APLICACIÓN: *REGRESIÓN LINEAL*

Nexus-Probability

CURSO 3 (PROCESOS ESTOCÁSTICOS I)

PARTE 1 / LECCIÓN 3

1. La esperanza condicional en un modelo de regresión lineal

La **esperanza condicional** es un concepto clave en probabilidad y estadística, que describe el valor esperado de una variable aleatoria Y dado un conjunto de variables X. En el contexto de un modelo de regresión lineal, la esperanza condicional tiene una interpretación central, ya que el modelo busca describir cómo Y depende sistemáticamente de X, aislando la variabilidad aleatoria.

La esperanza condicional en regresión lineal simple

Un modelo de regresión lineal simple asume que la relación entre Y y X está dada por:

$$Y = \beta_0 + \beta_1 X + \epsilon,$$

donde:

- β_0 y β_1 son los parámetros del modelo (intercepto y pendiente, respectivamente),
- ϵ es el término de error aleatorio, que satisface:

$$\mathbb{E}[\epsilon] = 0, \quad \text{y generalmente } \epsilon \sim \mathcal{N}(0, \sigma^2).$$

Dado este modelo, la esperanza condicional de Y dado X es:

$$\mathbb{E}[Y \mid X] = \mathbb{E}[\beta_0 + \beta_1 X + \epsilon \mid X].$$

Debido a la linealidad de la esperanza y al hecho de que ϵ es independiente de X con $\mathbb{E}[\epsilon] = 0$, se obtiene:

$$\mathbb{E}[Y \mid X] = \beta_0 + \beta_1 X.$$

Esto muestra que la esperanza condicional es una función lineal de X, que describe la relación promedio entre Y y X.

Propiedades probabilísticas del modelo

En un modelo de regresión lineal, la distribución condicional de Y dado X tiene las siguientes propiedades:

- La esperanza condicional $\mathbb{E}[Y \mid X]$ es la **tendencia central** de la distribución de Y para un valor dado de X.
- La varianza condicional de Y dado X es constante e igual a σ^2, es decir:

  $$\text{Var}(Y \mid X) = \mathbb{E}[(Y - \mathbb{E}[Y \mid X])^2 \mid X] = \sigma^2.$$

  Esto significa que la dispersión de Y alrededor de $\mathbb{E}[Y \mid X]$ no depende de X.
- La independencia entre X y el término de error ϵ implica que el modelo captura toda la relación sistemática entre Y y X en $\mathbb{E}[Y \mid X]$.

Regresión lineal múltiple

Para un modelo de regresión lineal múltiple con k variables independientes $X_1, X_2, \ldots, X_k$, el modelo se escribe como:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_k X_k + \epsilon.$$

En este caso, la esperanza condicional de Y dado el vector de variables independientes $\mathbf{X} = (X_1, X_2, \ldots, X_k)$ es:

$$\mathbb{E}[Y \mid \mathbf{X}] = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_k X_k.$$

Esto significa que:

- La relación esperada entre Y y las variables $X_1, X_2, \ldots, X_k$ es una función lineal de estas últimas.
- La distribución condicional de Y dado $\mathbf{X}$ sigue siendo normal, con:

  $$Y \mid \mathbf{X} \sim \mathcal{N}(\mathbb{E}[Y \mid \mathbf{X}], \sigma^2).$$

Interpretación en términos de predicción

En el modelo de regresión lineal, la esperanza condicional $\mathbb{E}[Y \mid X]$ o $\mathbb{E}[Y \mid \mathbf{X}]$ tiene una interpretación como el valor esperado de Y para un valor dado de X (o $\mathbf{X}$). Este valor representa la mejor predicción promedio de Y bajo el supuesto de mínimos cuadrados. La regresión lineal se basa en encontrar los coeficientes $\beta_0, \beta_1, \ldots, \beta_k$ que minimicen la suma de los errores al cuadrado:

$$\min_{\beta_0,\beta_1,\ldots,\beta_k} \sum_{i=1}^{n} (Y_i - \mathbb{E}[Y \mid \mathbf{X}_i])^2 .$$

La esperanza condicional $\mathbb{E}[Y \mid \mathbf{X}]$ es el corazón del modelo de regresión lineal, ya que describe la relación sistemática entre Y y las variables independientes. Su base probabilística garantiza que el modelo no solo capture la tendencia promedio, sino que también explique cómo el ruido ϵ se distribuye de manera independiente alrededor de esta tendencia. Este enfoque proporciona una herramienta poderosa para modelar y predecir relaciones en datos observados.

