Universidad de El Salvador
Facultad de Ciencias Naturales y Matemática
Maestría en Estadística y Ciencia de Datos
Inferencia Estadística y Regresión

Informe del Proyecto Final: Análisis y Predicción de Precios de Viviendas

Presentado por:
1. Juan Jose Moreno Ramirez
2. Víctor Mauricio Ochoa García
3. Salvador Enrique Rodríguez Hernández

Fecha de entrega: 08 de diciembre de 2024

1. Introducción

El mercado inmobiliario es un sector crucial en la economía de cualquier región, dado que la valoración de propiedades afecta decisiones importantes para compradores, vendedores y desarrolladores. Este estudio tiene como propósito analizar el conjunto de datos "House Prices: Advanced Regression Techniques," con el objetivo de identificar los factores que más influyen en los precios de las viviendas y construir un modelo predictivo robusto que permita realizar estimaciones precisas.

A través del uso de técnicas estadísticas inferenciales, se busca explorar la relación entre características físicas, contextuales y económicas de las propiedades con el precio de venta. Este análisis no solo contribuye al entendimiento del mercado inmobiliario, sino que también aporta herramientas prácticas para la toma de decisiones en este ámbito.

2. Objetivos

Objetivo general

Desarrollar un análisis estadístico inferencial que permita identificar los factores determinantes en el precio de las viviendas y construir un modelo predictivo basado en el conjunto de datos “House Prices: Advanced Regression Techniques”.

Objetivos especificos

(1) Realizar un análisis descriptivo de las variables disponibles en el conjunto de datos para explorar su distribución y relación con el precio de venta.
(2) Identificar y evaluar problemas de multicolinealidad entre las variables predictoras y establecer su relevancia estadística.
(3) Ajustar un modelo de regresión lineal múltiple y seleccionar el modelo óptimo utilizando métricas como AIC, R^2, MAE y $RMSE$.
(4) Verificar los supuestos del modelo mediante análisis de residuos y pruebas diagnósticas para garantizar su validez.

(5) Interpretar los resultados obtenidos y proporcionar conclusiones sobre el impacto de las características clave en los precios de las viviendas.

3. Análisis inicial de los datos

En términos de factibilidad, se decidió limitar el análisis a un conjunto reducido de variables seleccionadas, en lugar de incluir las 80 disponibles en el conjunto de datos. Esta decisión se fundamentó en revisiones de literatura académica que identifican consistentemente a variables como el área habitable *GrLivArea* y la calidad general de la construcción *OverallQual* como factores clave en la determinación del precio de las propiedades debido a su relación directa con el valor percibido por los compradores (Springer, 2007). Asimismo, factores relacionados con la ubicación, como *Neighborhood* y *MSZoning*, son reconocidos como críticos por su influencia en la deseabilidad y proximidad a servicios clave (Fondo Monetario Internacional, 2018). De manera similar, características estructurales como *LotArea* (tamaño del lote), *GarageArea*, *TotalBsmtSF* y el año de construcción (YearBuilt}) han sido identificadas como predictores significativos que reflejan la modernidad y el estado de las propiedades (Springer, 2007).

En el proceso de análisis de los datos, se verificó que no existían valores faltantes en las variables seleccionadas. Este paso fue fundamental, ya que garantizó que no fuera necesario realizar imputación de datos, eliminar registros incompletos o aplicar transformaciones adicionales para manejar datos ausentes. A continuación, se presentan dos tablas con estadísticas descriptivas que resumen las características principales de las variables seleccionadas.

La **Tabla 1** incluye estadísticas descriptivas clave para las variables numéricas, como el mínimo, el máximo, la mediana, la media y los cuartiles. Por ejemplo, el precio de venta de las propiedades (*SalePrice*) varía entre $34,900 y $755,000, con un promedio de $180,921.2. Del mismo modo, otras variables como el área habitable sobre el suelo (*GrLivArea*), el tamaño del lote (*LotArea*), y el área total del sótano (*TotalBsmtSF*) también se resumen, proporcionando información sobre su dispersión y tendencias centrales. Estas estadísticas permiten identificar rangos, distribuciones y posibles valores atípicos en el conjunto de datos.

