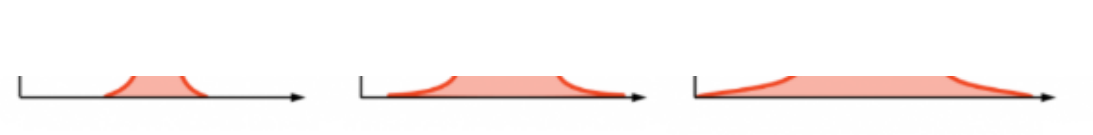


Estadística Inferencial

"La estadística inferencial es una parte de la estadística que comprende los métodos y procedimientos que por medio de la inducción determina propiedades de una población estadística, a partir de una parte de esta. Su objetivo es obtener conclusiones útiles para hacer razonamientos deductivos sobre una totalidad, basándose en la información numérica dada por la muestra.

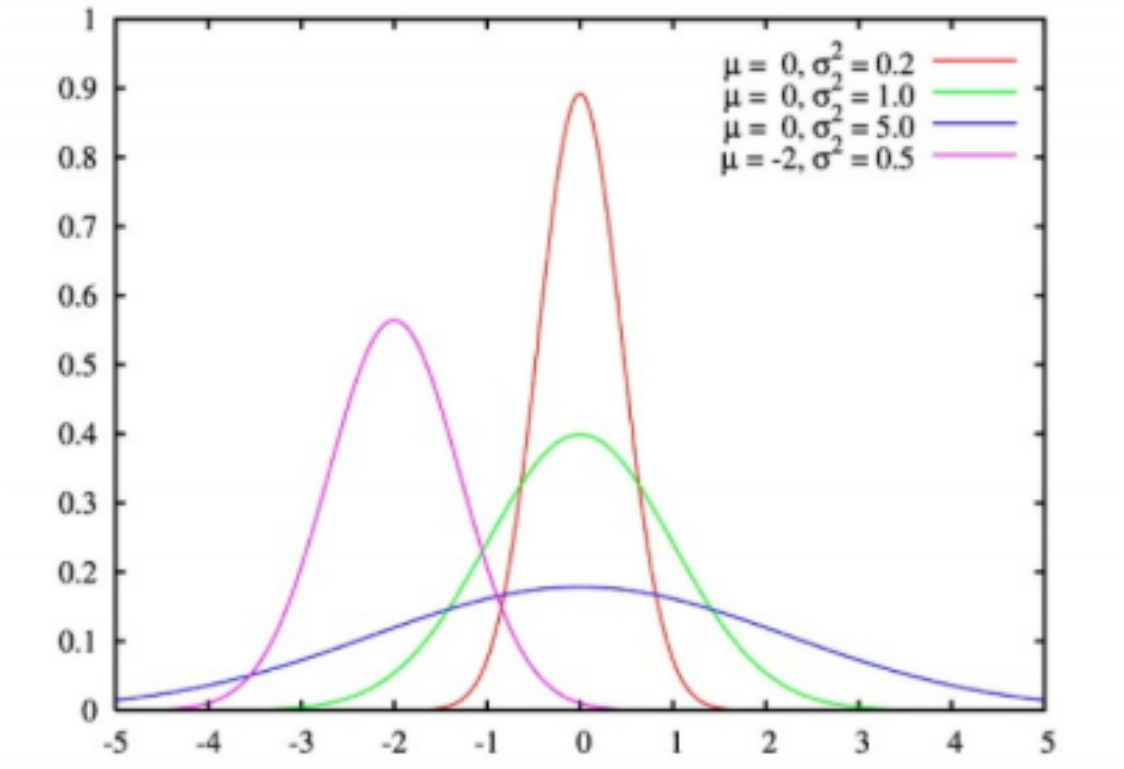
Se dedica a la generación de los modelos y predicciones asociadas a los fenómenos en cuestión teniendo en cuenta la aleatoriedad de las observaciones. Se usa para modelar patrones en los datos y extraer inferencias acerca de la población bajo estudio." definición completa aquí

Como su nombre lo dice, la estadística inferencial nos permite aprender propiedades de la población que queremos estudiar, sin tener que medir cada uno de los elementos de la población. Imaginemos que queremos saber cual es la altura media de los hombres de entre 18-30 años, ¿es razonable esperar que se tomen medidas de **todos los hombres de entre 18-30 años** en el mundo? Parece un poco disparatado.