2. Aplicación.

El siguiente modelo fue implementado en `Python` utilizando una variedad de bibliotecas importantes para el tratamiento de datos, visualización y modelado. Para la manipulación de datos se emplearon `pandas` y `numpy`. Las visualizaciones se realizaron con `matplotlib` y `seaborn`. El preprocesado y modelado se llevó a cabo utilizando `sklearn` , que incluye módulos como `LogisticRegression` y `train_test_split`, además de `statsmodels` para análisis estadísticos más avanzados. Finalmente, se utilizó la biblioteca `warnings` para gestionar advertencias y asegurar una ejecución fluida del código.

El siguente slot ejecuta todas las bibliotecas necesarias para este scrip.

```python
# Tratamiento de datos
import pandas as pd
import numpy as np

#Graficos
import matplotlib.pyplot as plt
from matplotlib import style
import seaborn as sns

#Procesado y modelado
from scipy.stats import pearsonr
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import r2_score
from sklearn.metrics import mean_squared_error
```

```python
        import statsmodels.api as sm
        import statsmodels.formula.api as smf
        from scipy import stats
        import statsmodels as sms
        from tqdm import tqdm
        import time

        # configuracion de matplotlib
        plt.rcParams['image.cmap']="bwr"
        plt.rcParams['figure.dpi']="100"
        plt.rcParams['savefig.bbox']="tight"
        style.use('ggplot') or plt.style.use('ggplot')

        # configuracion de warnings
        import warnings
        warnings.filterwarnings('ignore')

```

2.1. Descripcion de la base de datos.

Jamboree es una empresa educativa con sede en la India, especializada en la preparación y asesoramiento para estudiantes que buscan ingresar a universidades tanto en su país como en el extranjero. Ofrecen cursos intensivos de preparación para exámenes estandarizados como el SAT, ACT, GRE, GMAT y TOEFL, ayudando a mejorar los puntajes de los estudiantes. **Jamboree** se destaca por su enfoque personalizado y su amplia experiencia en la educación internacional.
A continuación se detalla el contexto y el significado de cada variable:

1. **GRE Score:** Puntaje obtenido en el examen GRE (Graduate Record Examination), utilizado para la admisión en programas de posgrado.

2. **TOEFL Score:** Puntaje obtenido en el examen TOEFL (Test of English as a Foreign Language), que evalúa la competencia en inglés.

3. **University Rating:** Calificación o clasificación de la universidad donde el estudiante ha solicitado admisión.

4. **SOP:** Statement of Purpose (Declaración de Propósito), un ensayo donde el estudiante explica sus motivaciones y metas académicas.

5. **LOR:** Letter of Recommendation (Carta de Recomendación), una carta escrita por un profesor o supervisor que evalúa las habilidades y aptitudes del estudiante.

6. **CGPA:** Cumulative Grade Point Average (Promedio de Calificaciones), el promedio acumulativo de las calificaciones obtenidas durante el grado anterior.

7. **Research:** Experiencia en investigación académica o científica del estudiante.
8. **Chance of Admit:** Probabilidad de admisión, una estimación de las posibilidades del estudiante de ser admitido en el programa deseado.

En este contexto específico:

- **Chance of Admit:** Probabilidad de admisión, una estimación de las posibilidades del estudiante de ser admitido en el programa deseado.
- **Variables predictoras:** Usaremos las variables mencionadas anteriormente en la lista, enumeradas del 1 al 7.

Al considerar todas estas variables como predictores, se obtiene una visión holística y completa del perfil académico, profesional y personal del estudiante, lo cual es crucial para hacer predicciones informadas sobre su probabilidad de admisión en un programa universitario específico.

Jamboree es reconocida por su compromiso en ayudar a los estudiantes a alcanzar sus metas académicas mediante una preparación rigurosa y personalizada para el proceso de admisión universitaria en instituciones de renombre a nivel mundial.

