A1.3 Solución de problemas y selección de características

La predicción del desempeño académico a partir de información demográfica y de rendimiento previo es un problema de interés en el análisis de datos educativos, ya que permite identificar patrones relevantes y comprender los factores que influyen en los resultados finales de los estudiantes. No obstante, en aplicaciones reales, los conjuntos de datos utilizados para este tipo de análisis rara vez se encuentran listos para ser utilizados directamente en modelos predictivos, debido a la presencia de variables cualitativas, valores atípicos, escalas inconsistentes y relaciones redundantes entre variables.

En este reporte se desarrolla un modelo de regresión lineal múltiple para predecir la calificacion final de estudiantes, utilizando un conjunto de datos que incluye información académica y demográfica. El análisis aborda de manera explícita los retos asociados al uso de datos reales, tales como el tratamiento de variables cualitativas, la identificación y análisis de valores atípicos, la detección de colinealidad entre variables explicativas y la selección de un subconjunto adecuado de características relevantes para la predicción.

El conjunto de datos se encuentra dividido en datos de entrenamiento y datos de prueba, lo que permite evaluar el desempeño del modelo en información no utilizada durante el proceso de ajuste. A lo largo del desarrollo se siguen buenas prácticas para evitar fuga de datos, asegurando que todas las transformaciones y decisiones relacionadas con la selección de características se realicen únicamente con los datos de entrenamiento y posteriormente se apliquen al conjunto de prueba. De esta manera, se busca construir un modelo interpretable, robusto y con capacidad de generalización, así como reflexionar sobre la importancia de la preparación de datos y la selección de variables en problemas reales de ciencia de datos.

In [19]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

# ====== Cargar datasets ======
train_path = "A1.3 Calificaciones entrenamiento.csv"
test_path  = "A1.3 Calificaciones pruebas.csv"

train_df = pd.read_csv(train_path)
test_df  = pd.read_csv(test_path)   # ← se carga pero NO se usa aún

# ====== Variable objetivo ======
target = "G3"

# ====== SOLO entrenamiento ======
X_train = train_df.drop(columns=[target])
y_train = train_df[target]
```

```
# ====== Identificar tipos de variables (solo train) ======
cat_cols = X_train.select_dtypes(include=["object"]).columns.tolist()
num_cols = X_train.select_dtypes(exclude=["object"]).columns.tolist()
```

Exploración y comprensión del conjunto de datos

Como primer paso, se realizó una exploración inicial del conjunto de datos de entrenamiento con el objetivo de comprender su estructura general, el tipo de variables disponibles y su posible relación con la calificación final. Este análisis preliminar es fundamental para identificar retos potenciales antes de construir un modelo de regresión lineal múltiple.

El conjunto de datos contiene variables de distinta naturaleza, incluyendo información académica y demográfica de los estudiantes, así como calificaciones parciales y finales. A partir de la revisión de los tipos de datos se identificaron variables cuantitativas, asociadas principalmente a valores numéricos y calificaciones, y variables cualitativas, relacionadas con características categóricas del estudiante o de su entorno académico. Esta distinción resulta clave, ya que las variables cualitativas no pueden incorporarse directamente en un modelo lineal y requieren un tratamiento específico en etapas posteriores.

Asimismo, se identificó la calificación final como la variable objetivo del análisis. La exploración de su distribución permite comprender su rango, variabilidad y comportamiento general dentro del conjunto de entrenamiento. Esta información proporciona una primera aproximación al problema de predicción y sienta las bases para las etapas posteriores de preparación, limpieza y selección de características.

In [20]:
```
print("Dimensiones del conjunto de entrenamiento:")
print(train_df.shape)

print("\nVariables disponibles:")
print(train_df.columns.tolist())

print("\nTipos de datos por variable:")
display(train_df.dtypes)
```

```
Dimensiones del conjunto de entrenamiento:
(197, 10)

Variables disponibles:
['Escuela', 'Sexo', 'Edad', 'HorasDeEstudio', 'Reprobadas', 'Internet', 'Faltas', 'G
1', 'G2', 'G3']

