

A1.5 Solución de problemas

```
In [1]: # Exportación de librerías.  
import pandas as pd  
import numpy as np  
import seaborn as sns  
import matplotlib.pyplot as plt  
from sklearn.model_selection import train_test_split  
from sklearn.linear_model import LinearRegression  
import statsmodels.api as sm
```

1. Abajo, se imprime el tipo de datos que hay en el dataframe de la base de datos "A1.5 Calificaciones".

```
In [2]: data = pd.read_csv("A1.5 Calificaciones.csv")
print(data.head(1), '\n')
print(data.dtypes)
```

```
Escuela Sexo Edad HorasDeEstudio Reprobadas Internet Faltas G1 G2 G3
0      GP   F    18                  2            0       no     6   5   6   6

Escuela          object
Sexo            object
Edad           int64
HorasDeEstudio  int64
Reprobadas      int64
Internet        object
Faltas          int64
G1              int64
G2              int64
G3              int64
dtype: object
```

2. La variables de internet, sexo y escuela se tratan con dummy encoding, se generarán nuevas columnas para nuestro modelo de regresión. El resultado será 1 si es positiva la selección y 0 con falso dependiendo del caso en cada columna.

```
In [3]: data = pd.get_dummies(data,columns=["Escuela","Sexo","Internet"])
print(data.head(1))
print(data.dtypes)
```

```

      Edad  HorasDeEstudio  Reprobadas  Faltas  G1  G2  G3  Escuela_GP  \
0      18              2            0       6   5   6   6           1

      Escuela_MS  Sexo_F  Sexo_M  Internet_no  Internet_yes
0          0        1        0                  1                   0

Edad          int64
HorasDeEstudio  int64
Reprobadas      int64
Faltas          int64
G1              int64
G2              int64
G3              int64
Escuela_GP     uint8
Escuela_MS     uint8
Sexo_F          uint8
Sexo_M          uint8
Internet_no    uint8
Internet_yes   uint8
dtype: object

```

3. Mediante el método de Tukey, se calculan los límites de Tukey a través del rango intercuartílico, que se aplicaran sobre la columna faltas, de tal forma que se trabajen únicamente con valores centrales y significativos. Después de identificarlos, se retiran del conjunto de datos para la generación del modelo.

```

In [4]: Q1 = data["Faltas"].quantile(0.25)
Q3 = data["Faltas"].quantile(0.75)

IQR = Q3 - Q1

lower_b = Q1-3*IQR
upper_b = Q3+3*IQR

print("Lower bound: ", lower_b)
print("Upper bound: ", upper_b, "\n")

outliers = data[(data["Faltas"] < lower_b) | (data["Faltas"] > upper_b)]
print("Los valores atípicos en la columna Faltas son: \n", outliers["Faltas"])

```

Lower bound: -24.0
 Upper bound: 32.0

Los valores atípicos en la columna Faltas son:

```

74      54
183     56
276     75
307     38
315     40
Name: Faltas, dtype: int64