Click para ver la Base de Datos de Jamboree

Lectura de la base de datos con pd.read_csv().

```
data = pd.read_csv("jamboree_dataset.csv")   #Lectura de la base de datos
data.head()     #Muestra las primeras 5 filas del DF.
```

 Chance of Admit	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research
0 0.92	1	337	118	4	4.5	4.5	9.65	1
1 0.76	2	324	107	4	4.0	4.5	8.87	1
2 0.72	3	316	104	3	3.0	3.5	8.00	1
3 0.80	4	322	110	3	3.5	2.5	8.67	1
4 0.65	5	314	103	2	2.0	3.0	8.21	0

Vemos que las columnas de **Serial No.** y de **Research** son categoricas,por tanto las desecharemos del analisis, usaremos data.drop("" , axis = 1)

```
data = data.drop(['Serial No.','Research'],axis=1) #Elimina las columnas mencionadas.
data.head()
```

2.2. Análisis Descriptivo.

Usaremos `data.describe()` para generar un resumen estadistico de las columnas.

```
        data.describe()

```

GRE Score

- **Media:** 316 puntos.
- **Desviación Estándar:** 11 puntos.
- **Rango:** Desde 290 puntos hasta 340 puntos.

TOEFL Score

- **Media:** 107.192 puntos.
- **Desviación Estándar:** 6.081868 puntos.
- **Rango:** Desde 92 puntos hasta 120 puntos.

University Rating

- **Media:** 3.114.
- **Desviación Estándar:** 1.143512.
- **Rango:** Desde 1 hasta 5.

SOP

- **Media:** 3.374 puntos.
- **Desviación Estándar:** 0.991004 puntos.
- **Rango:** Desde 1 hasta 5.

LOR

- **Media:** 3.484 puntos.
- **Desviación Estándar:** 0.92545 puntos.
- **Rango:** Desde 1 hasta 5.

CGPA

- **Media:** 8.57644 puntos.
- **Desviación Estándar:** 0.604813 puntos.
- **Rango:** Desde 6.8 puntos hasta 9.92 puntos.

Chance of Admit

- **Media:** 0.72174.
- **Desviación Estándar:** 0.14114.
- **Rango:** Desde 0.34 % hasta 0.97 %.

Ahora procedemos a la verificación de datos faltantes con `isna().sum()`

```
    data.isna().sum() #Cuenta el numero de celdas vacias.
```

```
GRE Score            0
TOEFL Score          0
University Rating    0
SOP                  0
LOR                  0
CGPA                 0
Chance of Admit      0
dtype: int64
```

En efecto, no se encontraron datos faltantes.

Ahora usaremos `.corr()`, calcula la correlación de Pearson entre las columnas de un DataFrame.

```
correlacion = data.corr() #Matriz de correlacion del DataFrame

#Creacion de mapa de calor de correlacion
sns.heatmap(correlacion, annot=True, cmap="YlGnBu")
plt.title('Mapa de calor de correlacion')
plt.show()
```

A continuación se muestra la salida del *slot* previo.

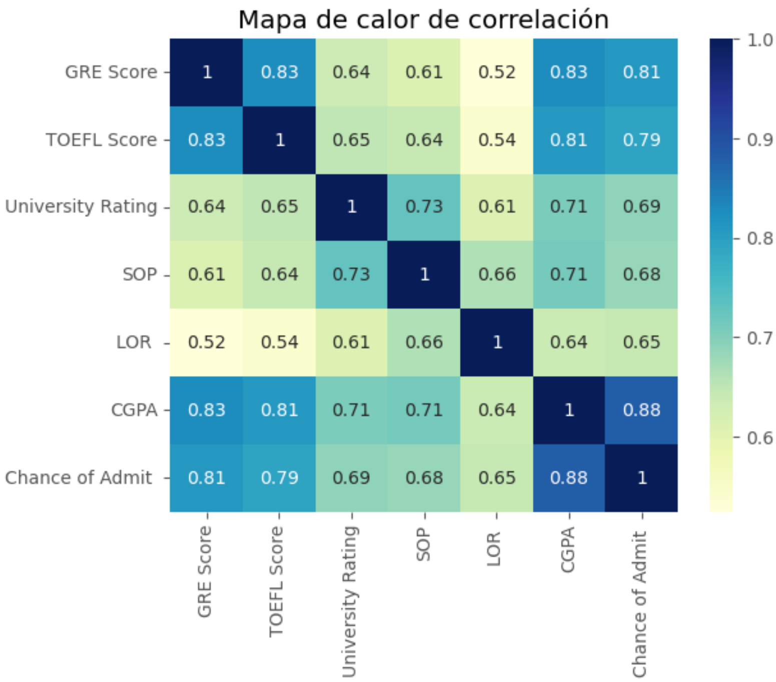

Figura 1: Mapa de Calor de Correlación de la Base de Datos.