La inferencia estadística nos permite obtener pistas o aproximaciones de este comportamiento a partir del comportamiento de un subconjunto de ésta. Por ejemplo, en el caso de la altura media, podríamos tomar la altura de 200 hombres de entre 18-30 en cada uno de los 5 continentes, e intentar "estimar" cual sería la altura promedio de todos los hombres de este grupo, a partir de esa medida en un subgrupo del total.

Distribución normal y el Teorema del límite central

Un tema muy frecuente cuando se habla de estadística es la distribución Gaussiana o normal. Esta distribución tiene una forma de campana en la que encontramos la mayoría de los valores de la distribución concentrados alrededor de la media. Es una función **simétrica** y **unimodal**. Muchas veces escucharemos que muchas cosas tienen una distribución que se puede modelar como una distribución normal, lo cual es **falso** en su mayoría. Esta confusión surge de un resultado muy importante en la estadística, conocido como el **Teorema del Límite Central**.



La distribución normal se caracteriza por estar centrada en la **media**. Existe una distribución normal "unitaria" o "estándar" que

tiene media 0 y desviación estándar 1, que es la distribución básica de la estadística inferencial. Si observamos, la distribución normal tiene una propiedad interesante, y es que la media, mediana y la moda coinciden.

Teorema del Límite Central

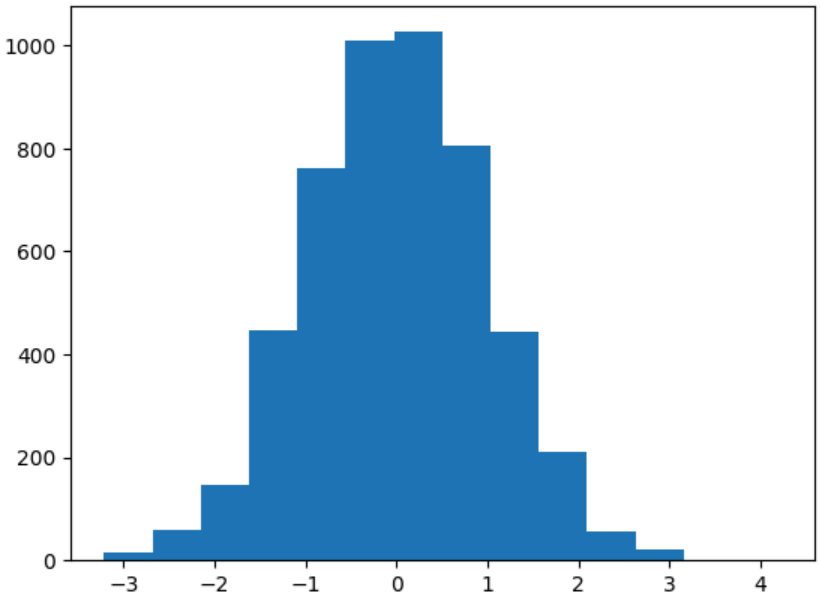
Y si no es cierto que muchas cosas tienen distribución normal? Para qué la utilizamos? **La distribución normal es la base de la estadística inferencial!!** Simplemente que no lo es porque la mayoría de las cosas se comporten como una distribución normal, sino porque se puede demostrar que la distribución de las medias de todas las distribuciones tienen a la distribución normal si la muestra es suficientemente grande!

Este resultado es muy potente y es la base de los tests de hipótesis que utilizaremos para determinar el comportamiento de nuestra población a partir de la información que obtenemos de una o varias muestras.

Debajo utilizaremos la función np.random.randn() para generar una muestra de 5000 puntos obtenidos de una distribución normal estándar (media cero y desviación estándar 1).

In [25]:
```
np.random.seed(4)

y = np.random.randn(5000)
plt.hist(y, bins=14)
```

Out[25]:
```
(array([1.400e+01, 6.000e+01, 1.460e+02, 4.460e+02, 7.610e+02, 1.008e+03,
        1.026e+03, 8.050e+02, 4.430e+02, 2.120e+02, 5.600e+01, 2.000e+01,
        2.000e+00, 1.000e+00]),
 array([-3.2146555 , -2.68387168, -2.15308787, -1.62230406, -1.09152025,
        -0.56073644, -0.02995262,  0.50083119,  1.031615  ,  1.56239881,
         2.09318263,  2.62396644,  3.15475025,  3.68553406,  4.21631787]),
 <BarContainer object of 14 artists>)
```



Otras distribuciones

También existen otras distribuciones que nos permiten modelar fenómenos utilizando otras expresiones matemáticas. Veremos que hay dos tipos de distribuciones: **continuas**, que responden a una variable numérica que puede tomar cualquier valor, o **discretas**, que responden a una variable que solo puede tomar un número finito de valores posibles.