```

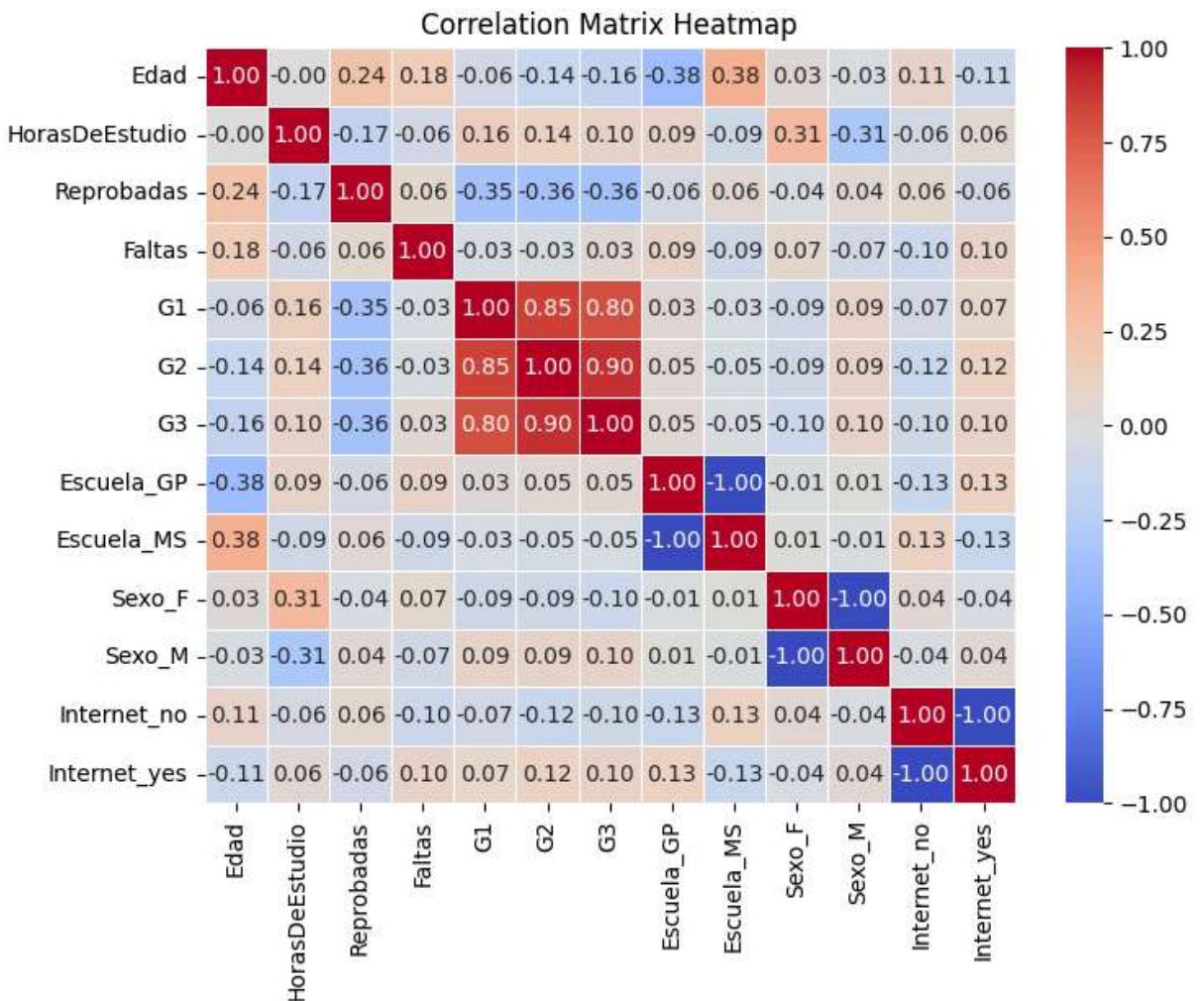
4. Se generá una matriz de correlaciones para detectar problemas de colinealidad en las variables independientes.

```
In [5]: corr_matrix = data.corr()

plt.figure(figsize=(8,6))
sns.heatmap(corr_matrix, annot=True, cmap="coolwarm", fmt=".2f", linewidths=0.5)
plt.title("Correlation Matrix Heatmap")
plt.show()
high_corr_vars = set()
threshold = 0.8

for i in range(len(corr_matrix.columns)):
    for j in range(i):
        if(corr_matrix.iloc[i, j]>threshold):
            high_corr_vars.add(corr_matrix.columns[j])
            high_corr_vars.add(corr_matrix.columns[i])

# Print analysis
if high_corr_vars:
    print(f"Variables que deben ser removidas por colinealidad alta: {high_corr_vars}")
    # data.drop(columns=high_corr_vars, inplace=True) # Se eliminan
else:
    print("Ninguna variable debe ser removida.")
```



VARIABLES que deben ser removidas por colinealidad alta: {'G3', 'G2', 'G1'}

A través de estos resultados, podemos indicar que G1, G2 y G3 representan información que está muy relacionada entre sí, uno pudiera optar por dejar únicamente G3, no obstante, se optará por dejar todas las variables independientes ya que se puede esperar que los resultados de un periodo influyan significativamente a otros.

5. Los posibles términos de interacción en el modelo serán las variables de las faltas y la calificación final, ya que el interés y el tiempo dedicado a los temas del cursos suelen tender a disminuir por cada una. Igualmente, se incluirá una relación entre la cantidad de materias reprobadas y la calificación final, ya que su impacto en la nota se esperaría que sea más evidente.

```
In [6]: data["Faltas_G3_Interaccion"] = data["Faltas"]*data["G3"]
data["Reprobadas_G3_Interaccion"] = data["Reprobadas"]*data["G3"]
print(data.head(5))
```