Tipos de datos por variable:
```

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

In [21]:
```
desc_num = train_df[num_cols + ["G3"]].describe().T
display(desc_num)
for c in cat_cols:
    print(f"\nDistribución de la variable categórica: {c}")
    display(train_df[c].value_counts())
```

	count	mean	std	min	25%	50%	75%	max
Edad	197.0	17.609137	0.960717	16.0	17.0	18.0	18.0	22.0
HorasDeEstudio	197.0	2.121827	0.823990	1.0	2.0	2.0	3.0	4.0
Reprobadas	197.0	0.304569	0.629731	0.0	0.0	0.0	0.0	3.0
Faltas	197.0	6.715736	8.740616	0.0	0.0	4.0	10.0	75.0
G1	197.0	11.106599	3.250401	3.0	9.0	11.0	13.0	19.0
G2	197.0	10.664975	3.403096	0.0	9.0	11.0	13.0	18.0
G3	197.0	10.238579	4.520806	0.0	8.0	11.0	13.0	19.0

```
Distribución de la variable categórica: Escuela
Escuela
GP    151
MS     46
Name: count, dtype: int64
Distribución de la variable categórica: Sexo
Sexo
F    115
M     82
Name: count, dtype: int64
Distribución de la variable categórica: Internet
Internet
yes    163
no      34
Name: count, dtype: int64
```

La exploración inicial del conjunto de datos de entrenamiento permitió confirmar la estructura general del conjunto de datos, así como el número de observaciones y las variables disponibles para el análisis. A partir de la revisión de los tipos de datos, se identificó claramente la presencia de variables cuantitativas y cualitativas, lo cual resulta fundamental para definir el tratamiento que recibirá cada tipo de variable en las etapas posteriores.

Asimismo, el análisis descriptivo de las variables numéricas proporcionó información sobre sus rangos, valores típicos y dispersión, mientras que el conteo de las variables categóricas permitió conocer la distribución de sus categorías dentro del conjunto de entrenamiento. Esta información preliminar facilita la identificación de posibles variables relevantes, así como de retos potenciales relacionados con escalas, codificación y redundancia de información, sentando una base sólida para la preparación y limpieza de los datos que se abordará en el siguiente apartado.

Preparación y limpieza de los datos

Una vez comprendida la estructura general del conjunto de datos, se procedió a la etapa de preparación y limpieza utilizando exclusivamente los datos de entrenamiento. Esta etapa es fundamental en problemas reales de ciencia de datos, ya que permite asegurar que la información utilizada para entrenar el modelo sea consistente y adecuada para un modelo de regresión lineal múltiple.

En primer lugar, se analizó la presencia de valores faltantes en las variables disponibles. La identificación de huecos en los datos es esencial, ya que la regresión lineal no puede manejar directamente observaciones incompletas. Dependiendo de la magnitud y ubicación de los valores faltantes, pueden tomarse distintas decisiones, como la eliminación de observaciones o la imputación de valores representativos.

Posteriormente, se revisaron las variables cualitativas identificadas en la etapa anterior. Dado que este tipo de variables no puede incorporarse directamente en un modelo lineal, se dejó establecido que deberán ser transformadas a una representación numérica adecuada en etapas posteriores del análisis.

Finalmente, se evaluó la posible presencia de valores atípicos en las variables numéricas. Los valores atípicos pueden influir de manera significativa en la estimación de los coeficientes del modelo, por lo que su detección temprana permite decidir si representan errores de medición o comportamientos válidos dentro del contexto del problema. Este análisis de limpieza y preparación sienta las bases para un modelado más estable e interpretable.

In [22]:
```
# Porcentaje de valores faltantes por variable (solo entrenamiento)
faltantes = train_df.isna().mean() * 100
faltantes_df = pd.DataFrame({
    "Variable": faltantes.index,
    "Porcentaje de valores faltantes (%)": faltantes.values
})

display(faltantes_df)

