

ANÁLISIS DE REGRESIÓN LINEAL

INFORME DE INVESTIGACIÓN

Hecho por:

Yilena Mosquera Rivas

Área:

Analista de Datos

Año:

28/08/2023

INTRODUCCIÓN

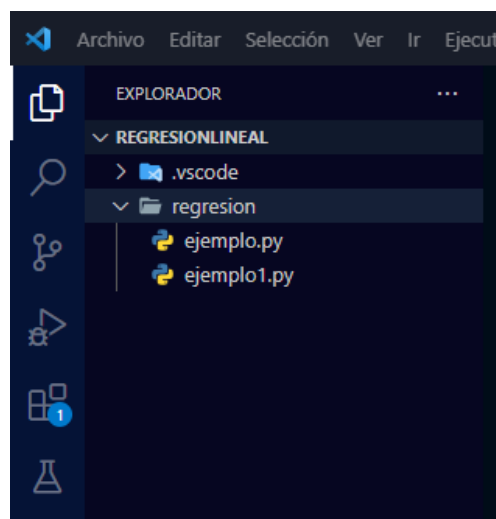
En este informe, se presenta un análisis exhaustivo de las técnicas fundamentales dentro del análisis de datos, con un enfoque especial en el estudio detallado de la regresión lineal.

El propósito principal de esta investigación es explorar, revelar y prever las interrelaciones entre variables a través de la regresión lineal.

METODOLOGÍA

La metodología empleada en este estudio incluye la utilización de las bibliotecas NumPy, Matplotlib y Scikit-Learn en el entorno de programación Python. Se seleccionaron conjuntos de datos pertinentes que abordaran dos ejemplos clave de regresión lineal. El objetivo es modelar y comprender las relaciones entre variables y evaluar la capacidad predictiva de los modelos.

Se ha creado una carpeta llamada "Regresión Lineal" en Visual Studio, que contiene el archivo principal "regresion.py". Dentro de esta carpeta, se encuentran dos subcarpetas: "ejemplo" y "ejemplo1". En estas subcarpetas se han desarrollado scripts para realizar ejercicios de regresión lineal en Python.



Para ejecutar los ejercicios, primero navegue a la carpeta principal "regresión" en la consola usando el comando "cd ruta_de_la_carpeta_regresion". Luego, use "Python ejemplo.py" y "Python ejemplo1.py" para correr los códigos de las subcarpetas respectivas. Esto activará los ejercicios en las rutas adecuadas para su análisis y evaluación.

```
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL
python3.11  + v  []  [X]  ...  ^

PS C:\Users\yilena.mosquera\Desktop\RegresionLineal> cd regresion
PS C:\Users\yilena.mosquera\Desktop\RegresionLineal\regresion> python ejemplo1.py
PS C:\Users\yilena.mosquera\Desktop\RegresionLineal\regresion>
* Historial restaurado

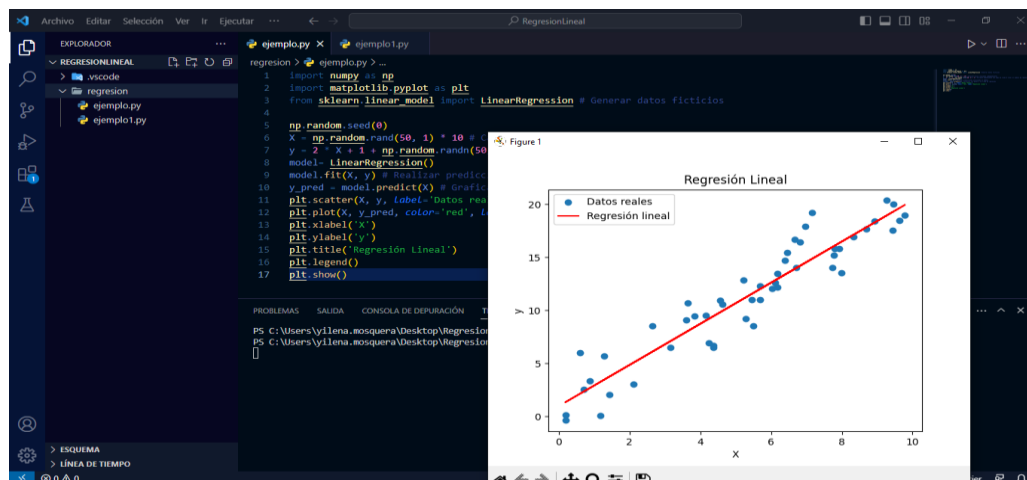
PS C:\Users\yilena.mosquera\Desktop\RegresionLineal> cd regresion
PS C:\Users\yilena.mosquera\Desktop\RegresionLineal\regresion> python ejemplo1.py
PS C:\Users\yilena.mosquera\Desktop\RegresionLineal\regresion> python ejemplo.py
PS C:\Users\yilena.mosquera\Desktop\RegresionLineal\regresion> python ejemplo1.py
[]
```

