```
import pandas as pd

# Read data from a specific sheet within the Excel file
df = pd.read_excel('/content/Datos Modelos Lineales.xlsx',
sheet_name='Regresión Lineal Múltiple Ej')

df.head()
```


```
{"summary":"{\n  \"name\": \"df\",\n  \"rows\": 108,\n  \"fields\": [\
n    {\n      \"column\": \"Factor Coagulaci\\u00f3n\",\n
\"properties\": {\n        \"dtype\": \"number\",\n        \"std\":
1.4251714658823018,\n        \"min\": 2.6,\n        \"max\": 11.2,\n
\"num_unique_values\": 48,\n        \"samples\": [\n          3.9,\n
6.9,\n          4.5\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"\\
u00cdndice pron\\u00f3stico\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 16,\n        \"min\": 8,\n
\"max\": 99,\n        \"num_unique_values\": 54,\n        \"samples\":
[\n          54,\n          88,\n          50\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"Funci\\u00f3n de enzima\",\n
\"properties\": {\n        \"dtype\": \"number\",\n        \"std\":
20,\n        \"min\": 23,\n        \"max\": 119,\n
\"num_unique_values\": 63,\n        \"samples\": [\n          33,\n
48,\n          81\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Funci\\u00f3n de h\\u00edgado\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 1.0056046943582424,\n
\"min\": 0.74,\n        \"max\": 6.4,\n        \"num_unique_values\":
90,\n        \"samples\": [\n          4.1,\n          3.5,\n
2.91\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Edad\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 11,\n        \"min\": 30,\n        \"max\": 70,\n
\"num_unique_values\": 40,\n        \"samples\": [\n          70,\n
67,\n          59\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"G\\
u00e9nero\",\n      \"properties\": {\n        \"dtype\": \"number\",\
n        \"std\": 0,\n        \"min\": 0,\n        \"max\": 1,\n
\"num_unique_values\": 2,\n        \"samples\": [\n          1,\n
0\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Alcohol\\n(moderado)\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 0,\n        \"min\": 0,\n
\"max\": 1,\n        \"num_unique_values\": 2,\n        \"samples\":
[\n          0,\n          1\n        ],\n        \"semantic_type\":
\"\",\n        \"description\": \"\"\n      }\n    },\n    {\n
\"column\": \"Alcohol\\n(severo)\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 0,\n        \"min\": 0,\n
\"max\": 1,\n        \"num_unique_values\": 2,\n        \"samples\":
```

```
[\n          1,\n          0\n        ],\n        \"semantic_type\":
\"\",\n        \"description\": \"\"\n      }\n    },\n    {\n
\"column\": \"Sobrevivencia\\n(d\\u00edas)\",\n      \"properties\":
{\n        \"dtype\": \"number\",\n        \"std\": 347,\n
\"min\": 181,\n        \"max\": 2343,\n        \"num_unique_values\":
101,\n        \"samples\": [\n          569,\n          716\
n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Unnamed: 9\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": null,\n        \"min\": null,\n
\"max\": null,\n        \"num_unique_values\": 0,\n
\"samples\": [],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Unnamed: 10\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": null,\n        \"min\": null,\n
\"max\": null,\n        \"num_unique_values\": 0,\n
\"samples\": [],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Unnamed: 11\",\n      \"properties\": {\n        \"dtype\":
\"category\",\n        \"num_unique_values\": 3,\n        \"samples\":
[],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
}\n    }\n  ]\n}","type":"dataframe","variable_name":"df"}
```


```
from sklearn.preprocessing import MinMaxScaler

# Crear el scaler
scaler = MinMaxScaler()

# Normalizar las columnas especificadas
df[['Factor Coagulación', 'Índice pronóstico', 'Función de enzima',
'Función de hígado']] = scaler.fit_transform(df[['Factor Coagulación',
'Índice pronóstico', 'Función de enzima', 'Función de hígado']])

import statsmodels.api as sm

X = df[['Factor Coagulación', 'Índice pronóstico', 'Función de
enzima', 'Función de hígado', 'Edad', 'Género', 'Alcohol\n(moderado)',
'Alcohol\n(severo)']]
y = df['Sobrevivencia\n(días)']

# Agregar constante (intercepto) al modelo
X = sm.add_constant(X)

# Ajustar el modelo
model = sm.OLS(y, X).fit()

print(model.summary())
```

```
                            OLS Regression Results
==============================================================================
```

```
==========
Dep. Variable:     Sobrevivencia
(días)   R-squared:                       0.775
Model:                            OLS   Adj. R-squared:
0.757
Method:                 Least Squares   F-statistic:
42.69
Date:                Wed, 04 Sep 2024   Prob (F-statistic):
8.70e-29
Time:                        01:17:03   Log-Likelihood:
-704.12
No. Observations:                 108   AIC:
1426.