# Detección de outliers con el método de Tukey (IQR)
outliers_resumen = []

for col in num_cols + ["G3"]:
```

```
    q1 = train_df[col].quantile(0.25)
    q3 = train_df[col].quantile(0.75)
    iqr = q3 - q1
    lim_inf = q1 - 1.5 * iqr
    lim_sup = q3 + 1.5 * iqr

    n_outliers = ((train_df[col] < lim_inf) | (train_df[col] > lim_sup)).sum()

    outliers_resumen.append({
        "Variable": col,
        "Outliers detectados": n_outliers,
        "Límite inferior": lim_inf,
        "Límite superior": lim_sup
    })

outliers_df = pd.DataFrame(outliers_resumen)
display(outliers_df)

print("Variables cualitativas identificadas (entrenamiento):")
print(cat_cols)
```

	Variable	Porcentaje de valores faltantes (%)
0	Escuela	0.0
1	Sexo	0.0
2	Edad	0.0
3	HorasDeEstudio	0.0
4	Reprobadas	0.0
5	Internet	0.0
6	Faltas	0.0
7	G1	0.0
8	G2	0.0
9	G3	0.0

	Variable	Outliers detectados	Límite inferior	Límite superior
0	Edad	5	15.5	19.5
1	HorasDeEstudio	0	0.5	4.5
2	Reprobadas	46	0.0	0.0
3	Faltas	5	-15.0	25.0
4	G1	0	3.0	19.0
5	G2	4	3.0	19.0
6	G3	20	0.5	20.5

```
Variables cualitativas identificadas (entrenamiento):
['Escuela', 'Sexo', 'Internet']
```

El análisis de preparación y limpieza de los datos permitió confirmar que el conjunto de entrenamiento no presenta valores faltantes en ninguna de sus variables, lo cual facilita el análisis posterior y elimina la necesidad de aplicar técnicas de imputación o eliminación de observaciones por este motivo. Esta característica asegura que toda la información disponible pueda ser utilizada de manera consistente durante el proceso de modelado.

En cuanto a la detección de valores atípicos, se identificaron observaciones extremas en algunas variables numéricas. En particular, se detectaron valores atípicos en las variables Edad, Reprobadas, Faltas, G2 y G3, mientras que otras variables, como HorasDeEstudio y G1, no presentaron outliers bajo el criterio utilizado. La presencia de estos valores sugiere comportamientos extremos que deberán ser evaluados cuidadosamente en etapas posteriores para determinar si representan errores de medición o situaciones válidas dentro del contexto académico.

Finalmente, se confirmó la presencia de variables cualitativas en el conjunto de entrenamiento, específicamente Escuela, Sexo e Internet. Estas variables no pueden ser incorporadas directamente en un modelo de regresión lineal múltiple, por lo que requerirán un proceso de transformación adecuado antes de su inclusión en el modelo. En conjunto, los resultados de esta etapa establecen una base clara para el análisis de relaciones entre variables y la selección de características que se abordará en el siguiente apartado.

2.1 Decisiones de limpieza aplicadas

Dado que no se detectaron valores faltantes en el conjunto de entrenamiento, no fue necesario aplicar imputación ni eliminar observaciones por datos incompletos.

Respecto a los valores atípicos detectados, se tomó la decisión de conservarlos en esta etapa por dos razones. Primero, varias variables donde aparecen outliers representan fenómenos plausibles en un contexto académico (por ejemplo, número de materias reprobadas, faltas o diferencias de desempeño). Segundo, eliminar observaciones extremas sin evidencia de error de captura puede sesgar el conjunto de entrenamiento y reducir la representatividad del modelo. En consecuencia, los outliers se mantienen para que el modelo aprenda también a partir de casos extremos reales.

Finalmente, se definió el tratamiento para las variables cualitativas (Escuela, Sexo e Internet). Debido a que un modelo de regresión lineal requiere entradas numéricas, estas variables serán transformadas mediante codificación one-hot, generando variables binarias por categoría. Esta transformación se realizará utilizando únicamente el conjunto de entrenamiento y posteriormente se aplicará al conjunto de prueba para evitar fuga de información.

3. Análisis de relaciones entre variables

Una vez preparados los datos de entrenamiento, se analizó la relación entre las variables explicativas con el objetivo de identificar posibles problemas de colinealidad o redundancia de información. La colinealidad ocurre cuando dos o más variables de entrada están altamente relacionadas entre sí, lo cual dificulta la interpretación de los coeficientes de un modelo de regresión lineal múltiple y puede incrementar la incertidumbre en sus estimaciones.

Para este análisis se evaluaron las relaciones entre las variables numéricas, ya que son las que pueden presentar asociaciones lineales directas. En particular, se puso especial atención en las calificaciones parciales y su relación con la calificación final, dado que estas variables suelen medir aspectos similares del desempeño académico.

El análisis permitió identificar variables con una fuerte asociación entre sí, lo cual sugiere la presencia de información redundante. Este resultado indica que no necesariamente es conveniente conservar todas las variables originales en el modelo, ya que mantener predictores altamente correlacionados puede aumentar la complejidad del modelo sin aportar mejoras significativas en su capacidad predictiva. Los hallazgos de esta sección sirven como base para el proceso de selección de características que se desarrollará en el siguiente apartado.

In [23]:
```
# Matriz de correlación de variables numéricas (entrenamiento)
corr_matrix = train_df[num_cols + ["G3"]].corr()
display(corr_matrix)
# Identificar pares con alta correlación absoluta
umbral = 0.7
pares_colineales = []