No veremos estas distribuciones en detalle, pero se incluyen debajo algunas de las distribuciones más utilizadas y una breve explicación de para qué se utilizan o qué intentan medir, aunque pueden leer más sobre esto aquí:

- Continuas:
  - Uniforme: Mide la probabilidad de obtener un resultado, cuando todos los resultados tienen la misma probabilidad de ocurrir.
  - Chi-cuadrado: Es una curva con k grados de libertad que a medida que k crece se "degrada" a la distribución normal. Se utiliza mucho en tests estadísticos para medir la bondad de ajuste de una distribución a la estimada y para medir si dos variables aleatorias son independientes.

    - Exponencial: (ver Poisson) Modela el tiempo que transcurrirá entre dos eventos que tienen distribución de Poisson.
    - Student: La distribución t de Student es una distribución de probabilidad que surge del problema de estimar la media de una población que tiene distribución normal cuando el tamaño de la muestra es pequeño y la desviación estándar poblacional es desconocida.
- Discretas:
    - Bernoulli: Dada la probabilidad de que suceda un evento, modela la frecuencia con la que ocurrirá cada uno de los eventos.
    - Binomial: Generalización de Bernoulli. Modela N eventos independientes, cada uno con una probabilidad p de suceder
    - Poisson: Esta distribución expresa la probabilidad de que un número de eventos dado (con distribuciones binomiales) ocurra en un intervalo de tiempo (o espacio) fijo si los eventos ocurren con una frecuencia constante y son independientes (no dependen de cuándo ocurrió el último evento).

Correlación

La correlación es una medida entre dos variables que cuantifica **cómo se comporta una cuando la otra cambia**. Existen muchas maneras de medir correlación, y sobre todo debemos tener mucho cuidado para saber qué tipo de correlación estamos midiendo, ya que muchas veces obtendremos un resultado que no tiene un nivel de correlación bueno, pero existe dependencia entre las variables. Esto es muy común cuando existe dependencia no lineal entre las variables!

Todas las medidas de correlación tienen el mismo principio de funcionamiento:

- Correlación positiva implica que cuando una variable crece, la otra también lo hace.
- Correlación negativa implica que cuando una variable creca, la otra disminuye.
- Correlación neutra implica que no se observa una relación de dependencia entre el comportamiento de ambas variables (**importante:** no se observa dependencia basado en el tipo de correlación que estamos midiendo!)

Podemos calcular la correlación entre dos o más variables utilizando NumPy con la función corrcoef, o utilizando el método .corr() que provee Pandas. Como ya tenemos cargado el dataframe de pandas con los datos de salarios anuales, utilizaremos éste último para medir la correlación entre la edad de las personas y sus ingresos anuales. Una observación importante es que todas las variables correlan al 100% con ellas mismas, por lo que solo debemos mirar los cruces entre ambas variables.

In [26]:
```
df_kaggle[['income', 'age']].corr()
```

Out[26]:

	income	age
income	1.000000	0.464433
age	0.464433	1.000000

La correlación entre income y age es baja positiva, lo que quiere decir que cuando la edad aumenta se observa una tendencia al aumento de los ingresos también.

Covarianza

La covarianza es una medida que indica la fuerza de la correlación entre dos variables respecto de su media. Podríamos decir que la correlación es una medida adimensional (o escalada) de la covarianza. Dicho de otra forma, la covarianza es sensible a la escala en la que se calcula, si las unidades son más grandes, los valores de covarianza serán más grandes, mientras que la correlación se mantendrá igual. Un valor de covarianza de cero indica que ambas variables son completamente independientes.

El coeficiente de correlación más utilizado en la práctica es el de Pearson, que calcula la correlación como la covarianza entre

dos variables dividido entre el producto de las desviaciones estandar de cada variable.

Podemos calcular la covarianza entre dos variables usando la función np.cov(). Cuidado! Si utilizamos columnas de DataFrames, debemos trasponer el resultado para obtener la matriz de covarianza!