Df Residuals:                      99   BIC:
1450.
Df Model:                           8

Covariance Type:               nonrobust

================================================================================
================
                         coef    std err          t      P>|t|
[0.025      0.975]
--------------------------------------------------------------------------------
----------------
const                -595.0929    119.740     -4.970      0.000    -
832.683    -357.503
Factor Coagulación   453.2609    125.622      3.608      0.000
204.000     702.522
Índice pronóstico    738.2324    103.242      7.151      0.000
533.379     943.086
Función de enzima    854.1430     97.046      8.801      0.000
661.582    1046.704
Función de hígado    429.4316    144.560      2.971      0.004
142.593     716.271
Edad                   0.6410      1.447      0.443      0.659      -
2.229       3.511
Género                13.0926     34.118      0.384      0.702     -
54.606      80.791
Alcohol
(moderado)   -41.2676     38.529     -1.071      0.287    -117.718
35.183
Alcohol
(severo)     195.7070     50.095      3.907      0.000      96.308
295.106
================================================================================
========
Omnibus:                       45.879   Durban-Watson:
1.770
```

```
Prob(Omnibus):                  0.000   Jarque-Bera (JB):
214.951
Skew:                           1.304   Prob(JB):
2.11e-47
Kurtosis:                       9.401   Cond. No.
629.
================================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is
correctly specified.
```

Observamos que todas las variables a excepción de edad y género son variables significativas.

```
# Variables predictoras significativas
X_significativas = df[['Factor Coagulación', 'Índice pronóstico',
'Función de enzima', 'Función de hígado']]
X_significativas = sm.add_constant(X_significativas)

# Ajustar el modelo con variables significativas
model_significativas = sm.OLS(y, X_significativas).fit()

# Ver el resumen del modelo
print(model_significativas.summary())
```

```
                            OLS Regression Results

==============================================================================
Dep. Variable:     Sobrevivencia
(días)   R-squared:                       0.711
Model:                            OLS   Adj. R-squared:
0.700
Method:                 Least Squares   F-statistic:
63.40
Date:                Wed, 04 Sep 2024   Prob (F-statistic):
6.30e-27
Time:                        01:17:19   Log-Likelihood:
-717.67
No. Observations:                 108   AIC:
1445.
Df Residuals:                     103   BIC:
1459.
Df Model:                           4

Covariance Type:               nonrobust

================================================================================
================
```

```
                            coef    std err          t      P>|t|
[0.025      0.975]
------------------------------------------------------------------------------
----------------
const                -580.0033     99.363     -5.837      0.000    -
777.067    -382.939
Factor Coagulación    564.8957    134.644      4.195      0.000
297.862     831.930
Índice pronóstico     768.7555    113.355      6.782      0.000
543.943     993.568
Función de enzima     888.3017    107.234      8.284      0.000
675.628    1100.975
Función de hígado     317.3125    155.538      2.040      0.044
8.840     625.785
==============================================================================
========
Omnibus:                       59.069   Durbin-Watson:
2.059
Prob(Omnibus):                  0.000   Jarque-Bera (JB):
336.468
Skew:                           1.699   Prob(JB):
8.65e-74
Kurtosis:                      10.952   Cond. No.
16.4
==============================================================================
========

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is
correctly specified.