La **Tabla 2**, por otro lado, detalla la distribución de las variables categóricas mediante frecuencias absolutas y porcentajes relativos. Por ejemplo, la variable *MSZoning*, que clasifica las propiedades según la zonificación, muestra que la mayoría de las propiedades están en la categoría *RL* (Residencial de Baja Densidad), con un 78.2%, seguida de *RM* (Residencial de Media Densidad) con un 14.8%. Asimismo, *Neighborhood* revela la distribución de las propiedades entre diferentes vecindarios, siendo NAmes el más representado con un 15.3% de las propiedades.

En conjunto, estas tablas proporcionan una visión general clara y detallada de las características principales de las variables, lo que facilita el análisis posterior y permite identificar patrones relevantes en los datos.

Tabla 1 Resumen de las principales medidas descriptivas para las variables numéricas.

Variable	Mínimo	1er Cuartil	Mediana	Media	3er Cuartil	Máximo
SalePrice	34900	129975	163000	180921.2	214000	755000
GrLivArea	334	1126	1464	1515.46	1776	5642
LotArea	1300	7553	9478	10516.83	11601	215245
OverallQual	1	5	6	6.1	7	10
GarageArea	0	334	480	472.98	576	1418
TotalBsmtSF	0	795	991	1057.49	1298	6110
YearBuilt	1872	1954	1973	1971.27	2000	2010

Tabla 2 Distribución de frecuencias absolutas y porcentajes relativos para las variables categóricas.

Variable	Categoría	Frecuencia	Porcentaje
MSZoning	RL	1151	78.20%
	RM	218	14.80%
	FV	65	4.40%
	RH	16	1.10%
	C (all)	10	0.70%
Neighborhood	NAmes	225	15.30%
	CollgCr	150	10.20%
	OldTown	113	7.70%
	Edwards	100	6.80%
	Somerst	86	5.90%
	... (otras)	...	...

A continuación, se presentan las gráficas que ilustran la distribución de las variables seleccionadas de forma individual. Estas gráficas permiten analizar la forma, simetría y dispersión de cada variable, proporcionando una identificación de posibles patrones en el conjunto de datos.

(1) Distribución de *GarageArea*

El histograma de *GarageArea* muestra cuánto espacio se destina típicamente a los garajes en pies cuadrados. La mayoría de las propiedades tienen garajes que oscilan entre 200 y 600 pies cuadrados, lo cual es común para garajes de uno o dos vehículos. Existen algunos valores atípicos con áreas de garaje significativamente más grandes, que podrían representar propiedades con garajes separados más grandes o múltiples espacios de estacionamiento.

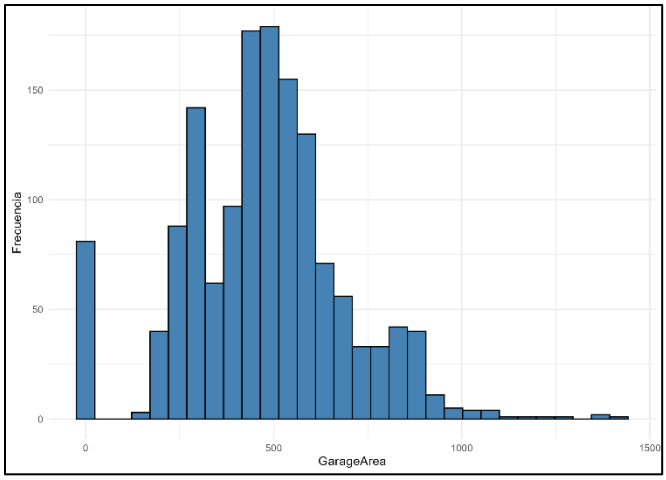

Ilustración 1 Distribución de los tamaños de garajes en pies.

