Física Estadística 2026-1 - Tarea 1

Emilio Moreno Ledesma

24 de agosto de 2025

Ejercicio 1: Volados. *Águila o sol*

Genera una gráfica de la frecuencia de soles $f(1)$ contra el número de volados n que haz lanzado hasta ese momento. Usa los datos obtenidos de:

a) 100 volados (hazlos a mano).

b) 1000 volados utilizando un generador aleatorio (a computadora).

c) 10,000 volados utilizando un generador aleatorio (a computadora).

¿Las fluctuaciones de $f(1)$ obtenidas en a), b) y c) disminuyen conforme más datos se agregan?

```
def toss_coins(N):
    accumulated_tails = np.zeros(N)
    accumulated_tails[0] = np.random.choice([1, 0])
    for i in range(1, N):
        accumulated_tails[i] = accumulated_tails[i-1] + np.random.choice([1, 0])
    frequency_tails = accumulated_tails / np.arange(1, N + 1)
    return accumulated_tails, frequency_tails


N = np.array([10**2, 10**3, 10**4])
accumulated = []
frequencies = []
# a)
accumulated_tails_100 = [1, 2, 2, 3, 4, 4, 4, 5, 5, 6,
                         7, 7, 7, 8, 9, 10, 11, 12, 12, 12,
                         12, 13, 14, 15, 15, 16, 17, 17, 17, 18,
                         19, 20, 21, 22, 22, 23, 24, 24, 25, 26,
                         27, 27, 28, 28, 28, 28, 29, 30, 30, 30,
                         31, 31, 31, 32, 33, 34, 34, 35, 36, 36,
                         37, 37, 38, 38, 39, 40, 41, 41, 42, 43,
                         44, 44, 44, 44, 44, 45, 46, 47, 47, 48,
                         48, 48, 49, 49, 49, 50, 51, 51, 51, 51,
                         51, 51, 52, 52, 52, 52, 53, 54, 55, 56]
accumulated.append(accumulated_tails_100)
frequencies.append(accumulated_tails_100 / np.arange(1, 101))

# b, c)
for n in N[1:]:
    accumulated_tails, frequency_tails = toss_coins(n)
    accumulated.append(accumulated_tails)
    frequencies.append(frequency_tails)
```

```python
fig, axs = plt.subplot_mosaic([["100"], ["1000"], ["10000"]], dpi=140)
fig.suptitle("Frecuencia de soles respecto a número de volados")

for ax_label, frequency in zip(axs, frequencies):
    axs[ax_label].plot(frequency, label=f"{ax_label} volados", zorder=1)
    axs[ax_label].axhline(1/2, label="Frecuencia esperada", c="r", ls="--", zorder=2)
    axs[ax_label].legend()
    axs[ax_label].set()

plt.subplots_adjust(hspace=0.4)
plt.savefig("Ejercicio1.png")
```

