Inteligencia Artificial
Act 9: Programando Regresión Lineal en Python

Estudiante: José Luis Calderón Galarza - 2132939
Docente: Luis Ángel Gutiérrez Rodríguez

1 Introducción

La regresión lineal es un método estadístico utilizado para modelar la relación entre una variable dependiente y y una o más variables independientes X. En su forma más simple, la regresión lineal busca ajustar una línea recta que minimice el error cuadrático entre los puntos de datos y la línea ajustada. Este tipo de modelo es ampliamente utilizado en predicción y análisis de tendencias, donde el objetivo es predecir un valor de salida en función de las características de entrada.

2 Metodología

En esta actividad, se implementó un modelo de regresión lineal en Python para analizar la relación entre la cantidad de palabras de un artículo y la cantidad de veces que es compartido en redes sociales. A continuación, se describe cada parte del código utilizado:

1. **Importación de librerías:** Se cargan las bibliotecas necesarias para la manipulación de datos, visualización y construcción del modelo.

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
plt.rcParams['figure.figsize'] = (16, 9)
plt.style.use('ggplot')
from sklearn import linear_model
from sklearn.metrics import mean_squared_error, r2_score
```

2. **Carga y exploración de datos:** Se carga el conjunto de datos desde un archivo CSV y se examinan las primeras filas, dimensiones y estadísticas descriptivas.

```
data = pd.read_csv("./articulos_ml.csv")
print(data.shape)
print(data.head())
print(data.describe())
```

3. **Visualización inicial:** Se generan histogramas para comprender la distribución de los datos.

```
data.drop(['Title', 'url', 'Elapsed days'], axis=1).hist()
plt.show()
```

4. **Filtrado de datos:** Se eliminan valores extremos para centrarse en la zona de mayor concentración de puntos.

```
filtered_data = data[(data['Word count'] <= 3500) & (data['# Shares'] <= 80000)]
```

5. **Gráfico de dispersión:** Se visualizan los datos filtrados y se colorean los puntos según si están por encima o por debajo de la media de palabras.

```
colores = ['orange', 'blue']
tamanios = [30, 60]
asignar = [colores[0] if row['Word count'] > 1808 else colores[1] for index, row in filter
plt.scatter(filtered_data['Word count'], filtered_data['# Shares'], c=asignar, s=tamanios[
plt.show()
```

6. **Entrenamiento del modelo:** Se selecciona la variable independiente (cantidad de palabras) y la dependiente (compartidos en redes sociales) para entrenar el modelo de regresión lineal.

```
X_train = np.array(filtered_data[['Word count']])
y_train = filtered_data['# Shares'].values
regr = linear_model.LinearRegression()
regr.fit(X_train, y_train)
```

7. **Predicciones y evaluación del modelo:** Se generan predicciones y se calculan los coeficientes, el error cuadrático medio y el puntaje de varianza.

```
y_pred = regr.predict(X_train)
print('Coefficients:', regr.coef_)
print('Independent term:', regr.intercept_)
print("Mean squared error: %.2f" % mean_squared_error(y_train, y_pred))
print('Variance score: %.2f' % r2_score(y_train, y_pred))
```

8. **Visualización del modelo:** Se grafica la línea de regresión ajustada sobre los datos de entrenamiento.

```
plt.scatter(X_train[:, 0], y_train, c=asignar, s=tamanios[0])
plt.plot(X_train[:, 0], y_pred, color='red', linewidth=3)
plt.xlabel('Cantidad de Palabras')
plt.ylabel('Compartido en Redes')
plt.title('Regresión Lineal')
plt.show()
```

9. **Predicción para 2000 palabras:** Se usa el modelo entrenado para predecir cuántas veces será compartido un artículo con 2000 palabras.

