Unidad 1. Regresión lineal simple y correlación

Medidas de dispersión

Suma de los cuadrados de las desviaciones de los valores de X con respecto a su media:

$$S_{xx} = \sum_{i=1}^{n} \left(X_i - \acute{X} \right)^2$$

Suma de los productos de las desviaciones de los valores de X y Y con respecto a sus medias:

$$S_{xy} = \sum_{i=1}^{n} \left(X_i - \acute{X} \right) \left(Y_i - \acute{Y} \right)$$

Suma de los cuadrados de las desviaciones de los valores de Y con respecto a su media:

$$S_{xx} = \sum_{i=1}^{n} \left(Y_i - \acute{Y} \right)^2$$

Coeficiente de correlación y de determinación

Coeficiente de correlación de Pearson:

$$r = \frac{S_{xy}}{\sqrt{S_{xx} \cdot S_{yy}}}$$

Coeficiente de determinación:

$$r^2$$

Recta de regresión ajustada

La regresión lineal ajustada se representa mediante estadísticos:

$$\hat{Y} = b_0 + b_1 X$$

donde $\hat{Y}$ representa el valor de Y obtenido mediante la recta de regresión ajustada (no la verdadera Y). Los estadísticos b_0 y b_1 se calculan de la siguiente manera:

$$b_1 = \frac{S_{xy}}{S_{xx}}$$

$$b_0 = \acute{Y} - b_1 \acute{X}$$

Cálculo de residuales

Residuales:

$$e_i = Y_i - \hat{Y}_i$$

Sumas de cuadrados SS (Sum of Squares)

Suma de los Cuadrados de los Errores:

$$SSE = \sum_{i=1}^{n} \left(Y_i - \hat{Y}_i\right)^2 = \sum_{i=1}^{n} e_i^2$$

Suma total de cuadrados:

$$SST = \sum_{i=1}^{n} \left(Y_i - \acute{Y}\right)^2$$

Suma de cuadrados de regresión:

$$SSR = \sum_{i=1}^{n} \left(\hat{Y}_i - \acute{Y}\right)^2$$

- SST: Mide la variabilidad total de los datos observados.
- SSR: Mide la variabilidad de los datos que el modelo de regresión explica.
- SSE: Mide la variabilidad no explicada por el modelo (es decir, los residuos).

Intervalo de confianza

Estadístico de prueba t:

$$t = \frac{b_1}{SE\left(b_1\right)}$$

Error estándar de b_1:

$$SE\left(b_1\right) = \frac{\sqrt{SSE/(n-2)}}{\sqrt{S_{xx}}}$$

Intervalo de confianza para b_1:

$$b_1 - t_{\alpha/2} \cdot SE\left(b_1\right) < \beta_1 < b_1 + t_{\alpha/2} \cdot SE\left(b_1\right)$$

donde n representa la cantidad de pares de datos.

Comprobación de supuestos

Comprobar suposiciones:

- Test de shapiro a los residuales e_i: Para comprobar si la distribución es normal sobre la recta.

- Grafico X vs Y: Para observar si los datos soportan la suposición de linealidad.
- Gráfico de residuales: Para observar si los datos soportan la suposición de linealidad, complementario al coeficiente de correlación
- Test de Breusch-Pagan: Para detectar heteroscedasticidad en regresion lineal

Test de Shapiro: `from scipy.stats import shapiro` Después, se obtiene el valor-p: `_, valor_p_sh = shapiro(data)`

- H_0: Los datos siguen una distribución normal
- H_1: Los datos no siguen una distribución normal

Test de Breusch-Pagan: `from statsmodels.stats.api import het_breuschpagan` Después, se obtiene el valor-p: `_, valor_p_bp, _, _ = het_breuschpagan(residuales, X)`

- H_0: Hay homoscedasticidad
- H_1: Hay heteroscedasticidad

ANOVA en regresión lineal

Fuente de variación	Suma de cuadrados (SS)	Grados de libertad (df)	Promedio de los cuadrados (MS)	Estadístico F
Regresión	SSR	p	$MSR = \frac{SSR}{p}$	$F = \frac{MSR}{MSE}$
Error	SSE	$n - p - 1$	$MSE = \frac{SSE}{n - p - 1}$	
Total	SST	$n - 1$		

donde p es el número de parámetros para la recta de regresión ajustada (en la regresión simple $p=1$). Las hipótesis son:

$$H_0 : \beta_1 = 0$$

$$H_1 : \beta_1 \neq 0$$

Problemario de la Unidad 1

Problema 1

Un profesor intenta mostrar a sus estudiantes la importancia de los exámenes cortos, aun cuando el 90% de la calificación final esté determinada por los exámenes parciales. Él cree que cuanto más altas sean las calificaciones de los exámenes cortos, más alta será la calificación final. Seleccionó una muestra aleatoria de 15 estudiantes de su clase con los siguientes datos:

Promedio de exámenes cortos	Promedio final
59	64
92	84
72	77

Promedio de exámenes cortos	Promedio final
90	80
95	77
87	81
89	80
77	84
76	80
65	69
97	83
42	40
94	78
62	65
91	90

1. Establesca una variable dependiente (Y) y una variable independiente (X).
2. Realice un diagrama de dispersión para estos datos.
3. ¿Los datos soportan la suposición de linealidad?
4. Calcule el coeficiente de correlación e interprete el resultado.
5. Calcule el coeficiente de determinación e interprete el resultado.
6. Obtenga la recta de regresión ajustada y grafíquelo sobre el gráfico de dispersión.
7. Obtenga un intervalo de confianza del 95% para la pendiente de la recta de regresión ajustada (b_1)
8. Calcule los residuales y trace un nuevo gráfico de dispersión. Comente, ¿Parece que se verifican los supuestos?
9. Realice la prueba de Shapiro para los residuales y comente el resultado.
10. Realice la prueba de Brausch-Pagan para los residuales y comente el resultado.
11. Tres estudiantes sacaron 70, 75 y 84 de calificación. Según la recta de regresión ajustada, ¿cuáles son los resultados esperados para estos tres alumnos?
12. Realice una tabla ANOVA e interprete el resultado.

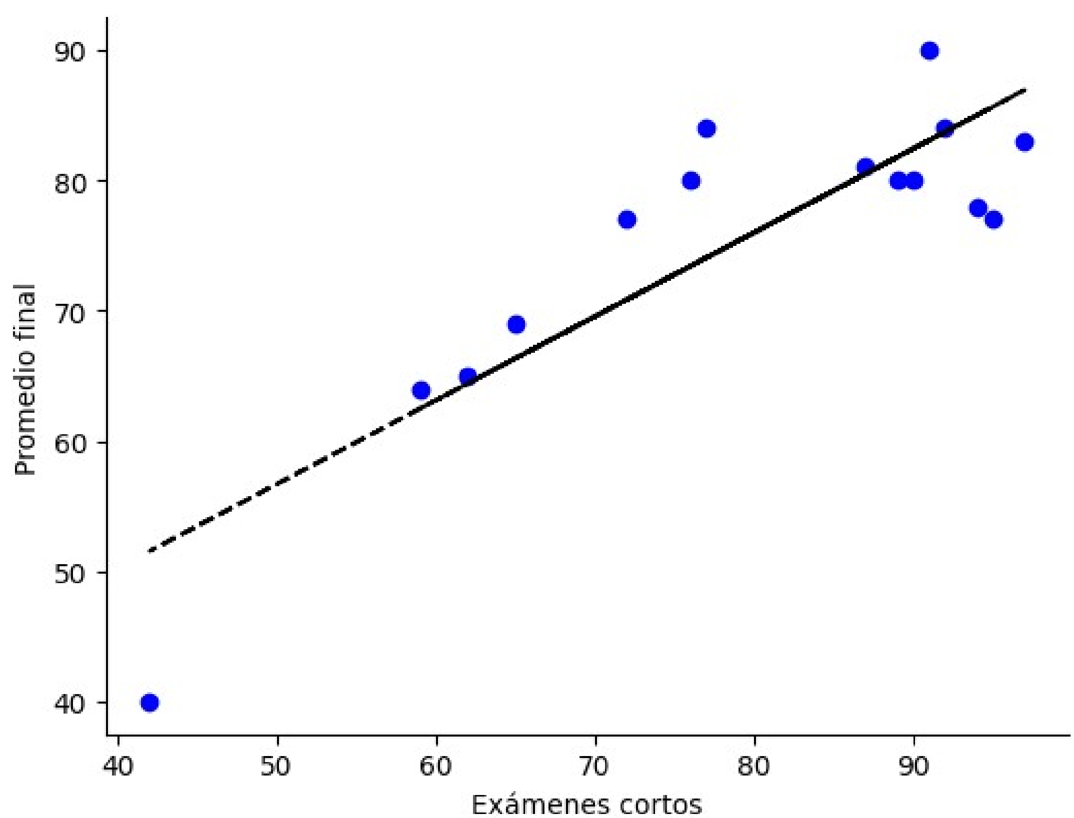
```python
import pandas as pd
import matplotlib.pyplot as plt
df = pd.DataFrame({
    'Examenes_cortos': [59,92,72,90,95,87,89,77,76,65,97,42,94,62,91],
    'Promedio_final': [64,84,77,80,77,81,80,84,80,69,83,40,78,65,90]})
df.head()
```