Ahora usaremos `sns.pairplot()` para visualizar las relaciones entre múltiples variables numéricas de nuestro DataFrame.

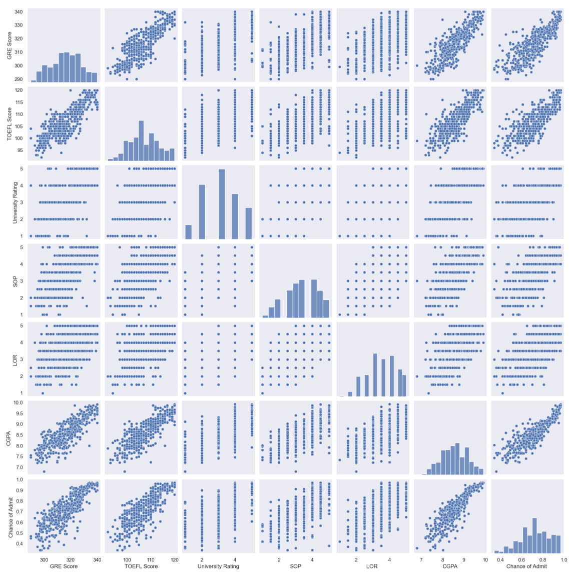

Figura 2: Pair-Plot de la Base de Datos.

El gráfico **pairplot** permite observar las relaciones por pares entre las distintas variables. En la diagonal principal se muestran los histogramas de cada variable, donde se puede notar que aquellas de carácter numérico continuo (como los scores) presentan mejores comportamientos. Las variables categóricas, por otro lado, presentan un nivel de visualización más complejo.

En los **gráficos de dispersión**, se observa que las variables que exhiben una mayor relación y un comportamiento más adecuado para predecir la variable **Çhance of Admit''** son: TOEFL Score, GRE Score y CGPA. Las variables categóricas como LOR, SOP y University Ranking, aunque parecen no tener un buen comportamiento para predecir Çhance of Admit.ª simple vista, muestran una correlación mayor a 0.6 en el diagrama de correlación. Por lo tanto, no podemos descartarlas de nuestro análisis.

Procedemos a la verificacion de posibles **Outliers**, usaremos la función `sns.boxplot()`

```
colors = ['#00b8a9', '#f7b801', '#db2b30', '#6a4c93', '#67b7dc', '#b7d63c', '#7f8fa6']
fig, axs = plt.subplots(4, 2, figsize=(12, 16))

# Plot 1: GRE Score
sns.boxplot(data[["GRE Score"]], color=colors[0], ax=axs[0, 0])
axs[0, 0].set_title('Boxplot de GRE Score')

# Plot 2: TOEFL Score
sns.boxplot(data[["TOEFL Score"]], color=colors[1], ax=axs[0, 1])
axs[0, 1].set_title('Boxplot de TOEFL Score')

# Plot 3: University Rating
sns.boxplot(data[["University Rating"]], color=colors[2], ax=axs[1, 0])
axs[1, 0].set_title('Boxplot de University Rating')

# Plot 4: SOP
sns.boxplot(data[["SOP"]], color=colors[3], ax=axs[1, 1])
axs[1, 1].set_title('Boxplot de SOP')

# Plot 5: LOR
sns.boxplot(data[["LOR "]], color=colors[4], ax=axs[2, 0])
axs[2, 0].set_title('Boxplot de LOR')

# Plot 6: CGPA
sns.boxplot(data[["CGPA"]], color=colors[5], ax=axs[2, 1])
axs[2, 1].set_title('Boxplot de CGPA')

# Plot 7: Chance of Admit
```

```
sns.boxplot(data[["Chance of Admit "]], color=colors[6], ax=axs
    [3, 0])
axs[3, 0].set_title('Boxplot de Chance of Admit')
fig.tight_layout()
plt.show()
```