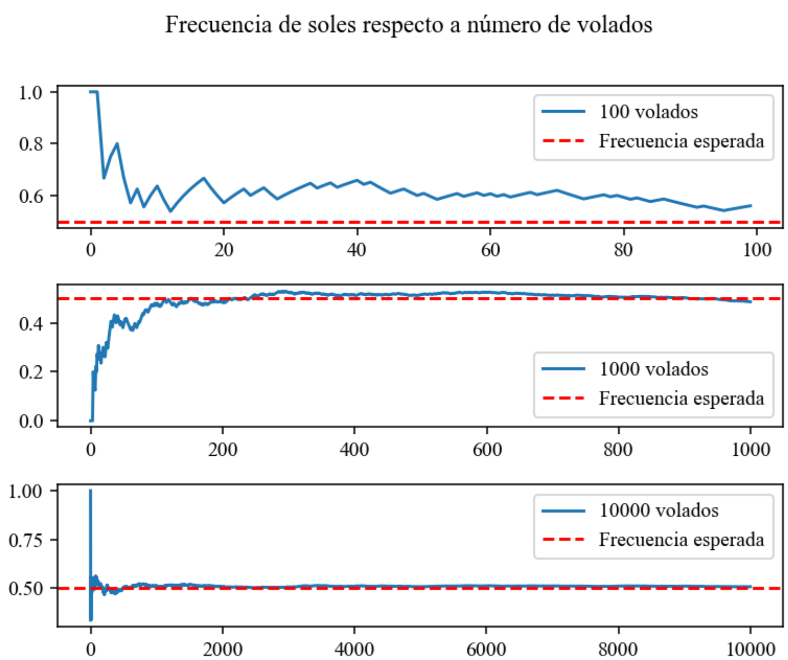

Figura 1: Frecuencia de soles vs número de volados.

Observemos que las fluctuaciones respecto a la frecuencia esperada de $1/2$ se reducen conforme aumentamos el número de volados. Notando que la frecuencia esperada es el valor esperado de la distribución de volados (μ), y que la frecuencia empírica es el promedio de la muestra ($\bar{x}$), podemos explicar este comportamiento observando que

$$\mathrm{var}\{\bar{x}\} = \mathrm{var}\left(\frac{\sum_i^N x_i}{N}\right) = \frac{\sum_i^N \mathrm{var}\,(x_i)}{N} = N\frac{\mathrm{var}\,(x_i)}{N} = \frac{\mathrm{var}\,(x_i)}{N}.$$

Así, vemos que la desviación del promedio respecto al valor esperado tiende a 0 conforme N aumenta.

Ejercicio 2: Modos de falla independientes. *Un motor*

Un sistema consiste de n componentes separados, donde cada uno falla independientemente de los otros con probabilidad P_i, donde $i = 1 \ldots n$. Dado que cada componente falla o no falla, la probabilidad de que el i-ésimo componente no falle es $1 - P_i$.

a) Supón que los componentes están conectados en paralelo de tal manera que la falla de todos los componentes es necesaria para que el sistema falle. ¿Cuál es la probabilidad de que el sistema falle?, y ¿cuál es la probabilidad de que el sistema funcione (no falle)?

2.a.

Sea x_i una variable indicadora para el componente i. Si $x_i = 1$, el componente falla y si $x_i = 0$, funciona. Así

$$P(x_i = 1) = P_i, \quad P(x_i = 0) = 1 - P_i.$$

Ahora, para que el sistema en paralelo falle, todos los componentes deben fallar. Esto sucede con probabilidad:

$$P(x_1 = 1 \wedge x_2 = 1 \wedge \dots x_n = 1) = \prod_{i=1}^{n} P(x_i = 1) = \prod_{i=1}^{n} P_i.$$

Donde hemos usado que la falla de componentes es independiente para la primera igualdad.

Ahora, la probabilidad de que el sistema funcione P_f más la probabilidad que el sistema no funcione P_{nf} deben sumar 1, pues son las únicas dos opciones del sistema. Así:

$$P_f = 1 - P_{nf} = 1 - \prod_{i=1}^{n} P_i.$$

b) Supón que los componentes están conectados en serie, de tal manera que la falla de un componente causa la falla de todo el sistema. ¿Cuál es la probabilidad de que el sistema falle? (Pista: Primero encuentra la probabilidad de que todos los componentes funcionen.)

b) Para que el sistema funcione en serie, todos los componentes deben de funcionar:

$$P_f = P(x_1 = 0 \wedge x_2 = 0 \wedge \ldots x_n = 0) = \prod_{i=1}^{n} P(x_i = 0)$$
$$= \prod_{i=1}^{n} (1 - P_i)$$

Donde usamos nuevamente la independencia de fallo. Así, la probabilidad de que el sistema falle es, usando $P_f + P_{nf} = 1$:

$$P_{nf} = 1 - P_f = 1 - \prod_{i=1}^{n} (1 - P_i).$$

Ejercicio 3: Caminantes Aleatorios.