for i in range(len(corr_matrix.columns)):
    for j in range(i):
        corr_val = corr_matrix.iloc[i, j]
        if abs(corr_val) >= umbral:
            pares_colineales.append({
                "Variable 1": corr_matrix.columns[i],
                "Variable 2": corr_matrix.columns[j],
                "Correlación": corr_val
            })

pares_colineales_df = pd.DataFrame(pares_colineales)
display(pares_colineales_df)
for col in ["G1", "G2"]:
    if col in train_df.columns:
        corr_val = train_df[[col, "G3"]].corr().iloc[0,1]
        print(f"Correlación entre {col} y G3: {corr_val:.3f}")
```

	Edad	HorasDeEstudio	Reprobadas	Faltas	G1	G2	
Edad	1.000000	0.002454	0.501370	0.130091	-0.069915	-0.107360	-0.
HorasDeEstudio	0.002454	1.000000	-0.150533	-0.116304	0.263725	0.191119	0.
Reprobadas	0.501370	-0.150533	1.000000	0.177095	-0.267695	-0.254499	-0.
Faltas	0.130091	-0.116304	0.177095	1.000000	-0.000903	-0.028947	0.
G1	-0.069915	0.263725	-0.267695	-0.000903	1.000000	0.898063	0.
G2	-0.107360	0.191119	-0.254499	-0.028947	0.898063	1.000000	0.
G3	-0.138181	0.129121	-0.247880	0.078163	0.810730	0.868786	1.

	Variable 1	Variable 2	Correlación
0	G2	G1	0.898063
1	G3	G1	0.810730
2	G3	G2	0.868786