(2) Distribución de *GrLivArea*

El histograma de *GrLivArea* refleja la distribución de las áreas habitables sobre el suelo en pies cuadrados. La mayoría de las viviendas tienen áreas habitables entre 1,000 y 2,000 pies cuadrados, lo que corresponde a hogares de tamaño familiar promedio. Algunas propiedades tienen áreas mucho más grandes, lo que indica casas de lujo o construcciones personalizadas. La distribución sesgada hacia la derecha sugiere que, aunque la mayoría de las viviendas están en un rango modesto, unas pocas son excepcionalmente grandes.

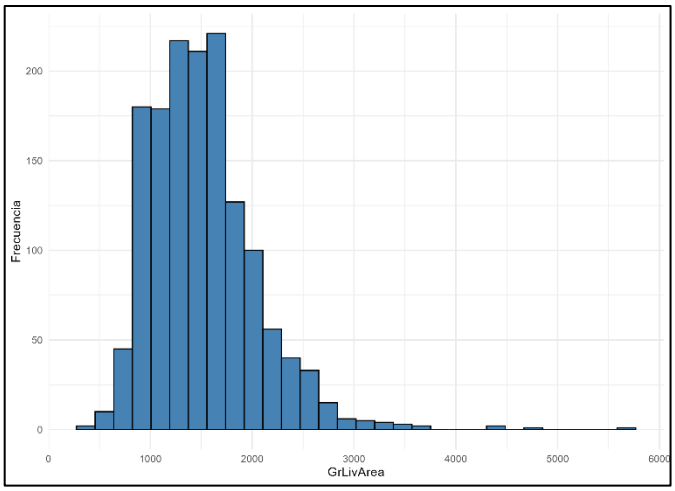

Ilustración 2 Distribución de las áreas habitables sobre el suelo en pies cuadrados.

(3) Distribución de *LotArea*

El histograma de *LotArea* revela la distribución de los tamaños de los lotes en pies cuadrados. La mayoría de las propiedades tienen lotes de menos de 20,000 pies cuadrados, lo que probablemente corresponde a terrenos residenciales típicos. Sin embargo, la cola larga de la distribución indica algunas propiedades con lotes mucho más grandes, que podrían representar fincas, terrenos rurales o casas con extensas áreas de terreno.

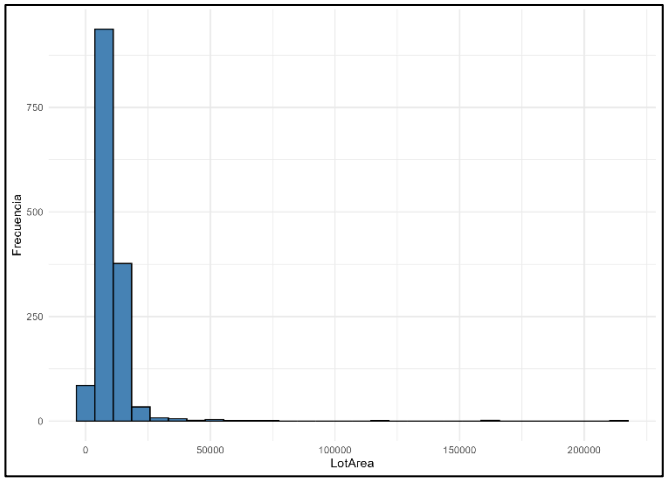

Ilustración 3 Distribución de los tamaños de los lotes residenciales en pies cuadrados.

(4) Distribución de *OverallQual*

El histograma de *OverallQual* representa la calificación general de calidad de las viviendas, basada en factores como los materiales de construcción y la calidad de los acabados. Esta variable varía de 1 a 10, donde los valores más altos indican mejor calidad. La distribución se concentra alrededor de los valores de 5 a 7, lo que sugiere que la mayoría de las casas tienen una calidad de construcción promedio o ligeramente superior al promedio.

