

Nombre de la práctica	Regresión Lineal			No.	4
Asignatura:	SIMULACIÓN	Carrera:	ISIC	Duración de la práctica (Hrs)	8

I. Lugar de realización de la práctica (laboratorio, taller, aula u otro): Aula

II. Material empleado:
Laptop

IV. Desarrollo de la práctica:

Regresion lineal: Costo de un incidente de seguridad

En este caso se explican los fundamentos basicos de la regresion lineal aplicada a un caso sencillo relacionado con la ciberseguridad.

Enunciado del ejercicio

El ejercicio consiste en predecir el costo de un incidente de seguridad en base al numero de equipos que se ven afectados. El Data Set es generado de manera aleatoria

1.- Generacion del DataSet

```
import numpy as np
x = 2 * np.random.rand(100, 1)
y = 4 + 3 * X + np.random.randn(100, 1)

print("La longitud del DataSet es: ", len(X))
```

2.- Visualizacion del DataSet

```
[1]: import matplotlib.pyplot as plt
    %matplotlib inline

[7]: plt.plot(X, y, "b.")
    plt.show()
```



```
[ ]: plt.plot(X, y, "b.")
plt.title("Grafico Chido")
plt.xlabel("Equipos Afectados(u/1,000)")
plt.ylabel("Costo del incidente(u/10,000)")
plt.show()
```

3.- Modificación del DataSet ¶

```
[ ]: import pandas as pd

[ ]: data = {'No_Equipos-Afectados': X.flatten(), 'Costo': y.flatten()}
df = pd.DataFrame(data)
df.head(10)

[ ]: # Escalado del numero de equipos afectados.
df['No_Equipos_Afectados'] = df['No_Equipos_Afectados'] * 1000
df['No_Equipos_Afectados'] = df['No_Equipos_Afectados'].astype('int')
#Escalado del costo
df['Costo'] = df['Costo'] * 10000
df.head(10)

[ ]: # Escalado del numero de equipos afectados.
df['No_Equipos_Afectados'] = df['No_Equipos_Afectados'] * 1000
df['No_Equipos_Afectados'] = df['No_Equipos_Afectados'].astype('int')
#Escalado del costo
df['Costo'] = df['Costo'] * 10000
df.head(10)

[ ]: plt.title("Grafico Chido")
plt.xlabel("Equipos Afectados(u/1,000)")
plt.ylabel("Costo del incidente(u/10,000)")
plt.show()
```

4.- Construcción del Modelo

```
[ ]: from sklearn.linear_model import LinearRegression

[ ]: #Construcción del modelo y ajuste de hipótesis
lin_reg = LinearRegression()
lin_reg.fit(df['No_Equipos_Afectados'].values.reshape(-1,1), df['costo'].values)

[ ]: # Parametro theta 0
lin_reg.intercept_

[ ]: # Parametro theta 1
lin_reg.coef_

[ ]: Predicción para el valor mínimo y máximo del DataSet
X_min_max = np.array([df['No_Equipos_Afectados'].min(), df['No_Equipos_Afectados'].max()])
y_train_pred = lin_reg.predict(X_min_max)
```

```
[ ]: # Representacion Grafica de la funcion de hipotesis generada.
plt.plot(X_min_max, y_train_pred, "g-")
plt.plot(dt['No_Equipos_Afectados', dt['Costo'])
plt.xlabel("Equipos Afectados")
plt.ylabel("Costo del Incidente")
plt.show()
```

5.- Prediccion de nuevos ejemplos

```
[ ]: x_new = np.array([[1255]]) # No. de equipos afectados.

# Prediccion del costo que tendria el incidente.
Costo = lin_reg.predic(x_new)

print("El costo del incidente seria: ", int(costo[0]))

[ ]: plt.plot(df['No_Equipos_Afectados'], df['Costo'], "b.")
plt.plot(X_min_max, y_train_pred, "g-")
plt.plot(x_new, Costo, "rX")
plt.xlabel("Equipos Afectados TESJI")
plt.ylabel("Costo del incidente")
plt.show()
```

V. Conclusiones:

En esta práctica, hemos abordado el análisis de regresión lineal como una herramienta fundamental para modelar la relación entre variables. Utilizando un conjunto de datos, aplicamos el método de regresión lineal para predecir valores y analizar patrones.

Los resultados obtenidos mostraron que la regresión lineal es eficaz para identificar tendencias y relaciones lineales entre variables. A través de la visualización de los datos y los residuos, pudimos evaluar la bondad del ajuste y la validez del modelo. También aprendimos a interpretar coeficientes y a realizar predicciones basadas en el modelo ajustado.