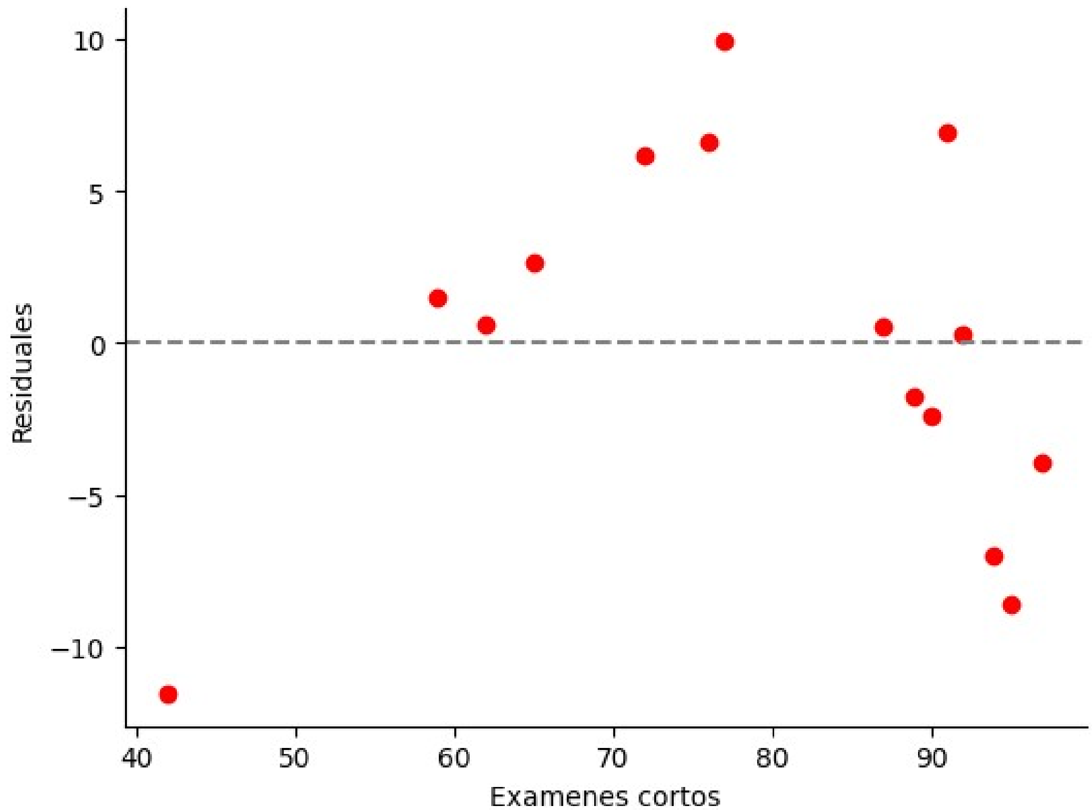
	Examenes_cortos	Promedio_final
0	59	64
1	92	84
2	72	77
3	90	80
4	95	77

```python
# 1. Establesca una variable dependiente ( Y ) y una variable
independiente ( X ).
X = df['Examenes_cortos']
Y = df['Promedio_final']

# 2. Realice un diagrama de dispersión para estos datos.
plt.scatter(X, Y, color = 'blue')
plt.xlabel('Exámenes cortos')
plt.ylabel('Promedio final')
ax = plt.gca()
ax.spines['top'].set_visible(False)
ax.spines['right'].set_visible(False)

# 3. ¿Los datos soportan la suposición de linealidad?
# Sí

# 4. Calcule el coeficiente de correlación e interprete el resultado.
from scipy.stats import pearsonr
r, _ = pearsonr(X, Y)
print(f'Coeficiente de correlación: {r: 0.4f}\n')

# 5. Calcule el coeficiente de determinación e interprete el
resultado.
print(f'Coeficiente de determinación: {r ** 2: 0.4f}\n')


# 6. Obtenga la recta de regresión ajustada y grafíquelo sobre el
gráfico de
# dispersión.
import statsmodels.api as sm
x_constante = sm.add_constant(X)
modelo = sm.OLS(Y, x_constante).fit()

b0, b1 = modelo.params

fun = lambda x: b0 + b1 * x

Yc = fun(X)

plt.plot(X, Yc, color = 'black', linestyle = '--')

from sklearn.metrics import r2_score  # recomendada
r2 = r2_score(Y, Yc)
print(f'Coeficiente de determinación: {r2: 0.4f}\n')

# 7. Obtenga un intervalo de confianza del 95% para la pendiente de la
recta de
# regresión ajustada ( b1 )
nivel_de_confianza = 0.95
```

```python
intervalo_de_confianza = modelo.conf_int(alpha = 1 -
nivel_de_confianza)
intervalo_de_confianza_b1 = intervalo_de_confianza.iloc[1]
print(f'Intervalo de confianza para b1 de {nivel_de_confianza: 0.0%}')
print(f'{intervalo_de_confianza_b1[0]: 0.4f} < b1 <
{intervalo_de_confianza_b1[1]: 0.4f}\n')


# 8. Calcule los residuales y trace un nuevo gráfico de dispersión.
Comente,
# ¿Parece que se verifican los supuestos?
residuales = modelo.resid
plt.figure()
plt.scatter(X, residuales, color = 'red')
plt.xlabel('Examenes cortos')
plt.ylabel('Residuales')
ax = plt.gca()
ax.spines['top'].set_visible(False)
ax.spines['right'].set_visible(False)
plt.axhline(y = 0, color = 'gray', linestyle = '--')

# 9. Realice la prueba de Shapiro para los residuales y comente el
resultado.
from scipy.stats import shapiro
_, valor_p_sh = shapiro(residuales)
print(f'valor-p de Shapiro: {valor_p_sh: 0.4f}\n')


# 10. Realice la prueba de Brausch-Pagan para los residuales y comente
el
# resultado.
from statsmodels.stats.api import het_breuschpagan
_, valor_p_bp, _, _ = het_breuschpagan(residuales, x_constante)
print(f'valor_p de Breusch-Pagan: {valor_p_bp: 0.4f}\n')

# 11. Tres estudiantes sacaron 70, 75 y 84 de calificación. Según la
recta de
# regresión ajustada, ¿cuáles son los resultados esperados para estos
tres alumnos?
print(f'para x = 70, y = {fun(70): 0.0f}')
print(f'para x = 75, y = {fun(75): 0.0f}')
print(f'para x = 84, y = {fun(84): 0.0f}\n')

# Realice una tabla ANOVA e interprete el resultado.
from statsmodels.formula.api import ols
# Y ~ X
modelo_2 = ols('Promedio_final ~ Examenes_cortos', data = df).fit()
tabla_anova = sm.stats.anova_lm(modelo_2)
tabla_anova
```

```
Coeficiente de correlación:  0.8646

Coeficiente de determinación:  0.7475

Coeficiente de determinación:  0.7475

Intervalo de confianza para b1 de  95%
 0.4192 < b1 <  0.8671

valor-p de Shapiro:  0.9018

valor_p de Breusch-Pagan:  0.2289

para x = 70, y =  70
para x = 75, y =  73
para x = 84, y =  79


                   df       sum_sq     mean_sq          F    PR(>F)
Examenes_cortos   1.0  1538.228959 1538.228959  38.492412  0.000032
Residual         13.0   519.504375   39.961875        NaN       NaN
```





Problema 2

William Hawkins, vicepresidente de personal de la International Motors, trabaja en la relación entre el salario de un trabajador y el porcentaje de ausentismo. Hawkins dividió el intervalo de salarios de International en 12 grados o niveles (1 es el menor grado, 12 el más alto) y después muestreó aleatoriamente a un grupo de trabajadores. Determinó el grado de salario de cada trabajador y el número de días que ese empleado había faltado en los últimos 3 años.

