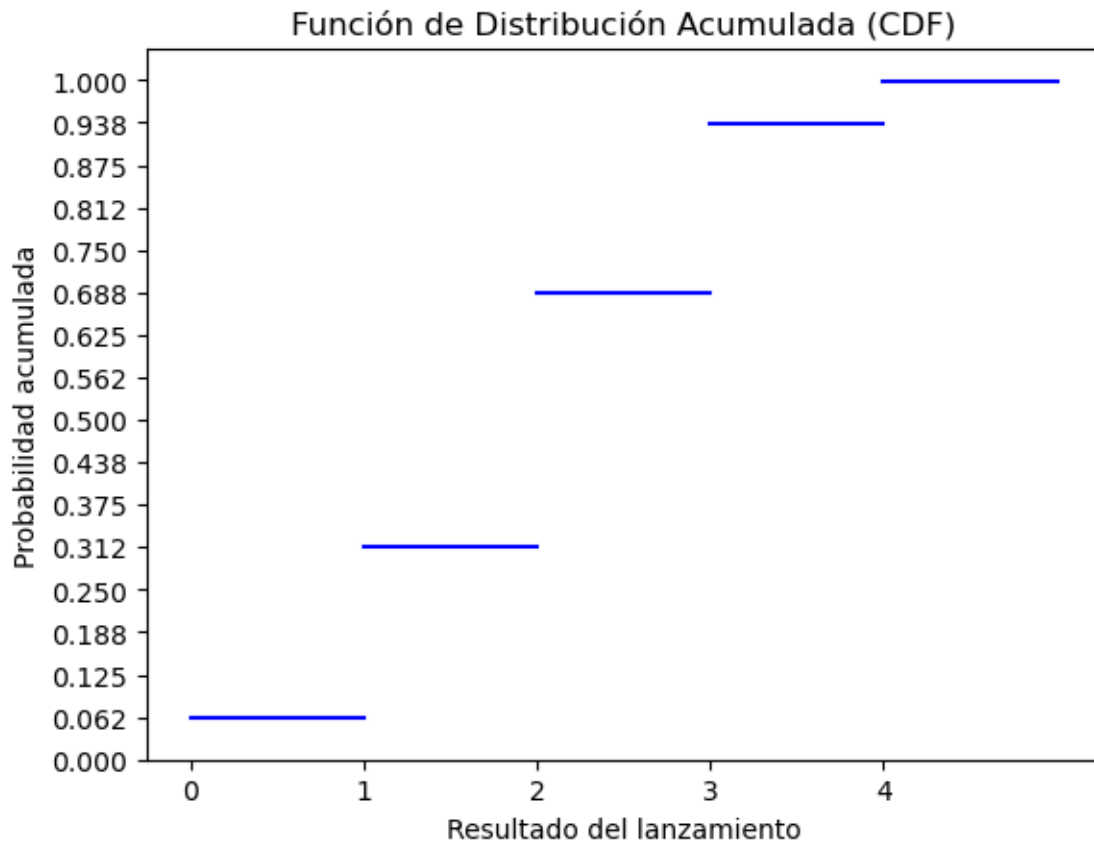
# Gráfico de escalón
acum = 0
for i in range(len(X)-1):
    acum += P[i]
    plt.step([X[i],X[i+1]], [acum, acum], where='pre', color='blue')
plt.step([X[-1], X[-1]+1], [sum(P), sum(P)], where='pre', color='blue')

plt.xlabel("Resultado del lanzamiento")
plt.ylabel("Probabilidad acumulada")
plt.title("Función de Distribución Acumulada (CDF)")
plt.xticks(X, [str(k) for k in X])
plt.yticks(np.linspace(0,1,int(1/P[0])+1))
plt.show()

```







### 1.5 Ejercicio 6.

En una caja con 20 pelotas numeradas del 1 al 20, se seleccionan cuatro pelotas al azar. Sea  $X$  el número más grande de pelota seleccionada una variable aleatoria.

- la función de masa de probabilidad de  $X$
- la función de densidad acumulada de probabilidad de  $X$

```
[16]: import itertools as it
import matplotlib.pyplot as plt
import numpy as np

A = set(range(1,21))
S = set(it.permutations(A,4))

R = [(m,len([k for k in S if max(k) == m])) for m in range(1,21)]
X = [k[0] for k in R] # Variable aleatoria
P = [k[1]/len(S) for k in R] # Medida de probabilidad de cada evento
```

```

# Gráfico de barras
plt.bar(X, P, tick_label=[str(k) for k in X])
plt.xlabel("Resultado del lanzamiento")
plt.ylabel("Probabilidad")
plt.title("Función de Masa de Probabilidad (PMF)")
plt.ylim(0, max(P)+0.1)

# Mostrar valores en las barras
for i, prob in zip(X, P):
    plt.text(i, prob + 0.05, str(round(prob, 2)), ha='center', fontsize=8)

plt.show()

# Gráfico de escalón
acum = 0
for i in range(len(X)-1):
    acum += P[i]
    plt.step([X[i], X[i+1]], [acum, acum], where='pre', color='blue')
plt.step([X[-1], X[-1]+1], [sum(P), sum(P)], where='pre', color='blue')

plt.xlabel("Resultado del lanzamiento")
plt.ylabel("Probabilidad acumulada")
plt.title("Función de Distribución Acumulada (CDF)")
plt.xticks(X, [str(k) for k in X])
#plt.yticks(np.linspace(0,1,int(1/P[0])+1))
plt.show()

```