```
Correlación entre G1 y G3: 0.811
Correlación entre G2 y G3: 0.869
```

El análisis de correlación entre las variables numéricas del conjunto de entrenamiento evidenció relaciones relevantes entre algunas de las variables explicativas. En particular, se observó una correlación muy alta entre las calificaciones parciales G1 y G2 ($\rho \approx 0.90$), lo cual indica que ambas variables capturan información muy similar sobre el desempeño académico del estudiante.

Asimismo, tanto G1 como G2 presentaron una asociación fuerte con la calificación final G3, con coeficientes de correlación aproximados de 0.81 y 0.87, respectivamente. Estos resultados confirman que las calificaciones parciales son predictores importantes del resultado final, pero también ponen de manifiesto la existencia de redundancia de información entre ellas.

Por otro lado, las variables restantes mostraron correlaciones considerablemente menores con la calificación final, lo que sugiere que su relación con G3 es más débil o indirecta. En conjunto, estos hallazgos indican la presencia de colinealidad entre algunas variables explicativas y respaldan la necesidad de aplicar un proceso de selección de características, con el fin de evitar modelos innecesariamente complejos y mejorar la interpretabilidad del modelo de regresión lineal múltiple.

4. Selección de características

Una vez analizadas las relaciones entre las variables explicativas y detectada la presencia de colinealidad, se procedió a realizar un proceso formal de selección de características. El

objetivo de esta etapa es identificar un subconjunto de variables que aporte la mayor cantidad de información posible para la predicción de la calificación final, evitando al mismo tiempo la inclusión de variables redundantes que puedan incrementar innecesariamente la complejidad del modelo.

La selección de características permite mejorar la interpretabilidad del modelo de regresión lineal múltiple y reducir la varianza de las predicciones, especialmente cuando algunas variables miden aspectos similares del fenómeno estudiado. En este trabajo se empleó un enfoque de selección basado en el desempeño del modelo, utilizando únicamente los datos de entrenamiento para evitar fuga de información.

El proceso de selección se centró en evaluar la contribución individual de las variables numéricas al modelo, considerando el impacto que tiene la inclusión o exclusión de cada predictor sobre la capacidad explicativa del modelo. Los resultados de este proceso servirán como base para definir el conjunto final de variables que se utilizarán en el entrenamiento del modelo.

In [24]:
```
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.impute import SimpleImputer
from sklearn.linear_model import LinearRegression
from sklearn.feature_selection import SequentialFeatureSelector

# Solo variables numéricas
X_train_num = train_df[num_cols].copy()
# Pipeline para selección de características
sfs_pipeline = Pipeline(steps=[
    ("imputer", SimpleImputer(strategy="median")),
    ("scaler", StandardScaler()),
    ("sfs", SequentialFeatureSelector(
        LinearRegression(),
        direction="forward",
        scoring="r2",
        cv=5
    )),
    ("model", LinearRegression())
])

# Ajustar SOLO con entrenamiento
sfs_pipeline.fit(X_train_num, y_train)

# Variables seleccionadas
mask = sfs_pipeline.named_steps["sfs"].get_support()
variables_seleccionadas = [v for v, m in zip(num_cols, mask) if m]

print("Variables numéricas seleccionadas:")
print(variables_seleccionadas)
from sklearn.metrics import r2_score, mean_squared_error

y_pred_train = sfs_pipeline.predict(X_train_num)

r2_train = r2_score(y_train, y_pred_train)
```

```
rmse_train = mean_squared_error(y_train, y_pred_train, squared=False)

print(f"R2 en entrenamiento: {r2_train:.4f}")
print(f"RMSE en entrenamiento: {rmse_train:.4f}")
```

```
Variables numéricas seleccionadas:
['Edad', 'Faltas', 'G2']
R2 en entrenamiento: 0.7689
RMSE en entrenamiento: 2.1676
```

El proceso de selección de características aplicado sobre los datos de entrenamiento permitió identificar un subconjunto reducido de variables numéricas relevantes para la predicción de la calificación final. En particular, las variables seleccionadas fueron Edad, Faltas y G2, lo cual indica que estas aportan información complementaria y no redundante al modelo.

La inclusión de la variable G2 resulta coherente con el análisis de colinealidad previo, ya que esta calificación parcial presenta una fuerte relación con la calificación final y resume de manera efectiva el desempeño académico del estudiante. Por su parte, las variables Edad y Faltas aportan información adicional relacionada con características personales y hábitos académicos, contribuyendo a mejorar la capacidad explicativa del modelo.

El modelo entrenado utilizando únicamente las variables seleccionadas alcanzó un coeficiente de determinación de aproximadamente 0.77 en el conjunto de entrenamiento, junto con un error cuadrático medio (RMSE) cercano a 2.17. Estos resultados indican que el subconjunto seleccionado logra explicar una proporción significativa de la variabilidad de la calificación final, manteniendo al mismo tiempo un modelo más simple e interpretable que aquel que utilizaría todas las variables disponibles.

5. Entrenamiento del modelo de regresión lineal múltiple

Una vez definido el subconjunto de variables relevantes mediante el proceso de selección de características, se procedió al entrenamiento del modelo de regresión lineal múltiple utilizando exclusivamente los datos de entrenamiento. En esta etapa se incorporaron tanto las variables numéricas seleccionadas como las variables cualitativas identificadas previamente, las cuales fueron transformadas a una representación numérica adecuada.

El modelo se construyó con el objetivo de capturar la relación lineal entre las variables explicativas y la calificación final, manteniendo un equilibrio entre capacidad predictiva e interpretabilidad. Para garantizar un entrenamiento consistente, las transformaciones necesarias, como la codificación de variables cualitativas y el escalado de variables numéricas, se definieron a partir del conjunto de entrenamiento.

El desempeño del modelo entrenado se evaluó inicialmente sobre los datos de entrenamiento, utilizando métricas de error y de ajuste que permiten analizar qué tan bien el

modelo logra representar el comportamiento observado en los datos disponibles. Los resultados obtenidos en esta etapa servirán como referencia para la evaluación final del modelo sobre el conjunto de prueba.

In [25]:
```
# Variables numéricas seleccionadas en el apartado 4
num_cols_final = ["Edad", "Faltas", "G2"]