Supón que tienes un caminante aleatorio "con duda", es decir, que en una iteración dada además de tener la posibilidad de moverse una unidad tanto hacia adelante como hacia atrás, también puede decidir no moverse. Calcula la varianza de la posición del caminante como función del número de pasos, n, tomados en los siguientes casos:

a) $P(x_i = -1) = \frac{1}{3},\ P(x_i = 0) = \frac{1}{3},\ P(x_i = 1) = \frac{1}{3}$

b) $P(x_i = -1) = \frac{1}{4},\ P(x_i = 0) = \frac{1}{2},\ P(x_i = 1) = \frac{1}{4}$

c) $P(x_i = -1) = \frac{1}{4},\ P(x_i = 0) = \frac{1}{4},\ P(x_i = 1) = \frac{1}{2}$

Ejercicio 3:

a). $P(X_i=-1)=1/3$, $P(X_i=0)=1/3$, $P(X_i=1)=1/3$.

Tenemos $\langle X_i\rangle = \frac{1}{3}(-1)+\frac{1}{3}\cdot 0+\frac{1}{3}\cdot 1 = 0$, y

$\langle X_i^2\rangle = \frac{1}{3}(-1)^2+\frac{1}{3}\cdot 0^2+\frac{1}{3}\cdot 1^2=\frac{2}{3}$. Recuerda que

$$\mathrm{var}\, X_i = \langle X_i^2\rangle - \langle X_i\rangle^2 = \frac{2}{3} \text{ y } \mathrm{var}\, Y_n = \mathrm{var}\left(\sum_{i=1}^{n} X_i\right)$$

$$\underset{\substack{\uparrow\\ \text{independencia}}}{=} \sum_{i=1}^{n} \mathrm{var}\, X_i = \frac{2}{3}\sum_{i=1}^{n} 1 = \frac{2}{3} n.$$

b) $P(X_i=-1)=1/4$, $P(X_i=0)=1/2$, $P(X_i=1)=1/4$

$$\Rightarrow \langle X_i\rangle = \frac{1}{4}\cdot(-1)+\frac{1}{2}\cdot 0+\frac{1}{4}\cdot 1 = 0,\quad \langle X_i^2\rangle = \frac{1}{4}\cdot(-1)^2+\frac{1}{2}\cdot 0^2+\frac{1}{4}\cdot 1^2=\frac{1}{2}$$

$$\Rightarrow \mathrm{var}\, X_i = \langle X_i^2\rangle - \langle X_i\rangle = \frac{1}{2}$$

$$\Rightarrow \mathrm{var}\, Y_n = \mathrm{var}\left(\sum_{i=1}^{n} X_i\right) \underset{\substack{\uparrow\\ \text{ind.}}}{=} \sum_{i=1}^{n} \mathrm{var}\, X_i = \frac{1}{2} n.$$

c) $P(X_i=-1)=1/4$, $P(X_i=0)=1/4$, $P(X_i=1)=1/2$

$$\langle X_i\rangle = \frac{1}{4}\cdot(-1)+\frac{1}{4}\cdot 0+\frac{1}{2}\cdot 1 = \frac{1}{4},\quad \langle X_i^2\rangle = \frac{1}{4}(-1)^2+\frac{1}{4}\cdot 0+\frac{1}{2}\cdot 1=\frac{3}{4}$$

$$\Rightarrow \mathrm{var}\, X_i = \langle X_i^2\rangle - \langle X_i\rangle^2 = \frac{3}{4}-\frac{1}{16}=\frac{11}{16}.$$

$$\Rightarrow \mathrm{var}\, Y_n = \mathrm{var}\left(\sum_{i=1}^{n} X_i\right) \underset{\substack{\uparrow\\ \text{ind.}}}{=} \sum_{i=1}^{n} \mathrm{var}\, X_i = \frac{11}{16} n.$$

Ejercicio 4: Más de caminantes aleatorios.