Categoría de salario	11	10	8	5	9	7	3
Ausencias	18	17	29	36	11	28	35

Categoría de salario	11	8	7	2	9	8	3
Ausencias	14	20	32	39	16	31	40

1. Establesca una variable dependiente (Y) y una variable independiente (X).
2. Realice un diagrama de dispersión para estos datos.
3. ¿Los datos soportan la suposición de linealidad?

4. Calcule el coeficiente de correlación e interprete el resultado.
5. Calcule el coeficiente de determinación e interprete el resultado.
6. Obtenga la recta de regresión ajustada y grafíquelo sobre el gráfico de dispersión.
7. Obtenga un intervalo de confianza del 95% para la pendiente de la recta de regresión ajustada (b_1)
8. Calcule los residuales y trace un nuevo gráfico de dispersión. Comente, ¿Parece que se verifican los supuestos?
9. Realice la prueba de Shapiro para los residuales y comente el resultado.
10. Realice la prueba de Brausch-Pagan para los residuales y comente el resultado.
11. Utiliza la recta de regresión para interpolar dos valores y extrapolar uno. Comenta estos resultados.
12. Realice una tabla ANOVA e interprete el resultado.

```
import pandas as pd
import matplotlib.pyplot as plt
df = pd.DataFrame({
    'Categoria_de_salario': [11, 10, 8, 5, 9, 7, 3, 11, 8, 7, 2, 9, 8,
3],
    'Ausencias': [18, 17, 29, 36, 11, 28, 35, 14, 20, 32, 39, 16, 31,
40]})
df.head()
```

	Categoria_de_salario	Ausencias
0	11	18
1	10	17
2	8	29
3	5	36
4	9	11

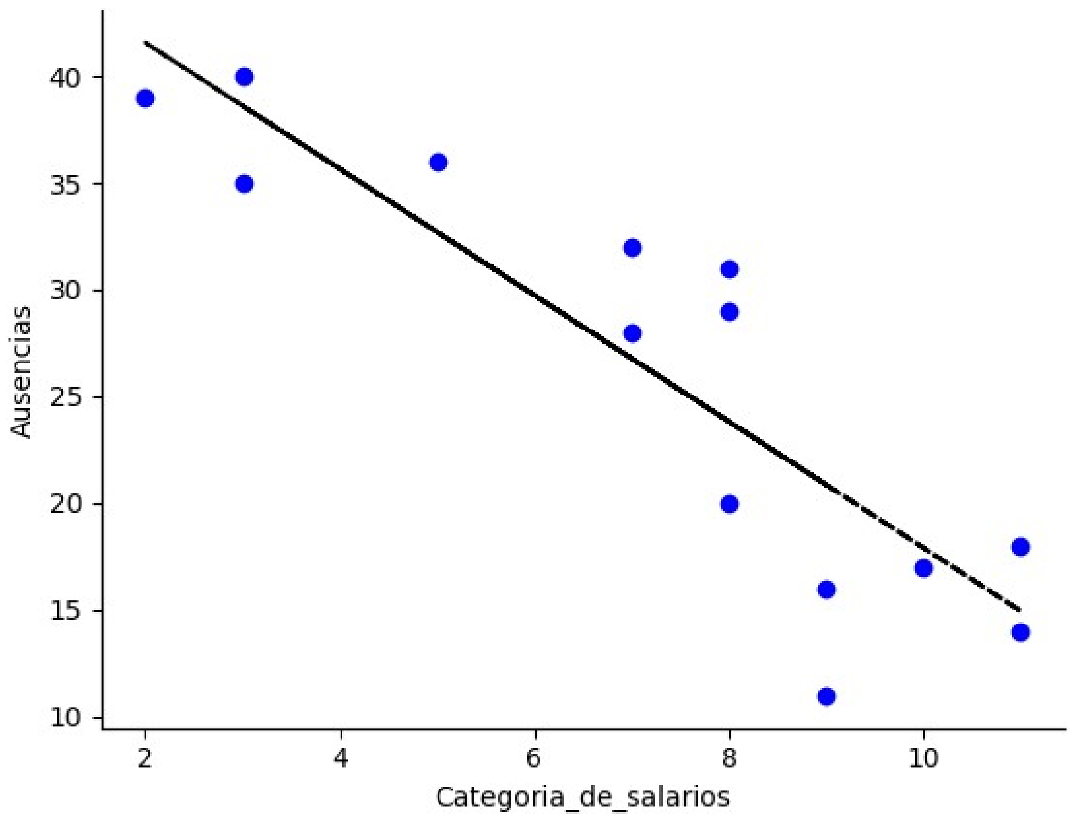
```
# 1. Establesca una variable dependiente ( Y ) y una variable
independiente ( X ).
X = df['Categoria_de_salario']
Y = df['Ausencias']

# 2. Realice un diagrama de dispersión para estos datos.
plt.scatter(X, Y, color = 'blue')
plt.xlabel('Categoria_de_salarios')
plt.ylabel('Ausencias')
ax = plt.gca()
ax.spines['top'].set_visible(False)
ax.spines['right'].set_visible(False)

# 3. ¿Los datos soportan la suposición de linealidad?
# Sí

# 4. Calcule el coeficiente de correlación e interprete el resultado.
from scipy.stats import pearsonr
r, _ = pearsonr(X, Y)
print(f'Coeficiente de correlación: {r: 0.4f}\n')
```