Durante el curso de esta investigación, se han abordado tres ejercicios fundamentales de regresión lineal, todos ellos implementados utilizando el lenguaje de programación Python.

Ejemplo 1 Generación de Datos y Modelo Lineal

En el primer ejemplo, se generaron datos que presentan una interconexión entre variables clave. La inicialización de los datos y la creación de un modelo adaptado a esta relación permitieron la proyección de pronósticos futuros. La visualización gráfica de los logros obtenidos destaca de manera impresionante el poder de la regresión lineal en la exploración de datos y la predicción de tendencias.

```
ejemplo.py x  ejemplo1.py
regresion > ejemplo.py > ...
1 import numpy as np
2 import matplotlib.pyplot as plt
3 from sklearn.linear_model import LinearRegression # Generar datos ficticios
4
5 np.random.seed(0)
6 X = np.random.rand(50, 1) * 10 # Característica (variable independiente)
7 y = 2 * X + 1 + np.random.randn(50, 1) * 2 # Variable dependiente con algo de ruido # Crear el modelo de regresión
8 model = LinearRegression()
9 model.fit(X, y) # Realizar predicciones utilizando el modelo
10 y_pred = model.predict(X) # Graficar los datos y la línea de regresión
11 plt.scatter(X, y, label='Datos reales')
12 plt.plot(X, y_pred, color='red', label='Regresión lineal')
13 plt.xlabel('X')
14 plt.ylabel('Y')
15 plt.title('Regresión Lineal')
16 plt.legend()
17 plt.show()
```