	Edad	HorasDeEstudio	Reprobadas	Faltas	G1	G2	G3	Escuela_GP	\
0	18		2	0	6	5	6	6	1
1	17		2	0	4	5	5	6	1
2	15		2	3	10	7	8	10	1
3	15		3	0	2	15	14	15	1
4	16		2	0	4	6	10	10	1

	Escuela_MS	Sexo_F	Sexo_M	Internet_no	Internet_yes	\
0	0	1	0	1	0	0
1	0	1	0	0	1	1
2	0	1	0	0	1	1
3	0	1	0	0	1	1
4	0	1	0	1	0	0

	Faltas_G3_Interaccion	Reprobadas_G3_Interaccion
0	36	0
1	24	0
2	100	30
3	30	0
4	40	0

6. Modelo de regresión lineal múltiple. Se genera una gráfica con base a los datos obtenidos.

```
In [7]: model = LinearRegression()
X = data[["Edad","HorasDeEstudio","Reprobadas","Faltas","G1","G2","Escuela_GP","Escuela_MS","Sexo_F","Sexo_M","Internet_no","Internet_yes"]]
y = data["G3"]

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

X_train = X_train.reset_index(drop=True)
y_train = y_train.reset_index(drop=True)
X_test = X_test.reset_index(drop=True)
y_test = y_test.reset_index(drop=True)

X_train_const = sm.add_constant(X_train)
X_test_const = sm.add_constant(X_test)
```

```
model = sm.OLS(y_train, X_train_const).fit()

# Resumen
print(model.summary())

# Predicción con los valores de prueba
y_pred = model.predict(X_test_const)

plt.figure(figsize=(8,6))
sns.scatterplot(x=y_test, y=y_pred)
plt.plot([0, 20], [0, 20], color='red', linestyle='--')
plt.xlabel("Calificación real actual")
plt.ylabel("Predicción de calificación")
plt.title("Actual vs. Predicción")
plt.show()

correlation = np.corrcoef(y_test, y_pred)[0,1]
print(f"\nConclusión: El modelo tiene una correlación igual a {correlation:.2f} entre\n" "Al ser la correlación cercana a 1, el modelo tiene una buena capacidad de pre
```