```
y_Dosmil = regr.predict([[2000]])
print(int(y_Dosmil[0]))
```

3 Resultados

3.1 Histograma de características de entrada

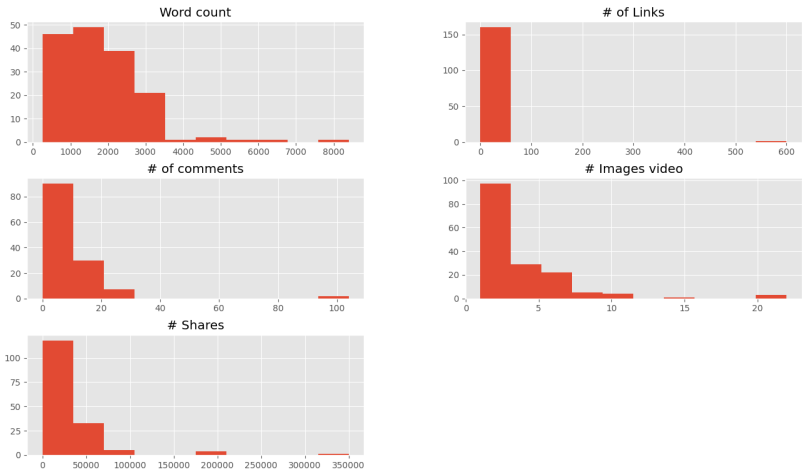

Figure 1: Histograma de las características de entrada, mostrando la distribución de la cantidad de palabras y los compartidos en redes sociales.

3.2 Gráfico de dispersión de datos filtrados

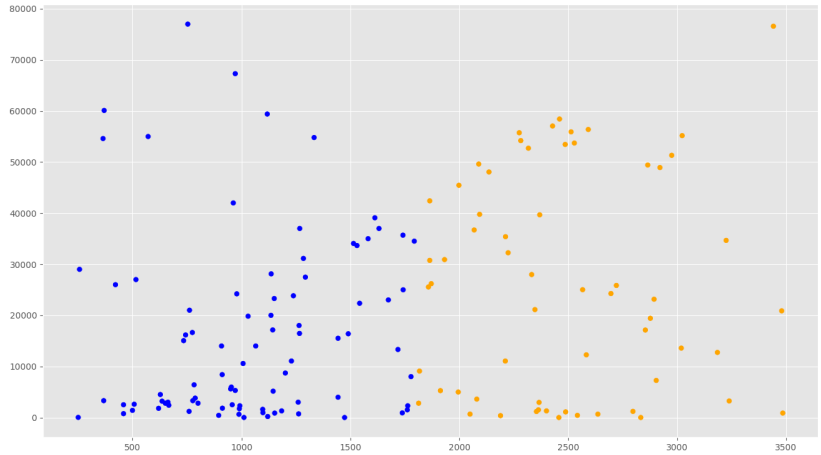

Figure 2: Gráfico de dispersión de los datos filtrados, coloreados según la media de la cantidad de palabras.

3.3 Línea de regresión ajustada

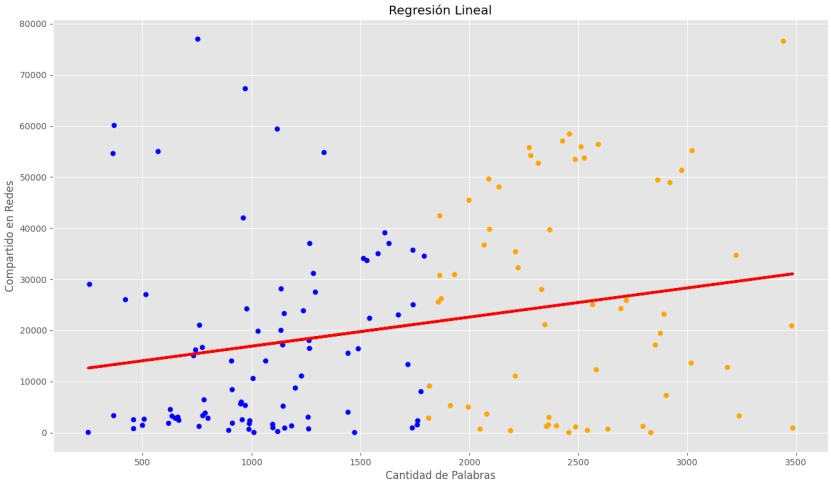

Figure 3: Línea de regresión lineal ajustada a los datos filtrados.

4 Conclusión

Se ha aplicado con éxito un modelo de regresión lineal para predecir el número de veces que un artículo será compartido en redes sociales, en función de la cantidad de palabras que contiene. Finalmente, se realizó una predicción para un artículo de 2000 palabras, obteniendo un valor estimado de 22595.