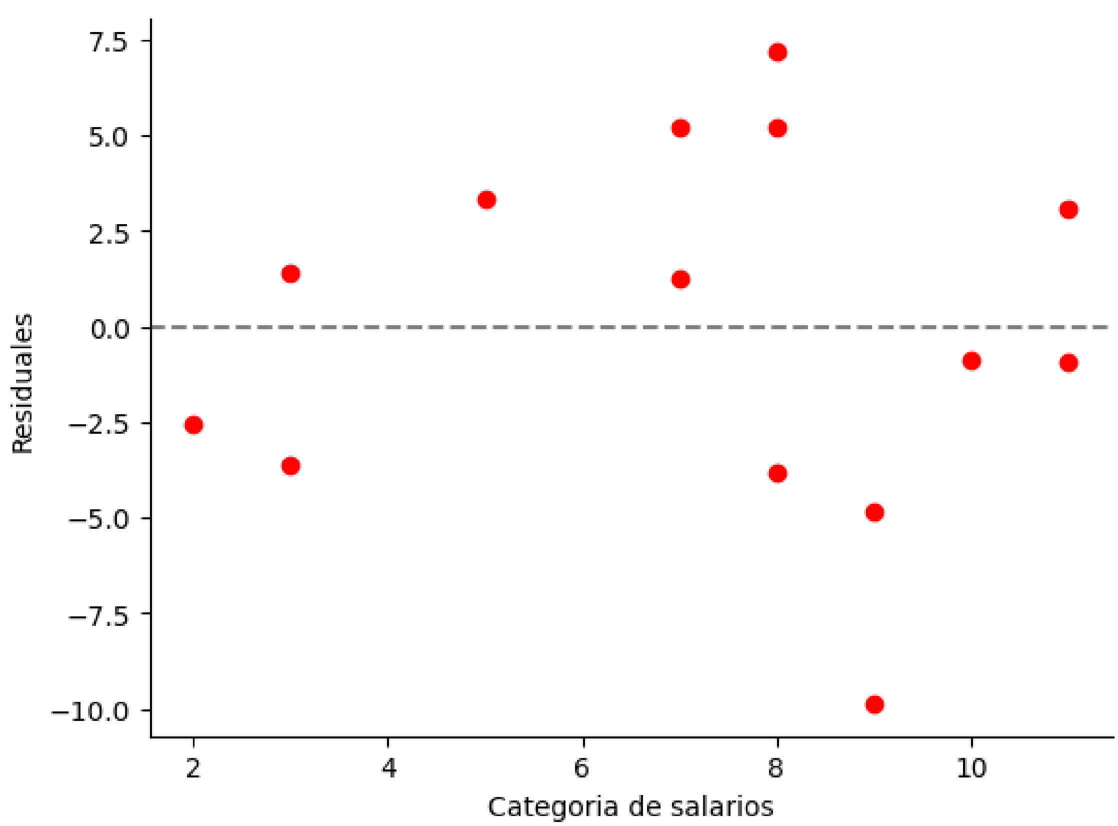
```
# 5. Calcule el coeficiente de determinación e interprete el
resultado.
print(f'Coeficiente de determinación: {r ** 2: 0.4f}\n')


# 6. Obtenga la recta de regresión ajustada y grafíquelo sobre el
gráfico de
# dispersión.
import statsmodels.api as sm
x_constante = sm.add_constant(X)
modelo = sm.OLS(Y, x_constante).fit()

b0, b1 = modelo.params

fun = lambda x: b0 + b1 * x

Yc = fun(X)

plt.plot(X, Yc, color = 'black', linestyle = '--')

from sklearn.metrics import r2_score  # recomendada
r2 = r2_score(Y, Yc)
print(f'Coeficiente de determinación: {r2: 0.4f}\n')

# 7. Obtenga un intervalo de confianza del 95% para la pendiente de la
recta de
# regresión ajustada ( b1 )
nivel_de_confianza = 0.95
intervalo_de_confianza = modelo.conf_int(alpha = 1 -
nivel_de_confianza)
intervalo_de_confianza_b1 = intervalo_de_confianza.iloc[1]
print(f'Intervalo de confianza para b1 de {nivel_de_confianza: 0.0%}')
print(f'{intervalo_de_confianza_b1[0]: 0.4f} < b1 <
{intervalo_de_confianza_b1[1]: 0.4f}\n')


# 8. Calcule los residuales y trace un nuevo gráfico de dispersión.
Comente,
# ¿Parece que se verifican los supuestos?
residuales = modelo.resid
plt.figure()
plt.scatter(X, residuales, color = 'red')
plt.xlabel('Categoria de salarios')
plt.ylabel('Residuales')
ax = plt.gca()
ax.spines['top'].set_visible(False)
ax.spines['right'].set_visible(False)
plt.axhline(y = 0, color = 'gray', linestyle = '--')
```

```python
# 9. Realice la prueba de Shapiro para los residuales y comente el
resultado.
from scipy.stats import shapiro
_, valor_p_sh = shapiro(residuales)
print(f'valor-p de Shapiro: {valor_p_sh: 0.4f}\n')


# 10. Realice la prueba de Brausch-Pagan para los residuales y comente
el
# resultado.
from statsmodels.stats.api import het_breuschpagan
_, valor_p_bp, _, _ = het_breuschpagan(residuales, x_constante)
print(f'valor_p de Breusch-Pagan: {valor_p_bp: 0.4f}\n')

# Realice una tabla ANOVA e interprete el resultado.
from statsmodels.formula.api import ols
# Y ~ X
modelo_2 = ols('Ausencias ~ Categoria_de_salario', data = df).fit()
tabla_anova = sm.stats.anova_lm(modelo_2)
tabla_anova
```

```
Coeficiente de correlación: -0.8801

Coeficiente de determinación:  0.7746

Coeficiente de determinación:  0.7746

Intervalo de confianza para b1 de  95%
-3.9625 < b1 < -1.9549

valor-p de Shapiro:  0.8933

valor_p de Breusch-Pagan:  0.4505
```

	df	sum_sq	mean_sq	F	PR(>F)
Categoria_de_salario	1.0	983.548996	983.548996	41.243954	0.000033
Residual	12.0	286.165289	23.847107	NaN	NaN


Ausencias
Categoria_de_salarios
40
35
30
25
20
15
10
2
4
6
8
10


Residuales
Categoria de salarios
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
2
4
6
8
10

Problema 3

A menudo, quienes hacen la contabilidad de costos estiman los gastos generales con base en el nivel de producción. En Standard Knitting Co. han reunido información acerca de los gastos generales y las unidades producidas en diferentes plantas.

Gastos generales	191	170	272	155	280	173	234	116	153	178
Unidades	40	42	53	35	56	39	48	30	37	40

1. Establesca una variable dependiente (Y) y una variable independiente (X).
2. Realice un diagrama de dispersión para estos datos.
3. ¿Los datos soportan la suposición de linealidad?
4. Calcule el coeficiente de correlación e interprete el resultado.
5. Calcule el coeficiente de determinación e interprete el resultado.
6. Obtenga la recta de regresión ajustada y grafíquelo sobre el gráfico de dispersión.
7. Obtenga un intervalo de confianza del 95% para la pendiente de la recta de regresión ajustada (b_1)
8. Calcule los residuales y trace un nuevo gráfico de dispersión. Comente, ¿Parece que se verifican los supuestos?
9. Realice la prueba de Shapiro para los residuales y comente el resultado.
10. Realice la prueba de Brausch-Pagan para los residuales y comente el resultado.
11. Utiliza la recta de regresión para interpolar dos valores y extrapolar uno. Comenta estos resultados.
12. Realice una tabla ANOVA e interprete el resultado.

```python
import pandas as pd
import matplotlib.pyplot as plt
df = pd.DataFrame({
    'Gastos_generales': [191, 170, 272, 155, 280, 173, 234, 116, 153, 178],
    'Unidades': [40, 42, 53, 35, 56, 39, 48, 30, 37, 40]})
df.head()
```

	Gastos_generales	Unidades
0	191	40
1	170	42
2	272	53
3	155	35
4	280	56

```python
# 1. Establesca una variable dependiente ( Y ) y una variable independiente ( X ).
X = df['Gastos_generales']
Y = df['Unidades']

# 2. Realice un diagrama de dispersión para estos datos.
plt.scatter(X, Y, color = 'blue')
plt.xlabel('Gastos_Generales')
plt.ylabel('Unidades')
```