In [27]:
```
df_kaggle[['income','age']].values
```

Out[27]:
```
array([[8.65196085e+04, 4.50000000e+01],
       [8.30858650e+04, 3.00000000e+01],
       [8.26062150e+04, 2.20000000e+01],
       ...,
       [1.50000000e+04, 3.20000000e+01],
       [1.50000000e+04, 2.10000000e+01],
       [1.50000000e+04, 2.70000000e+01]], shape=(100000, 2))
```

In [28]:
```
np.cov(df_kaggle[['income','age']].values.T)
```

Out[28]:
```
array([[1.17306078e+08, 4.72605802e+04],
       [4.72605802e+04, 8.82736017e+01]])
```

Test de Hipótesis

En estadística, las pruebas de hipótesis calculan la probabilidad de que un evento suceda, asumiendo que se cumple una cierta hipótesis de partida, llamada **H0**. Cuando hacemos un test de hipótesis generalmente queremos descartar la hipótesis H0 que asumimos, y por lo tanto nos interesa que esa probabilidad sea muy pequeña. Esta probabilidad que calculamos en estadística se conoce como el **p-valor** (p-value en inglés) y en general se busca que su valor sea menor que un valor arbitrario que se define como "poco probable", generalmente 5% o 0.05, pero podemos elegir el que consideremos mejor.

Si la probabilidad de que algo suceda dado que la Hipótesis H0 se cumple es muy muy baja, entonces podemos asumir que es poco probable que la hipótesis H0 se cumpla, y por lo tanto la descartamos. Si esto no es así, no podemos asegurar nada sobre la hipótesis H0.

Algunos casos de uso habituales de los test de hipótesis en ciencia de datos son:

- Pruebas de normalidad: queremos estimar si nuestros datos provienen de una distribución normal
- Pruebas de origan: queremos determinar si dos muestras provienen de la misma distribución
- Pruebas de mejora: queremos determinar si las nuevas mediciones realizadas después de un cambio son significativamente mejores que las anteriores (vale la pena cambiar la metodología?).

Pruebas de normalidad

Una aplicación buena de un test de Hipótesis son las llamadas pruebas de normalidad. Queremos saber si nuestros datos provienen de una distribución normal, o si debemos aplicar algún tipo de normalización (como veremos más adelante). Para ello podemos utilizar alguno de los tests de normalidad más frecuentes:

- Shapiro-Wilk
- Kolmogorov-Smirnov
- D'Agostino

La hipótesis nula (H0) para todos estos tests es que la función tiene distribuión normal, por lo tanto si realizamos el test y obtenemos un p-valor menor a 0.05, podemos decir que los datos no provienen de la distribución normal. Para ponerlo en práctica generaremos dos muestras: X1 será una nube de puntos aleatorios, X2 será una distribución normal.

Para importar el test de Shapiro-Wilk podemos utilizar la librería scipy.stats y utilizar el método .shapiro(). Para generar los puntos utilizaremos np.random.random() para generar números aleatorios, y np.random.randn() para generar números de una distribución normal estándar.

In [ ]:
```
X1 = np.random.random(100)
X2 = np.random.randn(100)

shap_x1 = stats.shapiro(X1)
shap_x2 = stats.shapiro(X2)
```

In [30]:
```
print(shap_x1)
print(shap_x2)
```

```
ShapiroResult(statistic=np.float64(0.9521330473092638), pvalue=np.float64(0.0011496052537900638))
ShapiroResult(statistic=np.float64(0.9845663191382376), pvalue=np.float64(0.2954025101733949))
```

Vemos que el p-valor para la distribución 1 (aleatoria) es muy pequeño, esto nos permite descartar la hipótesis de que los datos vienen de una distribución normal. No podemos asegurar nada de la segunda distribución (a pesar de que sabemos que proviene de una distribución normal).

Pruebas de Independencia

Otra aplicación interesante es la de las pruebas de independencia. Esto es si existe alguna asociación entre dos variables. Si las variables fueran categóricas, por ejemplo, positivo/negativo vs franja etárea, podríamos ver si existe alguna relación de dependencia entre el diagnóstico (positivo/negativo) y la edad (joven/adulto/adulto mayor).

Para esto se utilizan tablas de conteos con cada una de las ocurrencias para cada una de las combinaciones de categorías. Supongamos que tenemos la siguiente tabla:

	Joven	Adulto	Adulto Mayor
Positivo	10	20	30
Negativo	6	9	17

In [31]:

```
tabla = [[10, 20, 30],[6, 9, 17]]
stat, p, grados, t_esperada = stats.chi2_contingency(tabla)
print('grados=%d' % grados)
print(t_esperada)

# Prueba Chi-2
prob = 0.95
critico = stats.chi2.ppf(prob, grados)
print('Probabilidad=%.3f, Valor Critico=%.3f, Estadistico=%.3f' % (prob, critico, stat))
# interpret p-value
alpha = 1.0 - prob
print('Significancia=%.3f, p=%.3f' % (alpha, p))
if p <= alpha:
    print('Dependiente (Rechaza H0)')
else:
    print('Independent (No rechaza H0)')
```

```
grados=2
[[10.43478261 18.91304348 30.65217391]
 [ 5.56521739 10.08695652 16.34782609]]
Probabilidad=0.950, Valor Critico=5.991, Estadistico=0.272
Significancia=0.050, p=0.873
Independent (No rechaza H0)
```

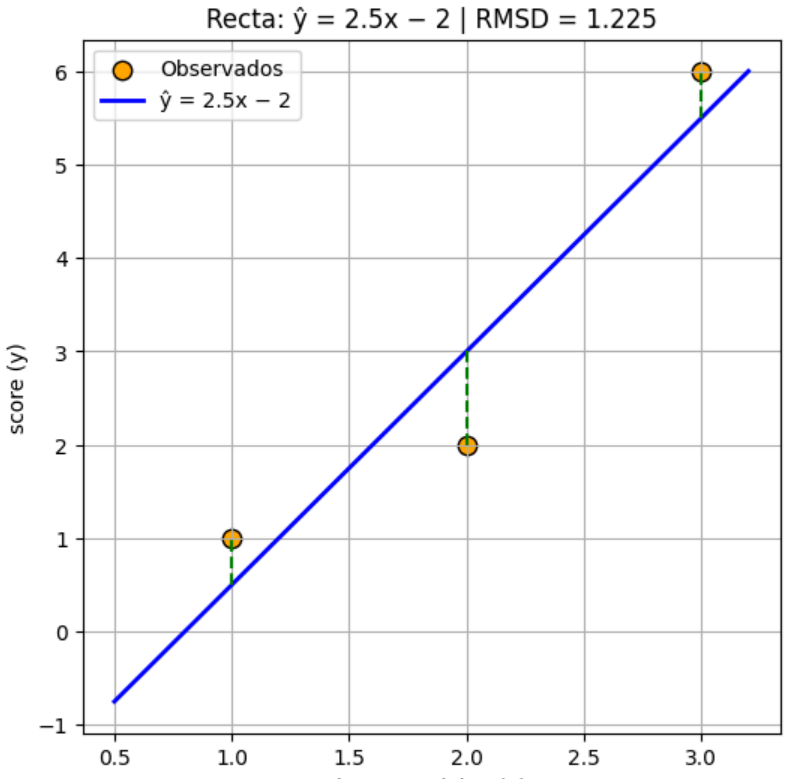
Ejemplos de regresion

In [32]:

```
import math
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

# =====================
# Datos de ejemplo
# =====================
x = np.array([1, 2, 3], dtype=float)
y = np.array([1, 2, 6], dtype=float)   # Valores observados

# Recta dada: ŷ = 2.5x - 2
def y_hat(x):
    return 2.5 * x - 2

y_pred = y_hat(x)
residuals = y - y_pred

# =====================
# Cálculo de RMSD
# =====================
n = len(x)
rss = np.sum(residuals**2)
```

```python
rmsd = math.sqrt(rss / (n - 2)) if n > 2 else float('nan')

# =====================
# Tabla de resultados
# =====================
df = pd.DataFrame({
    "x": x,
    "y": y,
    "ŷ = 2.5x - 2": y_pred,
    "residuo (y - ŷ)": residuals,
    "(y - ŷ)^2": residuals**2,
})
print(df)
print("\nRSS:", round(rss, 4))
print("RMSD:", round(rmsd, 4))

# =====================
# Gráfica
# =====================
plt.figure(figsize=(6,6))

# Puntos observados
plt.scatter(x, y, color="orange", edgecolor="k", s=80, label="Observados")

# Línea de regresión
x_line = np.linspace(0.5, 3.2, 100)
plt.plot(x_line, y_hat(x_line), "b-", linewidth=2, label="ŷ = 2.5x − 2")

# Residuos (líneas verticales)
for xi, yi, ypi in zip(x, y, y_pred):
    plt.plot([xi, xi], [ypi, yi], "g--")

# Etiquetas
plt.xlabel("hours studying (x)")
plt.ylabel("score (y)")
plt.title(f"Recta: ŷ = 2.5x − 2 | RMSD = {rmsd:.3f}")
plt.grid(True)
plt.legend()

plt.show()
```

```
     x    y  ŷ = 2.5x - 2  residuo (y - ŷ)  (y - ŷ)^2
0  1.0  1.0           0.5              0.5       0.25
1  2.0  2.0           3.0             -1.0       1.00
2  3.0  6.0           5.5              0.5       0.25

RSS: 1.5
RMSD: 1.2247
```



hours studying (x)