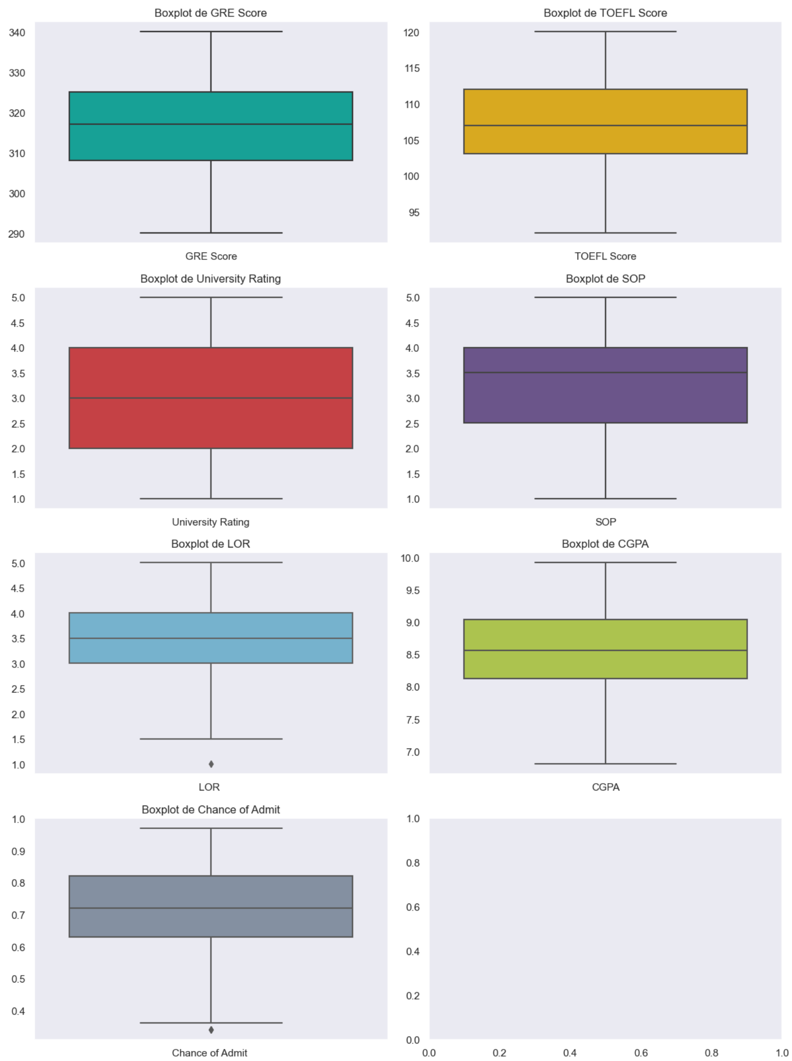

Figura 3: BoxPlot de las Varibles Predictoras.

Basandonos en los boxplots anteriores notamos que las variables con presencia de posibles outliers son LOR y Chance of admit.

La variables LOR muestra un puntaje bastante bajo en la carta de recomendación de un alumno mientras que Chance of admite asigna una probabilidad por debajo del 4 % de acceder a la universidad desada.

Algunas posibles causas de un valor tan bajo en LOR pueden ser las siguientes:

- Desempeño Académico Excepcional o Deficiente: Un puntaje bajo en la carta de recomendación podría deberse a un desempeño académico muy deficiente del alumno, lo que puede llevar a que los recomendadores no proporcionen una calificación alta.
- Conflictos Personales:El alumno podría haber tenido conflictos personales con la persona que escribió la carta de recomendación, lo que puede influir negativamente en la evaluación.
- Errores de Evaluación:Puede haber un error o sesgo en la evaluación realizada por la persona que escribió la carta, como malentendidos o errores al interpretar el desempeño del alumno.
- Falta de Información Completa: El recomendador puede no tener información completa o precisa sobre el desempeño y las cualidades del alumno, lo que puede llevar a una evaluación baja.