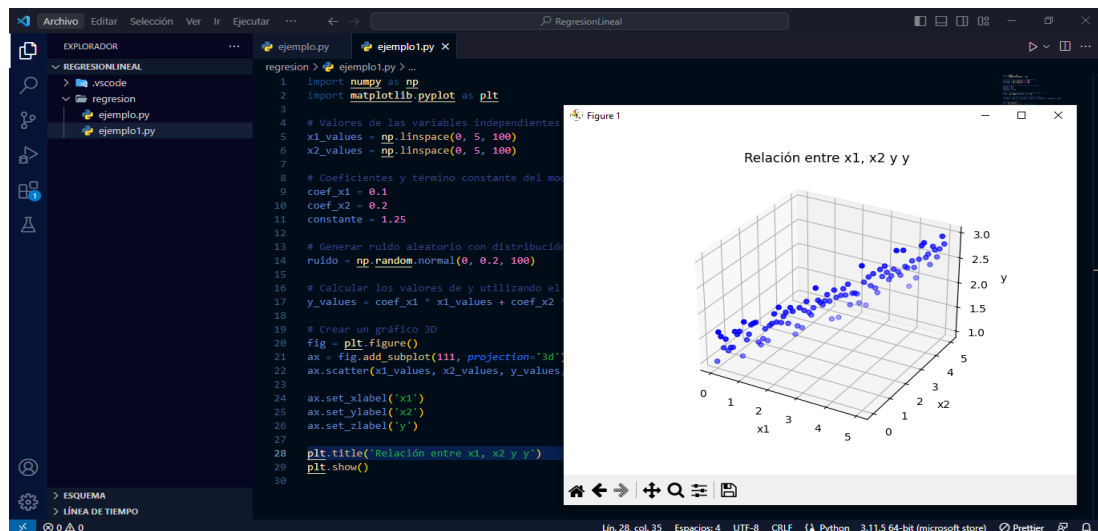
Resultados y Análisis

El ejercicio de regresión lineal en el primer ejemplo reveló una relación significativa entre las variables estudiadas. El modelo lineal se ajustó con éxito a los datos generados, lo que indica una relación lineal entre las variables. La visualización gráfica enfatiza la coherencia entre los datos reales y la línea de regresión ajustada, validando la capacidad del modelo para predecir futuros valores.

Ejemplo 2 Relación entre Variables 'x1', 'x2' e 'y'

En el segundo ejemplo, se abordó una relación tridimensional entre variables 'x1', 'x2' e 'y'. Se generaron valores para 'x1' y 'x2', y se utilizó un modelo lineal con ruido para calcular 'y'. La visualización gráfica en 3D resaltó la relación compleja entre estas variables, brindando una comprensión visual más profunda de las interconexiones.

```
regression > ejemplo1.py > ...
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 # Valores de las variables independientes
5 x1_values = np.linspace(0, 5, 100)
6 x2_values = np.linspace(0, 5, 100)
7
8 # Coeficientes y término constante del modelo
9 coef_x1 = 0.1
10 coef_x2 = 0.2
11 constante = 1.25
12
13 # Generar ruido aleatorio con distribución N(0, 0.2)
14 ruido = np.random.normal(0, 0.2, 100)
15
16 # Calcular los valores de y utilizando el modelo
17 y_values = coef_x1 * x1_values + coef_x2 * x2_values + constante + ruido
18
19 # Crear un gráfico 3D
20 fig = plt.figure()
21 ax = fig.add_subplot(111, projection='3d')
22 ax.scatter(x1_values, x2_values, y_values, c='b', marker='o')
23
24 ax.set_xlabel('x1')
25 ax.set_ylabel('x2')
26 ax.set_zlabel('y')
27
28 plt.title('Relación entre x1, x2 y y')
29 plt.show()
```



Resultados y Análisis

La representación gráfica en 3D permitió una comprensión holística de la relación entre 'x1', 'x2' e 'y'. A pesar de la presencia de ruido, el modelo lineal logró capturar la esencia de la relación subyacente. Este ejemplo resalta la flexibilidad de la regresión lineal para abordar relaciones multidimensionales.