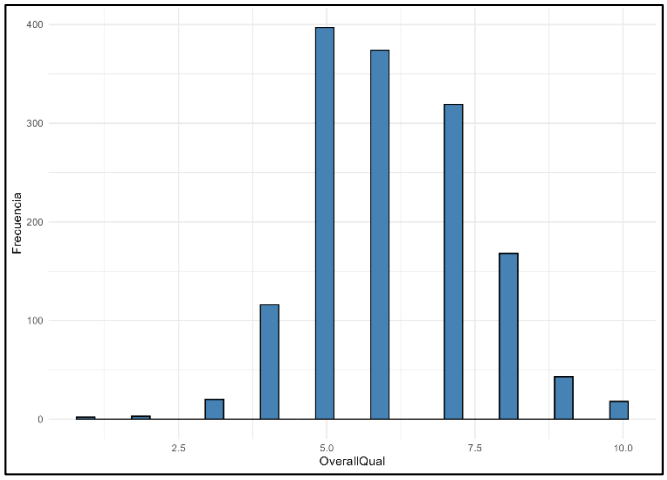

Ilustración 4 Calificación general de calidad de las viviendas en una escala de 1 a 10.

(5) Distribución de *SalePrice*

El histograma de *SalePrice* muestra la distribución de los precios de venta de las viviendas. La mayoría de las casas tienen precios inferiores a $200,000, con una disminución gradual en la frecuencia a medida que aumentan los precios. La naturaleza sesgada hacia la derecha del histograma indica que, aunque la mayoría de las casas están en un rango asequible, hay un número reducido de propiedades de lujo con precios más altos que amplían la cola de la distribución.

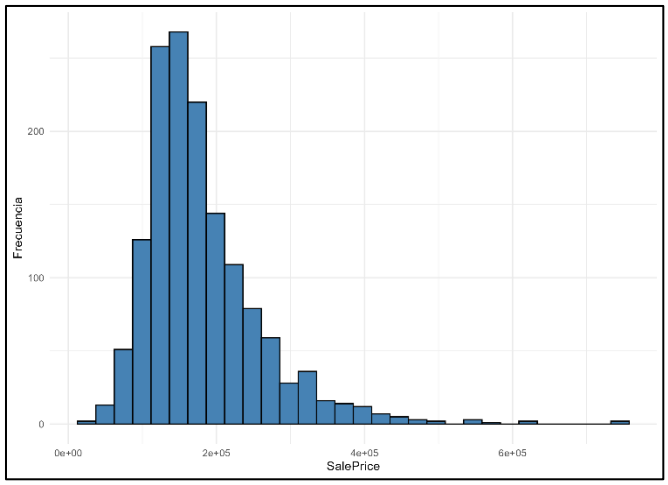

Ilustración 5 Distribución de los precios de venta de las propiedades.

(6) Distribución de *TotalBsmtSF*

El histograma de *TotalBsmtSF* muestra el área total de los sótanos en pies cuadrados. Los datos revelan un amplio rango, con la mayoría de las viviendas con sótanos que van de 0 a 1,000 pies cuadrados. La barra en 0 probablemente representa viviendas sin sótanos, mientras que la cola de la distribución incluye casas con sótanos muy amplios.