2.3. Creación del Modelo

Recuerde que en `X_Orig`, unicamente van almacenadas las variables predictoras.

```
X_orig = data.drop("Chance of Admit ",axis=1)
X_orig.head()
```

```
y_orig = data["Chance of Admit "]
y_orig.head()
```

```
0    0.92
1    0.76
2    0.72
3    0.80
4    0.65
Name: Chance of Admit , dtype: float64
```

```
# Dividir los datos originales X_orig (caracteristicas) y y_orig
    (etiquetas) en conjuntos de entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(X_orig,
   y_orig, random_state=230624, train_size=0.80)

# Crear un nuevo DataFrame df combinando las caracteristicas y
   etiquetas de entrenamiento
df = pd.concat([X_train, y_train], axis=1)
```

El siguiente slot, es la parte primordial del código, aquí se entrena todo el modelo, y como salida nos proporciona un resumen de toda la información de la sesión.

```python
# Anadir una columna constante (intercepto) a X_train
X_train = sm.add_constant(X_train, prepend=True)

# Crear un modelo de regresion lineal ordinaria (OLS) utilizando Statsmodels
modelo1 = sm.OLS(y_train, X_train)

# Ajustar el modelo utilizando los datos de entrenamiento
modelo1 = modelo1.fit()

# Imprimir un resumen detallado del modelo ajustado
print(modelo1.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:        Chance of Admit   R-squared:                       0.818
Model:                            OLS   Adj. R-squared:                  0.816
Method:                 Least Squares   F-statistic:                     354.0
Date:                Sun, 23 Jun 2024   Prob (F-statistic):          2.87e-143
Time:                        19:35:52   Log-Likelihood:                 558.78
No. Observations:                 400   AIC:                            -1106.
Df Residuals:                     394   BIC:                            -1082.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         -1.3889      0.107    -12.933      0.000      -1.600      -1.178
GRE Score      0.0021      0.001      4.028      0.000       0.001       0.003
TOEFL Score    0.0028      0.001      3.024      0.003       0.001       0.005
University.R   0.0105      0.004      2.636      0.009       0.003       0.018
LOR            0.0181      0.004      4.096      0.000       0.009       0.027
CGPA           0.1205      0.011     11.309      0.000       0.100       0.141
==============================================================================
Omnibus:                       89.788   Durbin-Watson:                   1.964
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              213.566
Skew:                          -1.118   Prob(JB):                     4.21e-47
Kurtosis:                       5.796   Cond. No.                     1.19e+04
==============================================================================
```

```python
from complete_analisis import *

coeficientes(modelo1)
```

```
Beta_0: -1.38976
Beta_1: 0.00214
Beta_2: 0.00285
Beta_3: 0.01060
Beta_4: -0.00044
Beta_5: 0.01817
Beta_6: 0.12069
```

Ahora, podemos construir el modelo de Regresión Lineal Múltiple, vea que:

$$\hat{y} = -\underbrace{1.38976}_{\beta_0} + \underbrace{0.00214}_{\beta_1} x_1 + \underbrace{0.00285}_{\beta_2} x_2 + \underbrace{0.01060}_{\beta_3} x_3 - \underbrace{0.00044}_{\beta_4} x_4 + \underbrace{0.01817}_{\beta_5} x_5 + \underbrace{0.12069}_{\beta_6} x_6$$

Prueba Global

Ahora, vamos a evaluar si al menos uno de los predictores **(variables independientes)** tiene un efecto significativo sobre la variable de respuesta **(variable dependiente)**.

H_0 : No hay efecto significativo de las variables predictoras en la variable de respuesta.

H_1 : Al menos una de las variables predictoras tiene un efecto significativo en la variable de respuesta.

Ahora en símbolos:

$$H_0 : \beta_1 = \beta_2 =, \cdots, = \beta_6 = 0$$
$$H_1 : \beta_1 \neq \beta_2 \neq, \cdots, \neq \beta_6 \neq 0$$

```python
prueba_global(modelo1)
```

```
Estadística F : 294.26472705879445
P_value_f: 0.0000000000
Progreso: 100% 100/100 [00:01<00:00, 96.89it/s]H_0 ES rechazado.
Al menos una de las variables predictoras tiene un efecto
significativo en la variable de respuesta.
```

En efecto, vea que al menos una de las variables predictoras tiene un efecto significativo en la variable de respuesta.

$$\text{P-value} = 0.000 < 0.05 = \alpha \text{ (True)}$$

Por lo tanto, H_0 es rechazado y podemos inferir que:

$$\beta_1 \neq \beta_2 \neq, \cdots, \neq \beta_6 \neq 0$$

Con un nivel $\alpha = 0.05$ de significancia.