OLS Regression Results

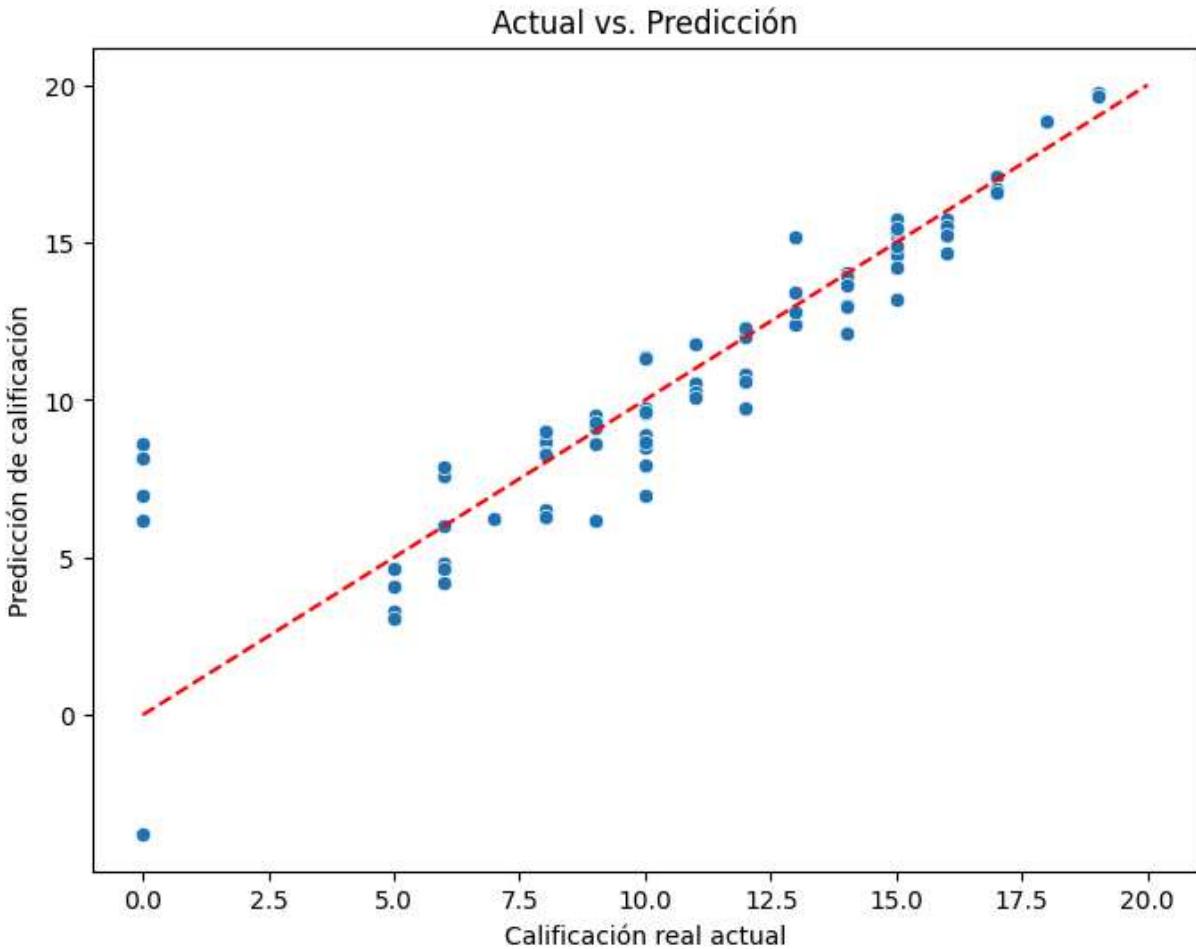
Dep. Variable:	G3	R-squared:	0.868		
Model:	OLS	Adj. R-squared:	0.863		
Method:	Least Squares	F-statistic:	182.1		
Date:	Wed, 19 Feb 2025	Prob (F-statistic):	1.15e-126		
Time:	19:00:44	Log-Likelihood:	-609.20		
No. Observations:	316	AIC:	1242.		
Df Residuals:	304	BIC:	1287.		
Df Model:	11				
Covariance Type:	nonrobust				
<hr/>					
<hr/>					
	coef	std err	t	P> t	
0.975]				[0.025	
<hr/>					
const	1.1413	0.617	1.850	0.065	-0.073
2.356					
Edad	-0.2323	0.086	-2.695	0.007	-0.402
-0.063					
HorasDeEstudio	-0.0364	0.127	-0.287	0.774	-0.286
0.213					
Reprobadas	-1.7289	0.223	-7.743	0.000	-2.168
-1.290					
Faltas	0.0341	0.043	0.797	0.426	-0.050
0.118					
G1	0.1927	0.058	3.299	0.001	0.078
0.308					
G2	0.8732	0.049	17.940	0.000	0.777
0.969					
Escuela_GP	0.5771	0.282	2.046	0.042	0.022
1.132					
Escuela_MS	0.5642	0.411	1.374	0.170	-0.244
1.372					
Sexo_F	0.4006	0.322	1.244	0.215	-0.233
1.034					
Sexo_M	0.7407	0.329	2.250	0.025	0.093
1.389					
Internet_no	0.5876	0.344	1.706	0.089	-0.090
1.265					
Internet_yes	0.5537	0.328	1.687	0.093	-0.092
1.200					
Faltas_G3_Interaccion	-0.0008	0.004	-0.182	0.855	-0.009
0.008					
Reprobadas_G3_Interaccion	0.2155	0.027	7.874	0.000	0.162
0.269					
<hr/>					
Omnibus:	176.576	Durbin-Watson:	2.085		
Prob(Omnibus):	0.000	Jarque-Bera (JB):	1425.225		
Skew:	-2.192	Prob(JB):	3.28e-310		
Kurtosis:	12.435	Cond. No.	1.33e+18		
<hr/>					

Notes:

[1] Standard Errors assume that the covariance matrix of the errors is correctly spe

cified.

[2] The smallest eigenvalue is 2.13e-30. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.



Conclusión: El modelo tiene una correlación igual a 0.89 entre los valores actuales y los predecidos. Al ser la correlación cercana a 1, el modelo tiene una buena capacidad de predicción.

Adicionalmente, se puede establecer que la suposición inicial de incluir la interacción de Faltas y calificación final no tiene una relación estadística suficiente ($t < -1.96$ es falso) ($p < 0.05$ es falso), así como las variables, Internet_no, Internet_yes, Sexo_F, Sexo_M y HorasDeEstudio. Faltas falla en rechazar la hipótesis nula, por lo que no se puede asegurar su valor t de manera confiable. Por lo tanto, se debería considerar retirar o replantear algunas de estas variables.

Código de honor: Doy mi palabra de que he realizado esta actividad con integridad académica.