Regresion lineal

Qué representa cada columna

- **x**: la variable independiente (horas de estudio).
- **y**: el valor observado (puntaje real).
- **ŷ = 2.5x − 2**: el valor **predicho** por la recta de regresión dada.
- **residuo (y − ŷ)**: el **error** de predicción para cada punto (lo que realmente pasó menos lo que el modelo predijo).
  - Residuo > 0: el punto está **por encima** de la recta.
  - Residuo < 0: el punto está **por debajo** de la recta.
- **(y − ŷ)²**: el residuo **al cuadrado** (sirve para medir el error total penalizando más los errores grandes).

Cálculo fila por fila

Fila 0

- $x = 1, y = 1$
- Predicción: $\hat{y} = 2.5(1) - 2 = 0.5$
- Residuo: $y - \hat{y} = 1 - 0.5 = 0.5$ → punto **0.5** unidades **arriba** de la recta
- Cuadrado del residuo: $0.5^2 = 0.25$

Fila 1

- $x = 2, y = 2$
- Predicción: $\hat{y} = 2.5(2) - 2 = 3$
- Residuo: $2 - 3 = -1$ → punto **1** unidad **abajo** de la recta
- Cuadrado del residuo: $(-1)^2 = 1$

Fila 2

- $x = 3, y = 6$
- Predicción: $\hat{y} = 2.5(3) - 2 = 5.5$
- Residuo: $6 - 5.5 = 0.5$ → punto **0.5** unidades **arriba** de la recta
- Cuadrado del residuo: $0.5^2 = 0.25$

Resumen numérico y qué significa

- **Residuos**: $0.5,\ -1,\ 0.5$
  - La suma $0.5 - 1 + 0.5 = 0$. Esto es coherente con una propiedad típica de la regresión con intercepto: los residuos suelen sumar ~0 (aunque aquí la recta fue dada, ocurre igual por estos datos).
- **Suma de cuadrados de residuos (RSS)**:

$$0.25 + 1 + 0.25 = 1.5$$

- **RMSD (desviación típica de los residuos)** para regresión lineal con dos parámetros (pendiente e intercepto):

$$\text{RMSD} = \sqrt{\frac{\text{RSS}}{n-2}} = \sqrt{\frac{1.5}{3-2}} = \sqrt{1.5} \approx 1.225$$

  - Se interpreta como el **error vertical típico** de la recta respecto a los puntos, **en las mismas unidades de** y (puntaje).
  - Aquí, en promedio, la recta se equivoca alrededor de **1.225 puntos**.

Aquí, en promedio, la recta se equivoca alrededor de 1.225 puntos.

Lectura intuitiva de la gráfica

- En $x = 1$ y $x = 3$ la recta **subestima** el valor real (residuos positivos de +0.5).
- En $x = 2$ la recta **sobreestima** el valor real (residuo −1).
- El error total (RSS=1.5) no es grande para estos tres puntos, pero el RMSD ≈ 1.225 te da una idea clara de **qué tan cerca** están los puntos, en promedio, de la recta dada.

📌 Fórmula de la pendiente

Dada una muestra de n puntos (x_i, y_i), la pendiente b_1 se calcula así:

$$b_1 = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sum(x_i - \bar{x})^2}$$

donde:

- $\bar{x}$ es el promedio de los valores de x,
- $\bar{y}$ es el promedio de los valores de y.

📌 Ejemplo con tus datos

Tus datos eran:

$$x = [1, 2, 3], \quad y = [1, 2, 6]$$

1. Promedios:

$$\bar{x} = \frac{1+2+3}{3} = 2, \quad \bar{y} = \frac{1+2+6}{3} = 3$$

2. Numerador:

$$\sum(x_i - \bar{x})(y_i - \bar{y}) = (1-2)(1-3) + (2-2)(2-3) + (3-2)(6-3) = (-1)(-2) + (0)(-1) + (1)(3) = 2 + 0 + 3 = 5$$

3. Denominador:

$$\sum(x_i - \bar{x})^2 = (1-2)^2 + (2-2)^2 + (3-2)^2 = 1 + 0 + 1 = 2$$

4. Pendiente:

$$b_1 = \frac{5}{2} = 2.5$$

📌 Intercepto