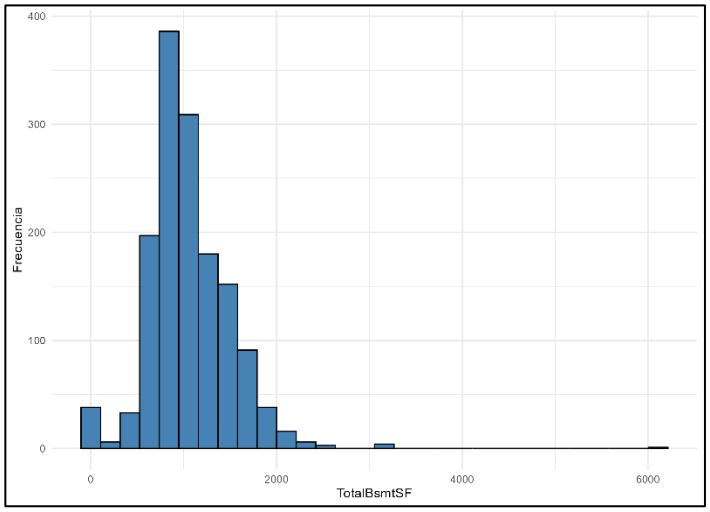

Ilustración 6 Distribución de las áreas totales de sótanos en pies cuadrados.

(7) Distribución de *YearBuilt*

El histograma de *YearBuilt* proporciona información sobre el año de construcción de las viviendas. La distribución está sesgada hacia casas más nuevas, con un pico notable para aquellas construidas entre las décadas de 1950 y 2000. Esto refleja un patrón de aumento en la construcción residencial en épocas más modernas, probablemente debido al crecimiento urbano y poblacional.

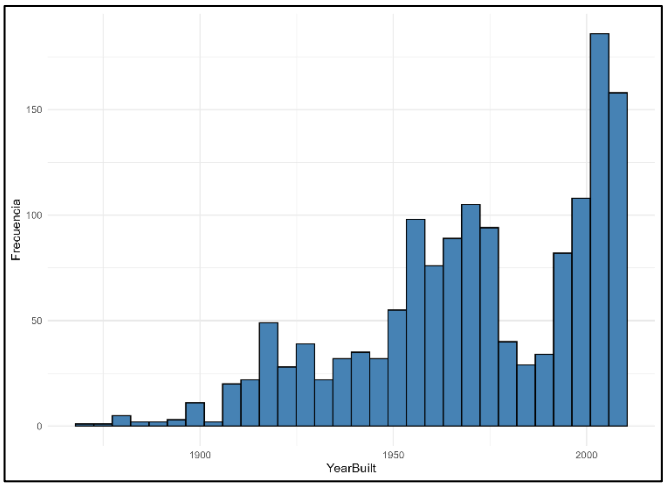

Ilustración 7 Distribución de los años de construcción de las viviendas

(8) Frecuencia de *MSZoning*

El gráfico de barras de *MSZoning* muestra la distribución de las clasificaciones de zonificación para las propiedades. La categoría de zonificación más común es *RL* (Residencial de Baja Densidad), seguida por *RM* (Residencial de Media Densidad). Otras categorías de zonificación, como *FV* (Residencial de Villa Flotante), son mucho menos comunes. Esto indica que la mayoría de las propiedades en el conjunto de datos son hogares residenciales típicos de baja densidad.

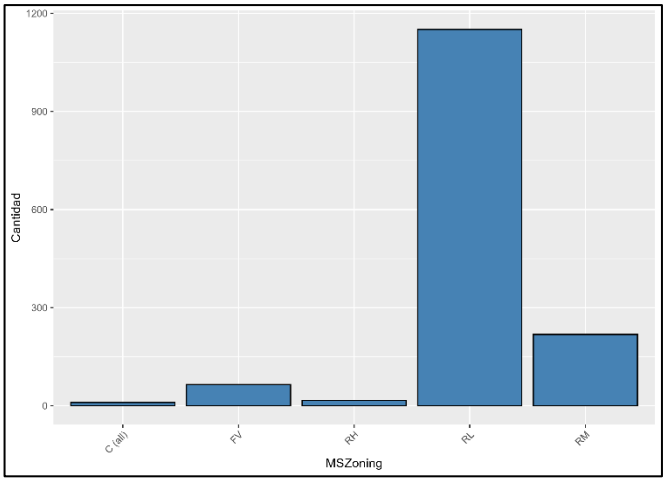

Ilustración 8 Frecuencia de las clasificaciones de zonificación residencial.