Pruebas Individuales.

H_0 : No hay efecto significativo de la variable β_i en la variable de respuesta.

H_1 : La variables predictora β_i tiene un efecto significativo en la variable de respuesta.

Ahora en simbolos:

$$H_0 : \beta_i = 0$$
$$H_1 : \beta_i \neq 0$$

```
prueba_ind(modelo1)
```

```
PRUEBA PARA BETA_0
  Estadistica t para Beta_0: -12.86207
  P-value para Beta_0: 0.00000
  H_0 ES rechazado.
 Beta_0 tiene un efecto significativo en la variable de respuesta.
__________________________________________________________________
PRUEBA PARA BETA_1
  Estadistica t para Beta_1: 4.01574
  P-value para Beta_1: 0.00007
  H_0 ES rechazado.
 Beta_1 tiene un efecto significativo en la variable de respuesta.
__________________________________________________________________
PRUEBA PARA BETA_2
  Estadistica t para Beta_2: 3.02080
  P-value para Beta_2: 0.00269
  H_0 ES rechazado.
 Beta_2 tiene un efecto significativo en la variable de respuesta.
__________________________________________________________________
PRUEBA PARA BETA_3
  Estadistica t para Beta_3: 2.44204
  P-value para Beta_3: 0.01504
  H_0 ES rechazado.
 Beta_3 tiene un efecto significativo en la variable de respuesta.
__________________________________________________________________
PRUEBA PARA BETA_4
  Estadistica t para Beta_4: -0.08248
  P-value para Beta_4: 0.93431
  H_0 NO es rechazado.
 Beta_4 NO tiene efecto significativo en la variable de respuesta.
__________________________________________________________________
```

```
PRUEBA PARA BETA_5
  Estadistica t para Beta_5: 3.91547
  P-value para Beta_5: 0.00011
  H_0 ES rechazado.
 Beta_5 tiene un efecto significativo en la variable de respuesta.
-------------------------------------------------------------------
PRUEBA PARA BETA_6
  Estadistica t para Beta_6: 11.04250
  P-value para Beta_6: 0.00000
  H_0 ES rechazado.
 Beta_6 tiene un efecto significativo en la variable de respuesta.
-------------------------------------------------------------------
```

En resumen, tenemos de las pruebas individuales:

$$\begin{aligned}
\beta_0 &\rightarrow \text{ Contribuye al modelo}\\
\beta_1 &\rightarrow \text{ Contribuye al modelo}\\
\beta_2 &\rightarrow \text{ Contribuye al modelo}\\
\beta_3 &\rightarrow \text{ Contribuye al modelo}\\
\beta_4 &\rightarrow \text{ NO contribuye al modelo}\\
\beta_5 &\rightarrow \text{ Contribuye al modelo}\\
\beta_6 &\rightarrow \text{ Contribuye al modelo}
\end{aligned}$$

Ecuación de la Mejora del Modelo.

Ahora, podemos construir el modelo de Regresión Lineal Multiple, vea que:

$$\hat{y} = -\underbrace{1.38889}_{\beta_0} + \underbrace{0.00214}_{\beta_1} x_1 + \underbrace{0.00285}_{\beta_2} x_2 + \underbrace{0.01045}_{\beta_3} x_3 + \underbrace{0.01805}_{\beta_4} x_4 + \underbrace{0.12050}_{\beta_5} x_5$$

La esperanza condicional $\mathbb{E}[Y \mid \mathbf{X}]$ es el corazón del modelo de regresión lineal, ya que describe la relación sistemática entre Y y las variables independientes. Su base probabilística garantiza que el modelo no solo capture la tendencia promedio, sino que también explique cómo el ruido ϵ se distribuye de manera independiente alrededor de esta tendencia. Este enfoque proporciona una herramienta poderosa para modelar y predecir relaciones en datos observados.