Para el sistema planteado en el inciso 3a), elabora 1000 listas (1000 caminantes) de 100[1] pasos cada una.

a) Gráfica 20 de las 1000 listas junto con la desviación estándar de la expresión analítica que encontraste

[1] Originalmente pone 30 aquí, pero usa 100 pasos para las gráficas. Como el comportamiento de las variables se ve mejor con 100 pasos, decidí utilizar eso.

h!

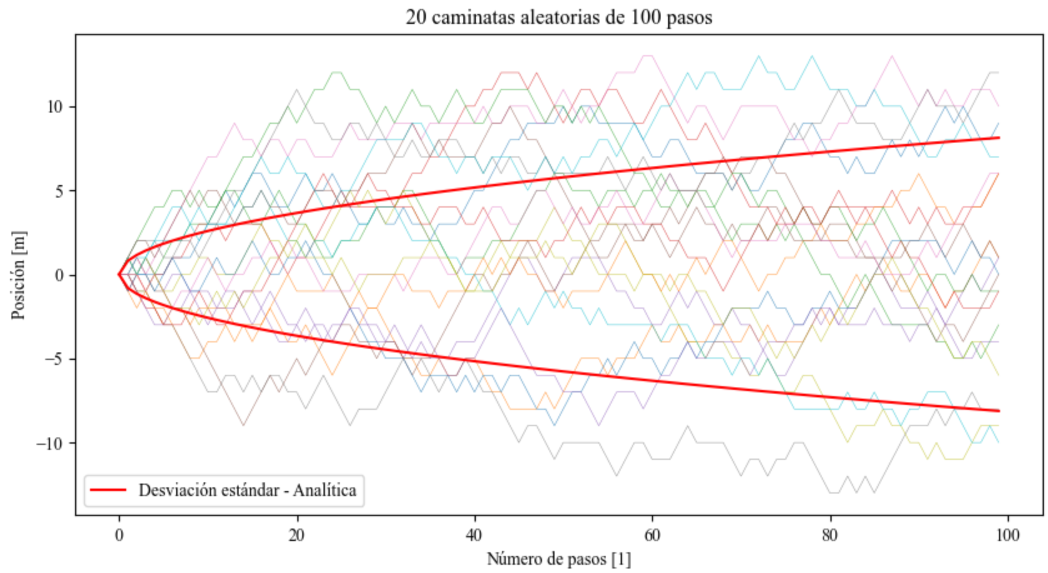

Figura 2: Comparación de 20 caminatas aleatorias de 100 pasos con la desviación estándar calculada en el Ejercicio 3.a)

para este sistema en el inciso 3a).

```python
# a)
n_steps = 10**2
n_walks = 10**3
# Array para las caminatas
# La i-ésima fila corresponde a la posición de todas las caminatas en
# el paso i. Mientras que la j-ésima columna corresponde a la caminata
# j completa.
walks = np.zeros((n_steps, n_walks))

for step in range(1, n_steps):
    walks[step] = random.choices([1, 0, -1], weights=[1/3, 1/3, 1/3], k=n_walks) + walks[step - 1]


n_walks_plot = 20
steps = np.arange(0, n_steps)
std_analytic = np.sqrt(2/3 * steps)

fig, ax = plt.subplots(figsize=(10, 5))
ax.plot(walks[:, 0:n_walks_plot], lw=0.3)
ax.plot(steps, std_analytic, color='r', label="Desviación estándar - Analítica")
ax.plot(steps, -std_analytic, color='r')
ax.set(title=f"{n_walks_plot} caminatas aleatorias de {n_steps} pasos",
       xlabel="Número de pasos [1]", ylabel="Posición [m]")

ax.legend()
plt.savefig("Ejercicio4a.png")
```

b) ¿Qué puedes decir del comportamiento de las caminatas individuales comparadas con la desviación estándar del inciso 3a)?