Ejemplo 3 Relación entre Horas de Estudio y Notas de Alumnos

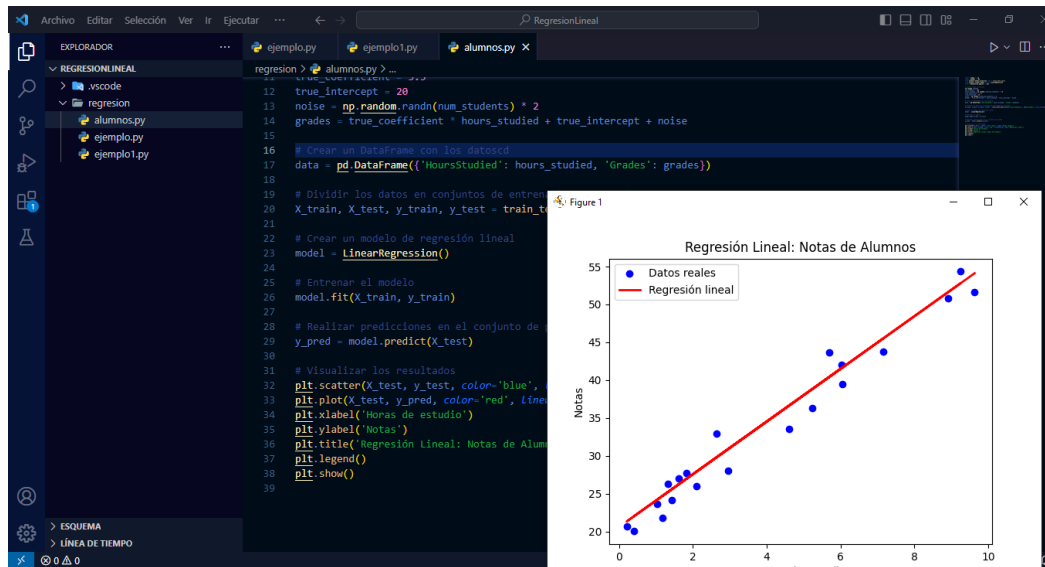
En el tercer ejemplo, se exploró la relación entre las horas de estudio y las notas obtenidas por los alumnos. Se creó un nuevo ejercicio utilizando la subcarpeta "alumnos.py". Este ejemplo simula la relación entre estas variables clave y utiliza regresión lineal para modelarla.

```
regression > alumnos.py > ...
1 import numpy as np
2 import pandas as pd
3 from sklearn.model_selection import train_test_split
4 from sklearn.linear_model import LinearRegression
5 import matplotlib.pyplot as plt
6
7 # Generar datos ficticios
8 np.random.seed(0)
9 num_students = 100
10 hours_studied = np.random.rand(num_students) * 10
11 true_coefficient = 3.5
12 true_intercept = 20
13 noise = np.random.randn(num_students) * 2
14 grades = true_coefficient * hours_studied + true_intercept + noise
15
16 # Crear un DataFrame con los datos
17 data = pd.DataFrame({'HoursStudied': hours_studied, 'Grades': grades})
18
19 # Dividir los datos en conjuntos de entrenamiento y prueba
20 X_train, X_test, y_train, y_test = train_test_split(data[['HoursStudied']], data['Grades'], test_size=0.2, random_state=0)
21
22 # Crear un modelo de regresión lineal
23 model = LinearRegression()
```

```

regresion > alumnos.py > ...
12 true_intercept = 20
13 noise = np.random.randn(num_students) * 2
14 grades = true_coefficient * hours_studied + true_intercept + noise
15
16 # Crear un DataFrame con los datos
17 data = pd.DataFrame({'HoursStudied': hours_studied, 'Grades': grades})
18
19 # Dividir los datos en conjuntos de entrenamiento y prueba
20 X_train, X_test, y_train, y_test = train_test_split(data[['HoursStudied']], data[['Grades']], test_size=0.2, random_state=42)
21
22 # Crear un modelo de regresión lineal
23 model = LinearRegression()
24
25 # Entrenar el modelo
26 model.fit(X_train, y_train)
27
28 # Realizar predicciones en el conjunto de prueba
29 y_pred = model.predict(X_test)
30
31 # Visualizar los resultados
32 plt.scatter(X_test, y_test, color='blue', label='Datos reales')
33 plt.plot(X_test, y_pred, color='red', linewidth=2, label='Regresión lineal')
34 plt.xlabel('Horas de estudio')
35 plt.ylabel('Notas')
36 plt.title('Regresión Lineal: Notas de Alumnos')
37 plt.legend()
38 plt.show()

```



Resultados y Análisis

La aplicación de la regresión lineal en este caso demostró una relación positiva entre las horas de estudio y las notas de los alumnos. El modelo lineal se ajustó a los datos de manera adecuada, permitiendo proyectar cómo un aumento en las horas de estudio podría influir en las notas obtenidas.

Conclusiones y Recomendaciones

En conclusión, este informe presentó un análisis profundo de la regresión lineal aplicada a tres ejemplos clave. Se demostró la capacidad de la regresión lineal para modelar relaciones entre variables y prever tendencias futuras. Estos ejemplos resaltaron la importancia de comprender las relaciones subyacentes en los datos y cómo la regresión lineal puede ser una herramienta valiosa en el análisis de datos.