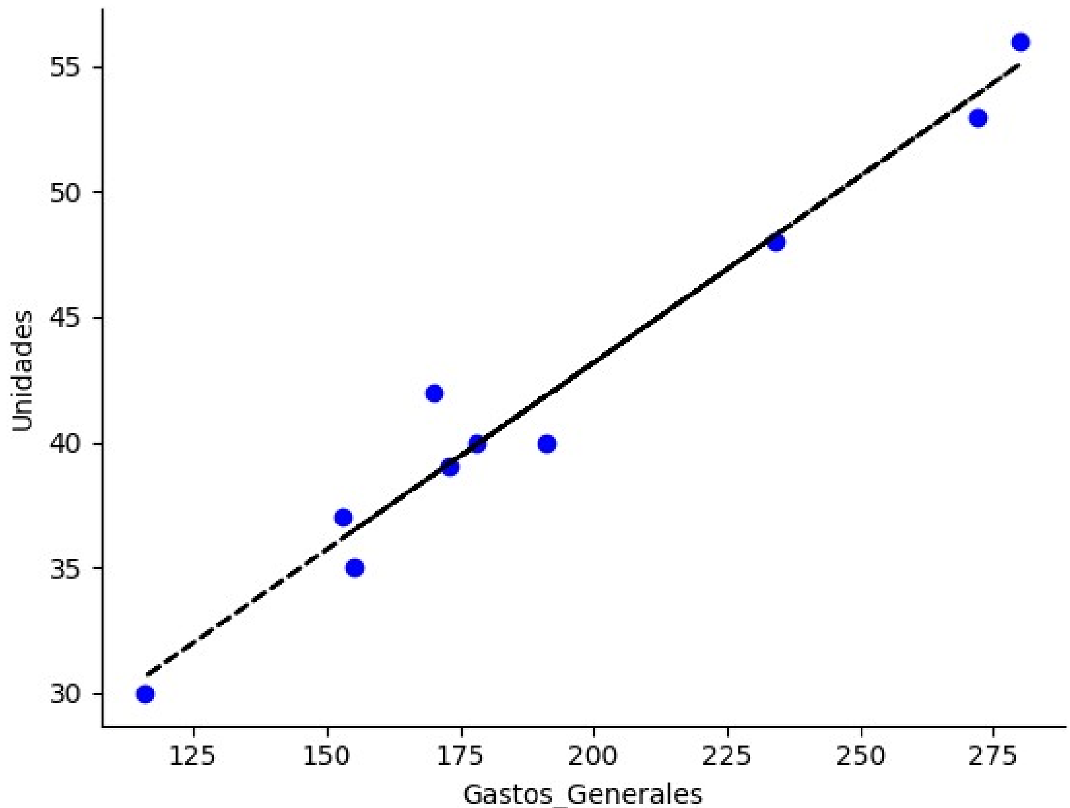
```
ax = plt.gca()
ax.spines['top'].set_visible(False)
ax.spines['right'].set_visible(False)

# 3. ¿Los datos soportan la suposición de linealidad?
# Sí

# 4. Calcule el coeficiente de correlación e interprete el resultado.
from scipy.stats import pearsonr
r, _ = pearsonr(X, Y)
print(f'Coeficiente de correlación: {r: 0.4f}\n')

# 5. Calcule el coeficiente de determinación e interprete el
resultado.
print(f'Coeficiente de determinación: {r ** 2: 0.4f}\n')


# 6. Obtenga la recta de regresión ajustada y grafíquelo sobre el
gráfico de
# dispersión.
import statsmodels.api as sm
x_constante = sm.add_constant(X)
modelo = sm.OLS(Y, x_constante).fit()

b0, b1 = modelo.params

fun = lambda x: b0 + b1 * x

Yc = fun(X)

plt.plot(X, Yc, color = 'black', linestyle = '--')

from sklearn.metrics import r2_score  # recomendada
r2 = r2_score(Y, Yc)
print(f'Coeficiente de determinación: {r2: 0.4f}\n')

# 7. Obtenga un intervalo de confianza del 95% para la pendiente de la
recta de
# regresión ajustada ( b1 )
nivel_de_confianza = 0.95
intervalo_de_confianza = modelo.conf_int(alpha = 1 -
nivel_de_confianza)
intervalo_de_confianza_b1 = intervalo_de_confianza.iloc[1]
print(f'Intervalo de confianza para b1 de {nivel_de_confianza: 0.0%}')
print(f'{intervalo_de_confianza_b1[0]: 0.4f} < b1 <
{intervalo_de_confianza_b1[1]: 0.4f}\n')


# 8. Calcule los residuales y trace un nuevo gráfico de dispersión.
Comente,
```

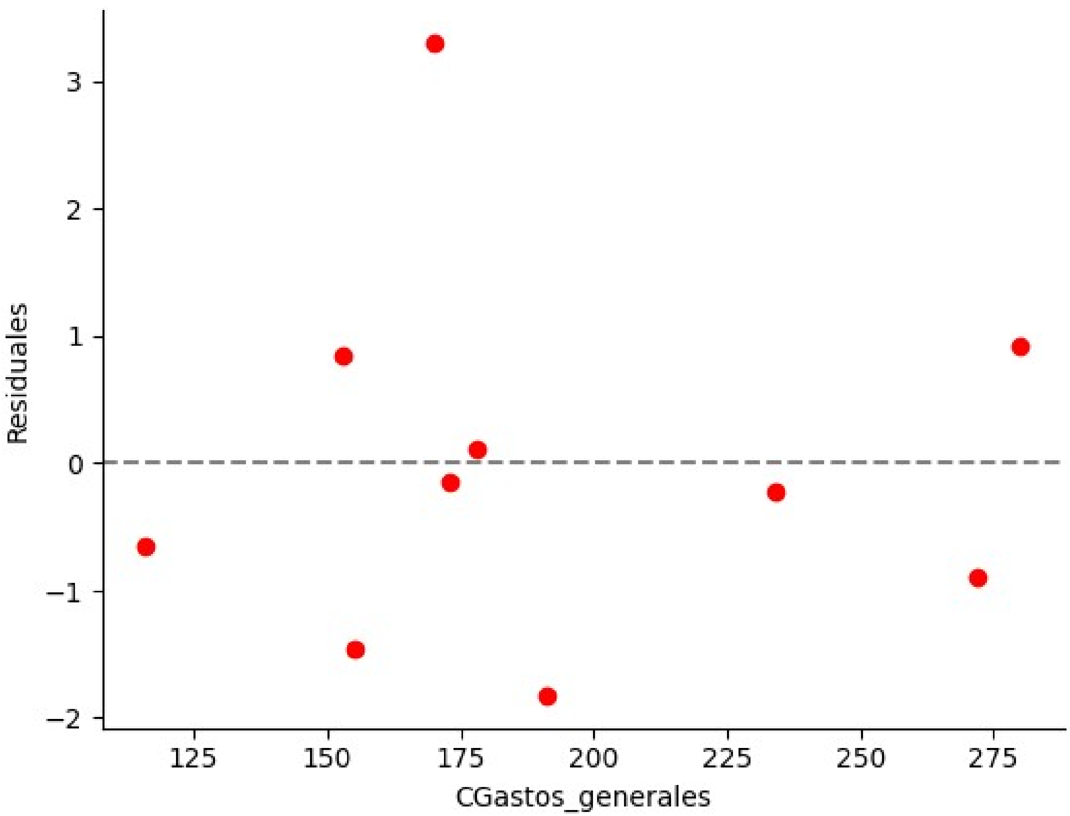
```
# ¿Parece que se verifican los supuestos?
residuales = modelo.resid
plt.figure()
plt.scatter(X, residuales, color = 'red')
plt.xlabel('CGastos_generales')
plt.ylabel('Residuales')
ax = plt.gca()
ax.spines['top'].set_visible(False)
ax.spines['right'].set_visible(False)
plt.axhline(y = 0, color = 'gray', linestyle = '--')

# 9. Realice la prueba de Shapiro para los residuales y comente el
resultado.
from scipy.stats import shapiro
_, valor_p_sh = shapiro(residuales)
print(f'valor-p de Shapiro: {valor_p_sh: 0.4f}\n')


# 10. Realice la prueba de Brausch-Pagan para los residuales y comente
el
# resultado.
from statsmodels.stats.api import het_breuschpagan
_, valor_p_bp, _, _ = het_breuschpagan(residuales, x_constante)
print(f'valor_p de Breusch-Pagan: {valor_p_bp: 0.4f}\n')

# Realice una tabla ANOVA e interprete el resultado.
from statsmodels.formula.api import ols
# Y ~ X
modelo_2 = ols('Unidades ~ Gastos_generales', data = df).fit()
tabla_anova = sm.stats.anova_lm(modelo_2)
tabla_anova
```

```
Coeficiente de correlación:  0.9835

Coeficiente de determinación:  0.9673

Coeficiente de determinación:  0.9673

Intervalo de confianza para b1 de  95%
 0.1267 < b1 <  0.1713

valor-p de Shapiro:  0.3096

valor_p de Breusch-Pagan:  0.6267
```

	df	sum_sq	mean_sq	F	PR(>F)
Gastos_generales	1.0	568.774067	568.774067	236.669535	3.167080e-07

```
Residual           8.0   19.225933    2.403242          NaN
NaN
```





Problema 4

Las ventas de línea blanca varían según el estado del mercado de casas nuevas: cuando las ventas de casas nuevas son buenas, también lo son las de lavaplatos, lavadoras de ropa, secadoras y refrigeradores. Una asociación de comercio compiló los siguientes datos históricos (en miles de unidades) de las ventas de línea blanca y la construcción de casas.