```

```
from statsmodels.tools import add_constant
from sklearn.preprocessing import PolynomialFeatures

poly = PolynomialFeatures(degree=2, include_bias=False)
X_poly = poly.fit_transform(X_significativas.drop('const', axis=1))
X_poly = pd.DataFrame(X_poly,
columns=poly.get_feature_names_out(X_significativas.columns[1:]))
X_poly = add_constant(X_poly)

# Ajustar el modelo con términos polinomiales
model_poly = sm.OLS(y, X_poly).fit()

# Ver el resumen del modelo con términos polinomiales
print(model_poly.summary())
```

```
                            OLS Regression Results

==============================================================================
==========
```

```
Dep. Variable:     Sobrevivencia
(días)   R-squared:                       0.790
Model:                            OLS   Adj. R-squared:
0.759
Method:                 Least Squares   F-statistic:
25.06
Date:                Wed, 04 Sep 2024   Prob (F-statistic):
1.49e-25
Time:                        01:17:27   Log-Likelihood:
-700.33
No. Observations:                 108   AIC:
1431.
Df Residuals:                      93   BIC:
1471.
Df Model:                          14

Covariance Type:               nonrobust

================================================================================
==================================
                                            coef    std err          t
P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------
--------------------------------
const                                   336.4146    381.004      0.883
0.380    -420.183    1093.013
Factor Coagulación                     -765.7353    808.553     -0.947
0.346   -2371.361     839.891
Índice pronóstico                       817.5406    819.336      0.998
0.321    -809.498    2444.579
Función de enzima                     -1239.9252    638.485     -1.942
0.055   -2507.830      27.980
Función de hígado                      -574.8781    906.262     -0.634
0.527   -2374.535    1224.778
Factor Coagulación^2                    589.1580    717.533      0.821
0.414    -835.721    2014.037
Factor Coagulación Índice pronóstico  -375.0895    803.616     -0.467
0.642   -1970.913    1220.734
Factor Coagulación Función de enzima  1992.7951    822.977      2.421
0.017     358.527    3627.064
Factor Coagulación Función de hígado    62.1117   1319.603      0.047
0.963   -2558.359    2682.582
Índice pronóstico^2                    -451.2967    522.440     -0.864
0.390   -1488.759     586.166
Índice pronóstico Función de enzima     17.2073    649.056      0.027
0.979   -1271.689    1306.103
Índice pronóstico Función de hígado   1724.8054   1044.696      1.651
0.102    -349.754    3799.365
Función de enzima^2                    1403.8805    459.054      3.058
```

```
0.003     492.291    2315.470
Función de enzima Función de hígado    -96.8656   1099.007     -0.088
0.930   -2279.275    2085.544
Función de hígado^2                   -465.8850    905.897     -0.514
0.608   -2264.817    1333.047
==============================================================================
========
Omnibus:                       23.564   Durbin-Watson:
2.121
Prob(Omnibus):                  0.000   Jarque-Bera (JB):
43.790
Skew:                           0.900   Prob(JB):
3.10e-10
Kurtosis:                       5.548   Cond. No.
231.
==============================================================================
========

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is
correctly specified.
```

```python
from statsmodels.stats.outliers_influence import
variance_inflation_factor

# Calcular FIV para el modelo con variables significativas
fiv = pd.DataFrame()
fiv['variable'] = X_significativas.columns
fiv['VIF'] = [variance_inflation_factor(X_significativas.values, i)
for i in range(X_significativas.shape[1])]
print(fiv)
```

```
             variable        VIF
0               const  29.370301
1  Factor Coagulación   1.467321
2   Índice pronóstico   1.291654
3   Función de enzima   1.610220
4   Función de hígado   2.250657
```

```python
import seaborn as sns
import matplotlib.pyplot as plt

# Gráfico de residuos vs. valores ajustados
plt.figure(figsize=(10, 6))
plt.subplot(1, 2, 1)
sns.scatterplot(x=fitted, y=residuals)
plt.axhline(y=0, color='r', linestyle='--')
plt.xlabel('Fitted values')
plt.ylabel('Residuals')
plt.title('Residuals vs Fitted')
```

```
# Histograma de residuos
plt.subplot(1, 2, 2)
sns.histplot(residuals, kde=True)
plt.xlabel('Residuals')
plt.title('Histogram of Residuals')

plt.tight_layout()
plt.show()
```