# Variables cualitativas (ya identificadas)
cat_cols_final = cat_cols
from sklearn.compose import ColumnTransformer
from sklearn.preprocessing import OneHotEncoder
from sklearn.pipeline import Pipeline
from sklearn.linear_model import LinearRegression
from sklearn.impute import SimpleImputer
from sklearn.preprocessing import StandardScaler

# Transformador para variables numéricas
numeric_transformer = Pipeline(steps=[
    ("imputer", SimpleImputer(strategy="median")),
    ("scaler", StandardScaler())
])

# Transformador para variables cualitativas
categorical_transformer = OneHotEncoder(
    drop="first",           # evita colinealidad perfecta
    handle_unknown="ignore"
)

# Preprocesador completo
preprocess = ColumnTransformer(
    transformers=[
        ("num", numeric_transformer, num_cols_final),
        ("cat", categorical_transformer, cat_cols_final)
    ]
)
# Pipeline completo del modelo
modelo_final = Pipeline(steps=[
    ("preprocess", preprocess),
    ("model", LinearRegression())
])

# Entrenamiento SOLO con datos de entrenamiento
modelo_final.fit(train_df, y_train)
from sklearn.metrics import r2_score, mean_squared_error

y_pred_train = modelo_final.predict(train_df)

r2_train = r2_score(y_train, y_pred_train)
rmse_train = mean_squared_error(y_train, y_pred_train, squared=False)

# ================================
# Mostrar la ecuación del modelo
# ================================

# Extraer el modelo lineal entrenado
```

```
reg = modelo_final.named_steps["model"]

# Obtener nombres de las variables después del preprocesamiento
feature_names_num = num_cols_final

feature_names_cat = modelo_final.named_steps["preprocess"] \
    .named_transformers_["cat"] \
    .get_feature_names_out(cat_cols_final)

# Unir nombres de variables
feature_names = list(feature_names_num) + list(feature_names_cat)

# Coeficientes
coeficientes = reg.coef_
intercepto = reg.intercept_

# Crear tabla de coeficientes
coef_df = pd.DataFrame({
    "Variable": feature_names,
    "Coeficiente": coeficientes
})

display(coef_df)

# Imprimir ecuación en texto
ecuacion = f"G3 = {intercepto:.3f}"
for var, coef in zip(feature_names, coeficientes):
    signo = "+" if coef >= 0 else "-"
    ecuacion += f" {signo} {abs(coef):.3f}·{var}"

print("\nEcuación del modelo:")
print(ecuacion)

print(f"R2 en entrenamiento: {r2_train:.4f}")
print(f"RMSE en entrenamiento: {rmse_train:.4f}")
```

	Variable	Coeficiente
0	Edad	-0.363552
1	Faltas	0.558742
2	G2	3.909950
3	Escuela_MS	0.647385
4	Sexo_M	0.483293
5	Internet_yes	0.151318