Construcción de casas (miles)	Ventas de línea blanca (miles)
2.0	5.0
2.5	5.5
3.2	6.0
3.6	7.0
3.7	7.2
4.0	7.7
4.2	8.4
4.6	9.0
4.8	9.7
5.0	10.0

1. Establesca una variable dependiente (Y) y una variable independiente (X).
2. Realice un diagrama de dispersión para estos datos.
3. ¿Los datos soportan la suposición de linealidad?

4. Calcule el coeficiente de correlación e interprete el resultado.
5. Calcule el coeficiente de determinación e interprete el resultado.
6. Obtenga la recta de regresión ajustada y grafíquelo sobre el gráfico de dispersión.
7. Obtenga un intervalo de confianza del 95% para la pendiente de la recta de regresión ajustada (b_1)
8. Calcule los residuales y trace un nuevo gráfico de dispersión. Comente, ¿Parece que se verifican los supuestos?
9. Realice la prueba de Shapiro para los residuales y comente el resultado.
10. Realice la prueba de Brausch-Pagan para los residuales y comente el resultado.
11. Utiliza la recta de regresión para interpolar dos valores y extrapolar uno. Comenta estos resultados.
12. Realice una tabla ANOVA e interprete el resultado.

```
import pandas as pd
import matplotlib.pyplot as plt
df = pd.DataFrame({
    'Construccion': [2.0, 2.5, 3.2, 3.6, 3.7, 4.0, 4.2, 4.6, 4.8, 
5.0],
    'Ventas': [5.0, 5.5, 6.0, 7.0, 7.2, 7.7, 8.4, 9.0, 9.7, 10.0]})
df.head()
```

	Construccion	Ventas
0	2.0	5.0
1	2.5	5.5
2	3.2	6.0
3	3.6	7.0
4	3.7	7.2

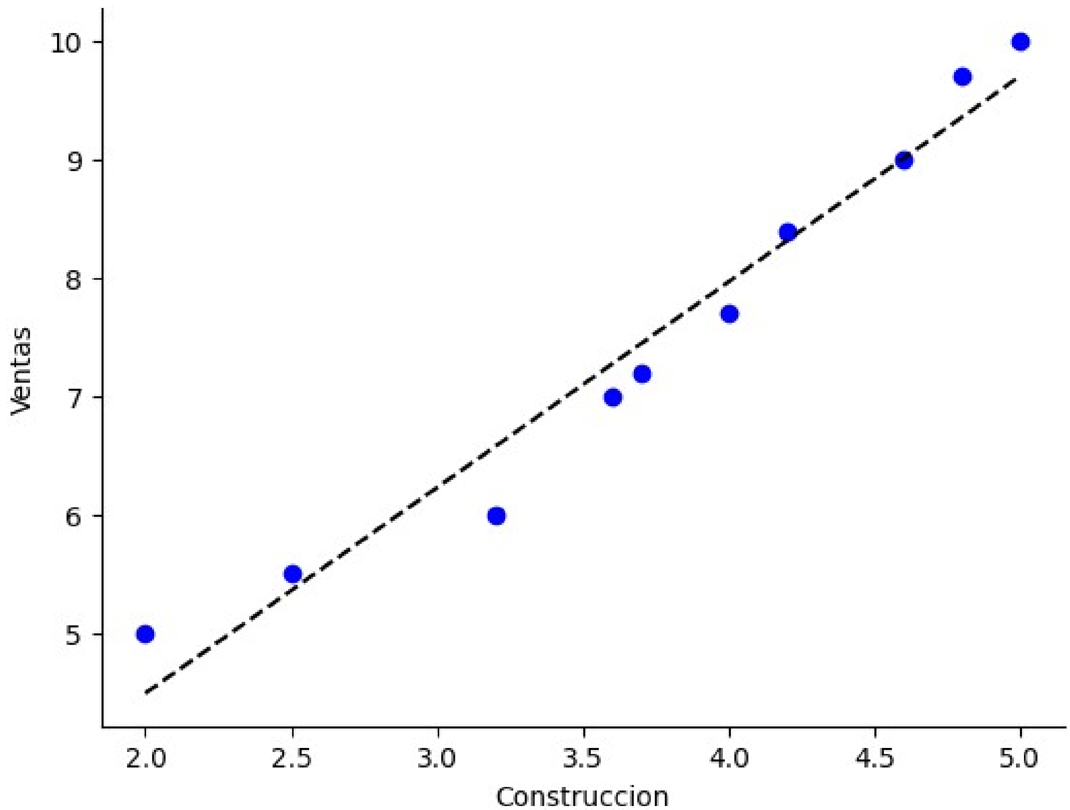
```
# 1. Establesca una variable dependiente ( Y ) y una variable 
independiente ( X ).
X = df['Construccion']
Y = df['Ventas']

# 2. Realice un diagrama de dispersión para estos datos.
plt.scatter(X, Y, color = 'blue')
plt.xlabel('Construccion')
plt.ylabel('Ventas')
ax = plt.gca()
ax.spines['top'].set_visible(False)
ax.spines['right'].set_visible(False)

# 3. ¿Los datos soportan la suposición de linealidad?
# Sí

# 4. Calcule el coeficiente de correlación e interprete el resultado.
from scipy.stats import pearsonr
r, _ = pearsonr(X, Y)
print(f'Coeficiente de correlación: {r: 0.4f}\n')
```

```python
# 5. Calcule el coeficiente de determinación e interprete el
resultado.
print(f'Coeficiente de determinación: {r ** 2: 0.4f}\n')


# 6. Obtenga la recta de regresión ajustada y grafíquelo sobre el
gráfico de
# dispersión.
import statsmodels.api as sm
x_constante = sm.add_constant(X)
modelo = sm.OLS(Y, x_constante).fit()

b0, b1 = modelo.params

fun = lambda x: b0 + b1 * x

Yc = fun(X)

plt.plot(X, Yc, color = 'black', linestyle = '--')

from sklearn.metrics import r2_score  # recomendada
r2 = r2_score(Y, Yc)
print(f'Coeficiente de determinación: {r2: 0.4f}\n')

# 7. Obtenga un intervalo de confianza del 95% para la pendiente de la
recta de
# regresión ajustada ( b1 )
nivel_de_confianza = 0.95
intervalo_de_confianza = modelo.conf_int(alpha = 1 -
nivel_de_confianza)
intervalo_de_confianza_b1 = intervalo_de_confianza.iloc[1]
print(f'Intervalo de confianza para b1 de {nivel_de_confianza: 0.0%}')
print(f'{intervalo_de_confianza_b1[0]: 0.4f} < b1 <
{intervalo_de_confianza_b1[1]: 0.4f}\n')


# 8. Calcule los residuales y trace un nuevo gráfico de dispersión.
Comente,
# ¿Parece que se verifican los supuestos?
residuales = modelo.resid
plt.figure()
plt.scatter(X, residuales, color = 'red')
plt.xlabel('Construccion')
plt.ylabel('Residuales')
ax = plt.gca()
ax.spines['top'].set_visible(False)
ax.spines['right'].set_visible(False)
plt.axhline(y = 0, color = 'gray', linestyle = '--')

# 9. Realice la prueba de Shapiro para los residuales y comente el
```