En el inciso 3.a) encontramos que la desviación estándar está dada por $\sigma(n) = \sqrt{2/3N}$ con N el número de pasos. Las líneas rojas en la gráfica de arriba corresponden a $\sigma(n)$, $-\sigma(n)$ y observamos que la mayoría de las caminatas se encuentran dentro de este intervalo; las caminatas tienden a agruparse a una desviación estándar del valor esperado 0.

c) Ahora, considerando los siguientes grupos de listas, calcula el promedio de la posición como función del número de pasos y gráficalo.

I) Grupo 1: Incluye 5 caminatas.

II) Grupo 2: Incluye 100 caminatas.

III) Grupo 3: Incluye 500 caminatas.

IV) Grupo 4: Incluye 1000 caminatas.

```
# c)
n_members = [5, 100, 500, 1000]
walk_groups = []

for n in n_members:
    walk_groups.append(walks[:, 0:n])

group_averages = []
group_squared_averages = []
for group in walk_groups:
    group_average = np.zeros(n_steps)
    group_squared_average = np.zeros(n_steps)
    for n in range(0, n_steps):
        group_average[n] = group[n, :].mean()
        group_squared_average[n] = (group[n, :]**2).mean()
    group_averages.append(group_average)
    group_squared_averages.append(group_squared_average)

group_variances = np.array(group_squared_averages) - np.array(group_averages)**2


fig, ax = plt.subplots(figsize=(10, 5))

for n, average in zip(n_members, group_averages):
    ax.plot(average, lw=0.5, label=f"{n} caminatas", marker='.', markersize=1)

ax.axhline(0, color='r', label="Valor esperado")
ax.set(title=f"Promedio en función del número de pasos",
       xlabel="Número de pasos [1]", ylabel="Promedio [m]")

ax.legend()
plt.savefig("Ejercicio4c.png")
```

d) ¿Qué puedes decir del comportamiento de los distintos conjuntos de las caminatas comparadas con 0?

El promedio de la muestra sigue, como vimos en el Ejercicio 1, $\text{var}(\bar{S}) = \frac{\text{var}(S_i)}{M} = \frac{3/2N}{M}$, siendo M el número de caminatas que promediamos y N el número de pasos. Este es el comportamiento que observamos en la gráfica de arriba: para valores bajos de M, la desviación del promedio de la muestra crece rápidamente conforme aumenta el número de pasos; cuando promediamos más caminatas, la variación del promedio respecto al valor esperado crece más lentamente.

e) Calcula las varianzas para cada uno de los pasos n, para cada uno de los grupos de caminatas.