```
Ecuación del modelo:
G3 = 9.761 - 0.364·Edad + 0.559·Faltas + 3.910·G2 + 0.647·Escuela_MS + 0.483·Sexo_M
+ 0.151·Internet_yes
R2 en entrenamiento: 0.7751
RMSE en entrenamiento: 2.1383
```

El modelo de regresión lineal múltiple fue entrenado utilizando el conjunto final de variables explicativas definido en los apartados anteriores, incorporando tanto las variables numéricas seleccionadas como las variables cualitativas transformadas a una representación binaria. De esta manera, el modelo considera simultáneamente información académica cuantitativa y características categóricas del estudiante.

La ecuación final del modelo muestra explícitamente la contribución de cada variable al valor estimado de la calificación final. En particular, se observa la presencia de coeficientes asociados a las variables numéricas Edad, Faltas y G2, así como coeficientes correspondientes a las variables binarias derivadas de las variables cualitativas Escuela, Sexo e Internet. Esto confirma que dichas variables fueron efectivamente incorporadas en el proceso de entrenamiento y que su efecto es considerado de forma explícita en la predicción.

El desempeño del modelo sobre los datos de entrenamiento indica que el conjunto de variables seleccionado es capaz de explicar una proporción significativa de la variabilidad de la calificación final, manteniendo al mismo tiempo una estructura relativamente simple e interpretable. Estos resultados establecen una base sólida para evaluar la capacidad de generalización del modelo utilizando el conjunto de datos de prueba, lo cual se abordará en el siguiente apartado.

6. Evaluación del modelo en datos de prueba

Una vez entrenado el modelo de regresión lineal múltiple utilizando únicamente los datos de entrenamiento, se procedió a evaluar su desempeño sobre el conjunto de datos de prueba. Esta evaluación permite analizar la capacidad de generalización del modelo, es decir, su habilidad para realizar predicciones adecuadas sobre información no utilizada durante el proceso de entrenamiento.

El conjunto de prueba se mantuvo completamente separado durante las etapas previas de exploración, limpieza, selección de características y entrenamiento, con el fin de evitar cualquier tipo de fuga de información. De esta manera, las métricas obtenidas en esta etapa reflejan un escenario más realista del desempeño esperado del modelo.

El desempeño del modelo se evaluó mediante métricas cuantitativas que permiten comparar los valores predichos con las calificaciones reales. Asimismo, se analizó el comportamiento de los errores de predicción para identificar posibles patrones que indiquen sesgos o limitaciones del modelo. Los resultados obtenidos en esta sección serán comparados con el desempeño observado en el conjunto de entrenamiento para evaluar la estabilidad del modelo.

In [26]:
```
# Variable objetivo en prueba
y_test = test_df["G3"]
# Predicciones del modelo en prueba
```

```
y_pred_test = modelo_final.predict(test_df)
from sklearn.metrics import r2_score, mean_squared_error, mean_absolute_error

r2_test = r2_score(y_test, y_pred_test)
rmse_test = mean_squared_error(y_test, y_pred_test, squared=False)
mae_test = mean_absolute_error(y_test, y_pred_test)

print(f"R2 en prueba: {r2_test:.4f}")
print(f"RMSE en prueba: {rmse_test:.4f}")
print(f"MAE en prueba: {mae_test:.4f}")
# Residuales
residuales = y_test - y_pred_test