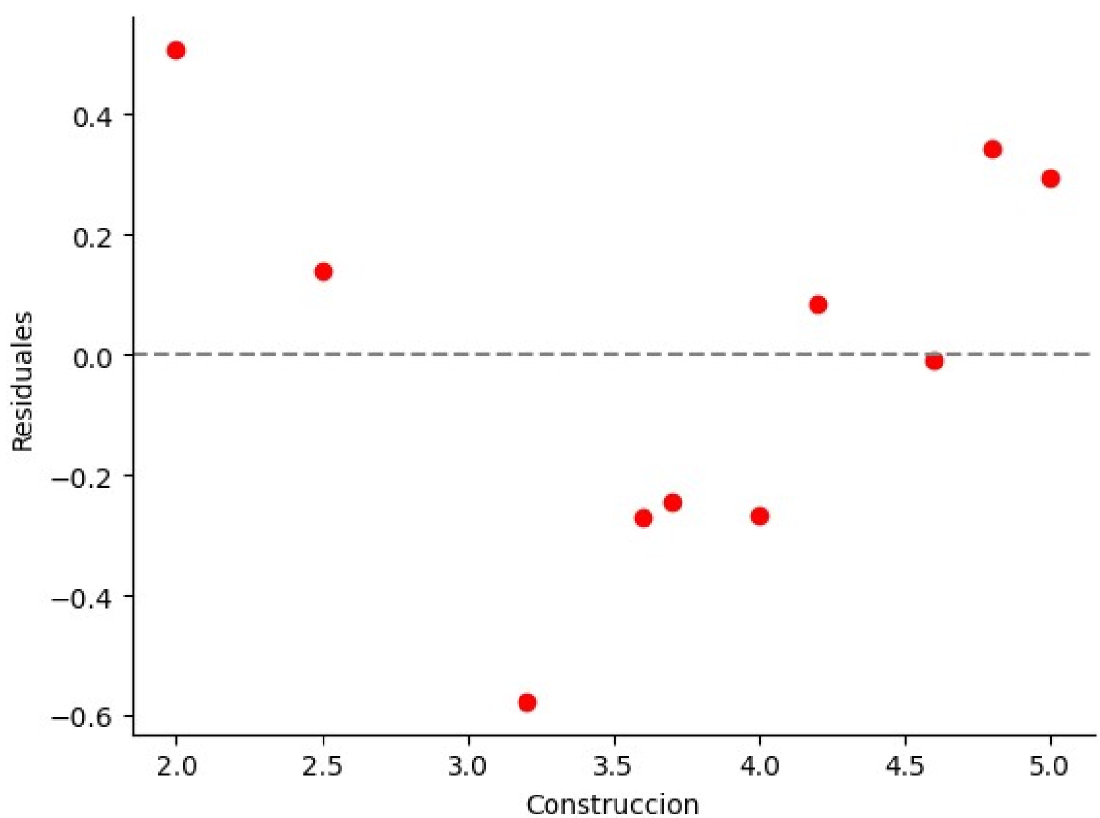
```
resultado.
from scipy.stats import shapiro
_, valor_p_sh = shapiro(residuales)
print(f'valor-p de Shapiro: {valor_p_sh: 0.4f}\n')


# 10. Realice la prueba de Brausch-Pagan para los residuales y comente
el
# resultado.
from statsmodels.stats.api import het_breuschpagan
_, valor_p_bp, _, _ = het_breuschpagan(residuales, x_constante)
print(f'valor_p de Breusch-Pagan: {valor_p_bp: 0.4f}\n')



#Realice una tabla ANOVA e interprete el resultado.
from statsmodels.formula.api import ols
# Y ~ X
modelo_2 = ols('Ventas ~ Construccion', data = df).fit()
tabla_anova = sm.stats.anova_lm(modelo_2)
tabla_anova
```

```
Coeficiente de correlación:  0.9808

Coeficiente de determinación:  0.9619

Coeficiente de determinación:  0.9619

Intervalo de confianza para b1 de  95%
 1.4557 < b1 <  2.0194

valor-p de Shapiro:  0.8464

valor_p de Breusch-Pagan:  0.1581
```

	df	sum_sq	mean_sq	F	PR(>F)
Construccion	1.0	25.976581	25.976581	202.069951	5.841003e-07
Residual	8.0	1.028419	0.128552	NaN	NaN



Ventas
Construccion



Residuales
Construccion

Problema 5

William C. Andrews, consultor de comportamiento organizacional de Victory Motorcycles, ha diseñado una prueba para mostrar a los supervisores de la compañía los peligros de sobrevigilar a sus trabajadores. Un trabajador de la línea de ensamble tiene a su cargo una serie de tareas complicadas. Durante el desempeño del trabajador, un inspector lo interrumpe constantemente para ayudarlo a terminar las tareas. El trabajador, después de terminar su trabajo, recibe una prueba psicológica diseñada para medir la hostilidad del trabajador hacia la autoridad (una alta puntuación implica una hostilidad baja). A ocho distintos trabajadores se les asignaron las tareas y luego se les interrumpió para darles instrucciones útiles un número variable de veces (línea X). Sus calificaciones en la prueba de hostilidad se dan en el renglón Y.

número interrupciones al trabajador	5	10	10	15	15	20	20	25
calificación del trabajador en la prueba de hostilidad	58	41	45	27	26	12	16	3

1. Establesca una variable dependiente (Y) y una variable independiente (X).
2. Realice un diagrama de dispersión para estos datos.
3. ¿Los datos soportan la suposición de linealidad?
4. Calcule el coeficiente de correlación e interprete el resultado.
5. Calcule el coeficiente de determinación e interprete el resultado.
6. Obtenga la recta de regresión ajustada y grafíquelo sobre el gráfico de dispersión.
7. Obtenga un intervalo de confianza del 95% para la pendiente de la recta de regresión ajustada (b_1)
8. Calcule los residuales y trace un nuevo gráfico de dispersión. Comente, ¿Parece que se verifican los supuestos?
9. Realice la prueba de Shapiro para los residuales y comente el resultado.
10. Realice la prueba de Brausch-Pagan para los residuales y comente el resultado.
11. Utiliza la recta de regresión para interpolar dos valores y extrapolar uno. Comenta estos resultados.
12. Realice una tabla ANOVA e interprete el resultado.

```python
import pandas as pd
import matplotlib.pyplot as plt
df = pd.DataFrame({
    'Interrupciones': [5, 10, 10, 15, 15, 20, 20, 25],
    'Calificacion': [58, 41, 45, 27, 26, 12, 16, 3]})
df.head()
```

	Interrupciones	Calificacion
0	5	58
1	10	41
2	10	45
3	15	27
4	15	26

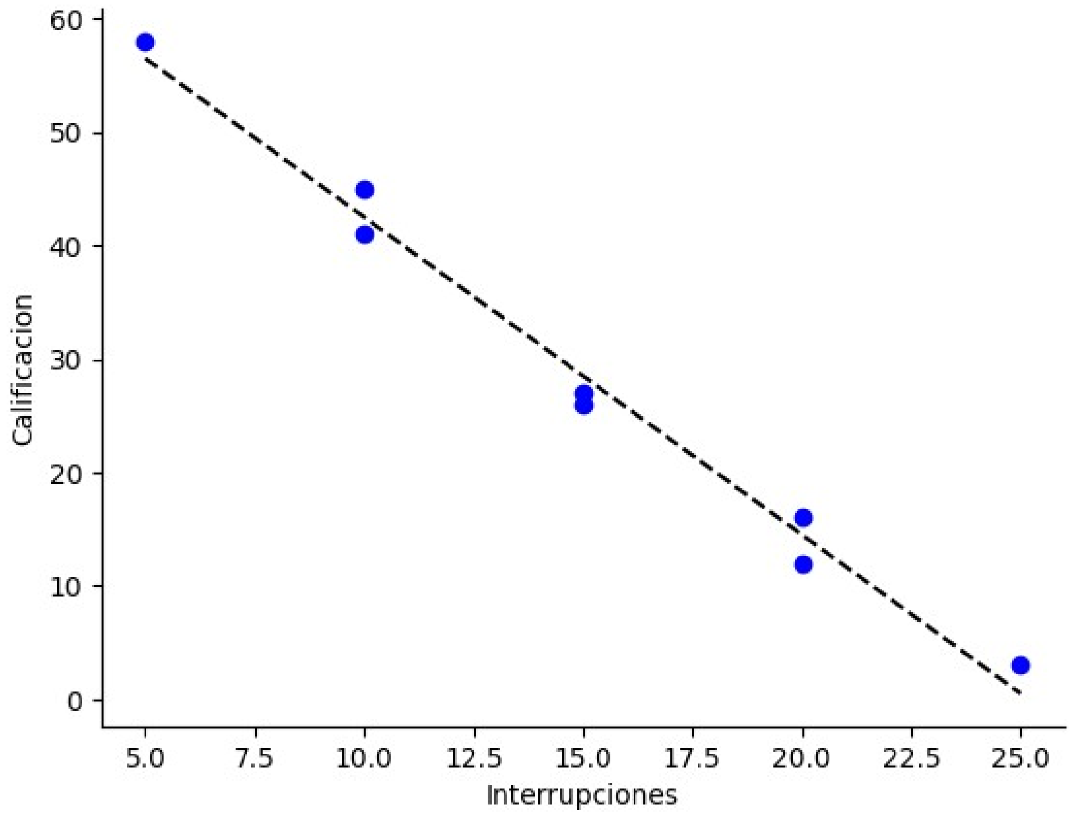
```python
# 1. Establesca una variable dependiente ( Y ) y una variable
independiente ( X ).
X = df['Interrupciones']
Y = df['Calificacion']
```