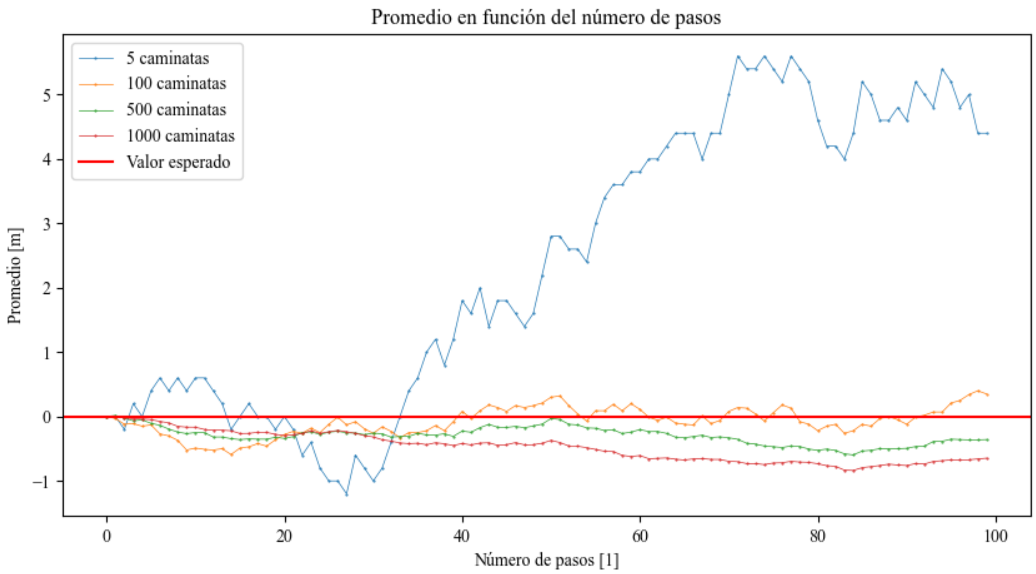

Figura 3: Promedio de la muestra para grupos de distinto número de caminatas vs el número de pasos.

```
fig, ax = plt.subplots(figsize=(10, 5))

for n, variance in zip(n_members, group_variances):
    ax.plot(variance, lw=0.5, label=f"{n} caminatas", marker='.', markersize=1)

var_X = 2 / 3
ax.plot((0, n_steps), (0, var_X * n_steps), color='r', label="Varianza esperada")
ax.set(title=f"Varianza en función del número de pasos",
       xlabel="Número de pasos [1]", ylabel="Varianza [m]")

ax.legend()
plt.savefig("Ejercicio4d.png")
```

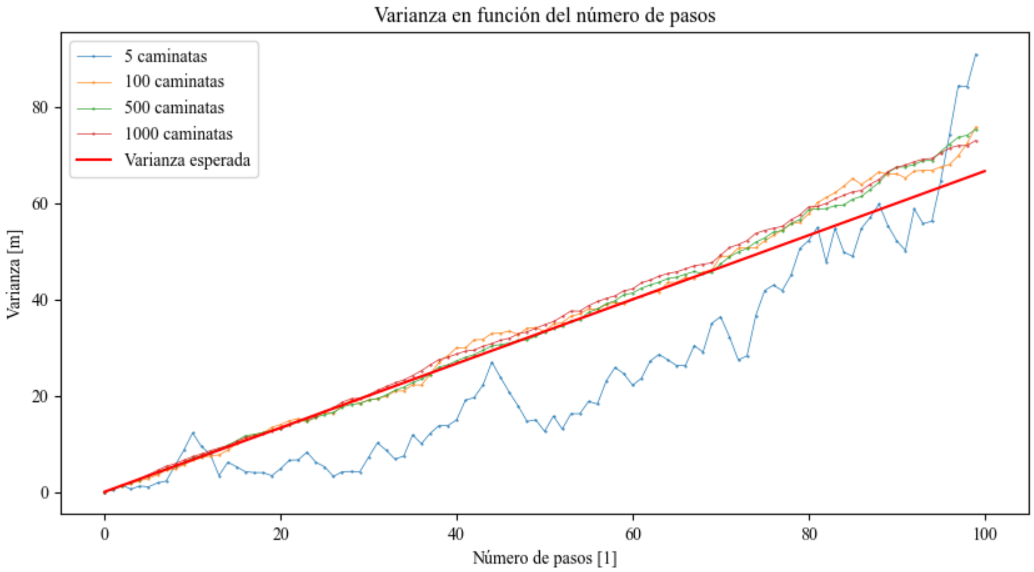

Figura 4: Variación de la muestra para grupos de distinto número de caminatas vs el número de pasos.

f) ¿Qué observas?, ¿qué puedes concluir?

Observamos un comportamiento similar al inciso d): conforme el número de caminatas utilizadas para calcular la varianza empírica aumenta, la varianza empírica varía en menor medida respecto a la varianza esperada.

Cualitativamente, la varianza de la muestra es una función del número de caminatas M y el número de pasos N: $\varsigma^2 = \varsigma^2(M, N)$. Además, como la varianza de cada caminata aumenta con N (i.e., $\sigma^2 = 3/2N$), ς^2 también aumenta. Así, para un valor de M fijo, ς^2 crece con el número de pasos, produciendo mayor variación respecto a la varianza esperada.