print("Resumen de residuales:")
display(residuales.describe())
```

```
R2 en prueba: 0.8580
RMSE en prueba: 1.7463
MAE en prueba: 1.2097
Resumen de residuales:
count    198.000000
mean      -0.227990
std        1.735765
min       -9.037287
25%       -1.069410
50%       -0.137074
75%        0.662595
max        3.592727
Name: G3, dtype: float64
```

La evaluación del modelo sobre el conjunto de datos de prueba mostró un desempeño sólido y consistente con los resultados obtenidos durante el entrenamiento. El coeficiente de determinación alcanzado en prueba fue aproximadamente 0.86, lo que indica que el modelo es capaz de explicar una proporción elevada de la variabilidad observada en la calificación final de los estudiantes no utilizados durante el ajuste.

En términos de error, el modelo presentó un error cuadrático medio (RMSE) cercano a 1.75 y un error absoluto medio (MAE) alrededor de 1.21 en el conjunto de prueba. Estos valores sugieren que, en promedio, las predicciones del modelo difieren en poco más de un punto respecto a la calificación real, lo cual resulta razonable considerando la escala de la variable objetivo.

El análisis de los residuales mostró una distribución centrada cerca de cero, con una mediana ligeramente negativa y una dispersión moderada. Si bien se observaron algunos errores extremos, la mayoría de los residuales se concentraron dentro de un rango acotado, lo que indica que el modelo no presenta sesgos sistemáticos significativos y mantiene un comportamiento estable en la mayoría de los casos.

En conjunto, estos resultados evidencian que el modelo de regresión lineal múltiple presenta una buena capacidad de generalización y que el proceso de preparación de datos, selección

de características e inclusión de variables cualitativas permitió construir un modelo robusto y confiable para la predicción de la calificación final.

7. Reflexión y conclusiones finales

En este trabajo se desarrolló un modelo de regresión lineal múltiple para la predicción de la calificación final de estudiantes, abordando de manera explícita los retos asociados al uso de datos reales. A lo largo del análisis se llevó a cabo una exploración cuidadosa del conjunto de datos, la preparación y limpieza de la información, el análisis de relaciones entre variables y un proceso formal de selección de características, todo ello utilizando exclusivamente los datos de entrenamiento para evitar fuga de información.

El análisis exploratorio permitió identificar la presencia de variables cuantitativas y cualitativas, así como la ausencia de valores faltantes en el conjunto de entrenamiento. Asimismo, se detectaron valores atípicos en diversas variables numéricas, los cuales fueron conservados al considerarse representativos de comportamientos reales dentro del contexto académico. Este enfoque permitió mantener la diversidad de casos en los datos y evitar sesgos derivados de la eliminación arbitraria de observaciones.

El estudio de colinealidad evidenció una fuerte asociación entre las calificaciones parciales, particularmente entre G1 y G2, así como entre estas y la calificación final G3. Con base en este análisis, se aplicó un proceso de selección de características que permitió reducir el conjunto de variables explicativas a un subconjunto más compacto e informativo. Las variables numéricas seleccionadas fueron Edad, Faltas y G2, las cuales capturan tanto el desempeño académico como aspectos personales y de comportamiento del estudiante. Adicionalmente, se incorporaron variables cualitativas transformadas a una representación binaria, lo que permitió integrar información categórica al modelo de forma adecuada.

El modelo entrenado con este conjunto final de variables mostró un buen desempeño tanto en el conjunto de entrenamiento como en el conjunto de prueba. En particular, el modelo alcanzó un coeficiente de determinación cercano a 0.77 en entrenamiento y aproximadamente 0.86 en prueba, acompañado de errores promedio relativamente bajos. Estos resultados indican que el modelo no solo logra ajustar adecuadamente los datos de entrenamiento, sino que también presenta una buena capacidad de generalización, lo cual refleja la efectividad del proceso de selección de características y de la preparación de los datos.

A pesar de los buenos resultados obtenidos, el modelo presenta algunas limitaciones inherentes a la regresión lineal múltiple. Este tipo de modelo asume relaciones lineales entre las variables explicativas y la variable objetivo, y puede no capturar interacciones complejas o comportamientos no lineales presentes en los datos. Como trabajo futuro, sería interesante explorar modelos no lineales, como árboles de decisión o métodos de ensamble, así como la inclusión de términos de interacción o técnicas de regularización, con el fin de comparar su desempeño y evaluar posibles mejoras en la predicción.

En conclusión, este análisis pone de manifiesto la importancia de la preparación de datos y la selección adecuada de características en problemas reales de ciencia de datos. Más allá del uso de un algoritmo específico, los resultados obtenidos demuestran que un enfoque sistemático y bien fundamentado puede conducir a modelos predictivos robustos, interpretables y con buen desempeño en escenarios reales.