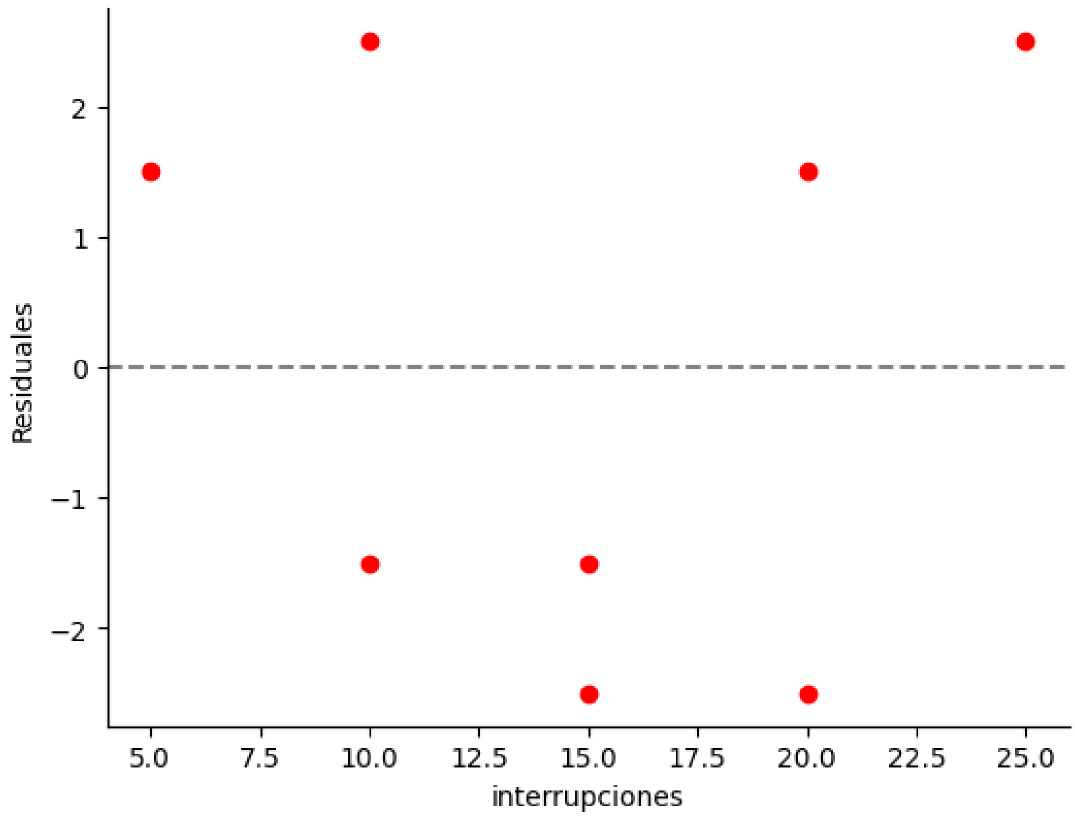
```python
# 2. Realice un diagrama de dispersión para estos datos.
plt.scatter(X, Y, color = 'blue')
plt.xlabel('Interrupciones')
plt.ylabel('Calificacion')
ax = plt.gca()
ax.spines['top'].set_visible(False)
ax.spines['right'].set_visible(False)

# 3. ¿Los datos soportan la suposición de linealidad?
# Sí

# 4. Calcule el coeficiente de correlación e interprete el resultado.
from scipy.stats import pearsonr
r, _ = pearsonr(X, Y)
print(f'Coeficiente de correlación: {r: 0.4f}\n')

# 5. Calcule el coeficiente de determinación e interprete el
resultado.
print(f'Coeficiente de determinación: {r ** 2: 0.4f}\n')


# 6. Obtenga la recta de regresión ajustada y grafíquelo sobre el
gráfico de
# dispersión.
import statsmodels.api as sm
x_constante = sm.add_constant(X)
modelo = sm.OLS(Y, x_constante).fit()

b0, b1 = modelo.params

fun = lambda x: b0 + b1 * x

Yc = fun(X)

plt.plot(X, Yc, color = 'black', linestyle = '--')

from sklearn.metrics import r2_score  # recomendada
r2 = r2_score(Y, Yc)
print(f'Coeficiente de determinación: {r2: 0.4f}\n')

# 7. Obtenga un intervalo de confianza del 95% para la pendiente de la
recta de
# regresión ajustada ( b1 )
nivel_de_confianza = 0.95
intervalo_de_confianza = modelo.conf_int(alpha = 1 -
nivel_de_confianza)
intervalo_de_confianza_b1 = intervalo_de_confianza.iloc[1]
print(f'Intervalo de confianza para b1 de {nivel_de_confianza: 0.0%}')
print(f'{intervalo_de_confianza_b1[0]: 0.4f} < b1 <
```

```
{intervalo_de_confianza_b1[1]: 0.4f}\n')


# 8. Calcule los residuales y trace un nuevo gráfico de dispersión.
Comente,
# ¿Parece que se verifican los supuestos?
residuales = modelo.resid
plt.figure()
plt.scatter(X, residuales, color = 'red')
plt.xlabel('interrupciones')
plt.ylabel('Residuales')
ax = plt.gca()
ax.spines['top'].set_visible(False)
ax.spines['right'].set_visible(False)
plt.axhline(y = 0, color = 'gray', linestyle = '--')

# 9. Realice la prueba de Shapiro para los residuales y comente el
resultado.
from scipy.stats import shapiro
_, valor_p_sh = shapiro(residuales)
print(f'valor-p de Shapiro: {valor_p_sh: 0.4f}\n')


# 10. Realice la prueba de Brausch-Pagan para los residuales y comente
el
# resultado.
from statsmodels.stats.api import het_breuschpagan
_, valor_p_bp, _, _ = het_breuschpagan(residuales, x_constante)
print(f'valor_p de Breusch-Pagan: {valor_p_bp: 0.4f}\n')

#Realice una tabla ANOVA e interprete el resultado.
from statsmodels.formula.api import ols
# Y ~ X
modelo_2 = ols('Calificacion ~ Interrupciones', data = df).fit()
tabla_anova = sm.stats.anova_lm(modelo_2)
tabla_anova
```

```
Coeficiente de correlación: -0.9928

Coeficiente de determinación:  0.9858

Coeficiente de determinación:  0.9858

Intervalo de confianza para b1 de  95%
-3.1363 < b1 < -2.4637

valor-p de Shapiro:  0.0548

valor_p de Breusch-Pagan:  0.2482
```

	df	sum_sq	mean_sq	F	PR(>F)
Interrupciones	1.0	2352.0	2352.000000	415.058824	9.090964e-07
Residual	6.0	34.0	5.666667	NaN	NaN




Residuales
2
1
0
−1
−2
5.0
7.5
10.0
12.5
15.0
17.5
20.0
22.5
25.0
interrupciones