(9) Frecuencia de *Neighborhood*

El gráfico de barras de *Neighborhood* muestra la cantidad de viviendas en cada vecindario. Algunos vecindarios, como *NAmes*, *CollgCr* y *OldTown*, tienen una mayor cantidad de viviendas, lo que indica áreas residenciales más densas o un mayor número de datos recopilados de esos lugares. Otros vecindarios tienen menos viviendas, lo que podría reflejar áreas más pequeñas o menos representadas.

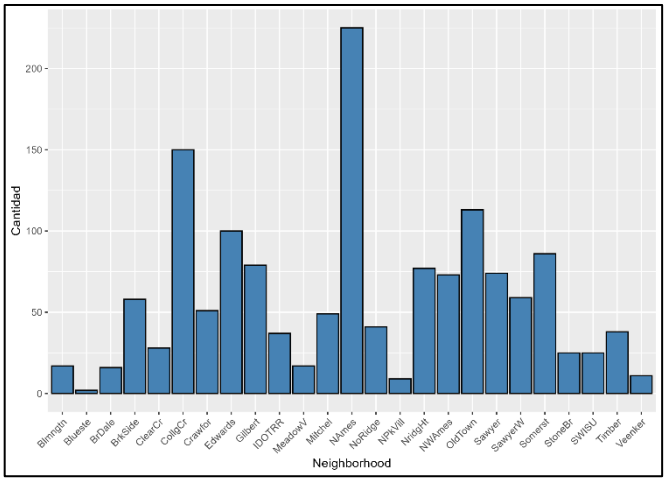

Ilustración 9 Frecuencia de las propiedades en cada vecindario

4. Análisis de Correlación y Evaluación de Multicolinealidad

En el análisis de correlación, se calcularon las relaciones lineales entre las variables numéricas seleccionadas del conjunto de datos para identificar aquellas que presentaran asociaciones significativas. Los resultados del análisis, incluyendo la matriz de correlación completa, se presentan en el Apéndice debido a su carácter exploratorio. Tal como se puede ver en la Tabla 3, las dos variables con correlaciones mayores o iguales a 0.7 con respecto al precio de venta $SalePrice$ son el área habitable sobre nivel del suelo *GrLivArea* con un coeficiente de 0.71, y la calidad general de la construcción *OverallQual* con un coeficiente de 0.79.

Tabla 3 Correlaciones Significativas (>|0.7|)

Variable 1	Variable 2	Correlation
GrLivArea	SalePrice	0.71
OverallQual	SalePrice	0.79

Para visualizar estas relaciones, se generaron gráficas específicas para cada par significativo. En particular, la relación entre $GrLivArea$ y $SalePrice$ se representa mediante el diagrama de dispersión con una línea de tendencia ajustada de la ilustración 10, mientras que la relación entre *OverallQual* y *SalePrice* se muestra como un diagrama de cajas en la ilustración 11.

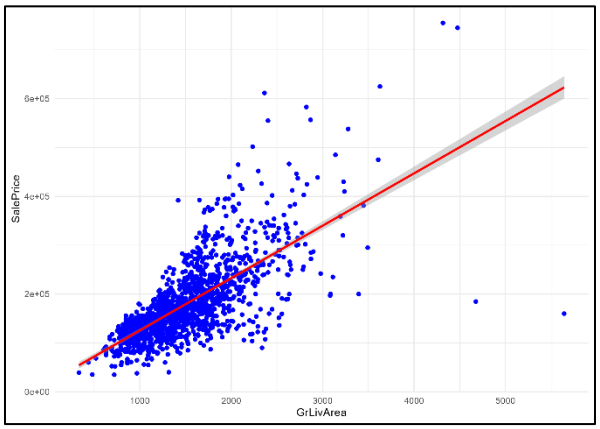

Ilustración 10 Relación entre GrLivArea y SalePrice

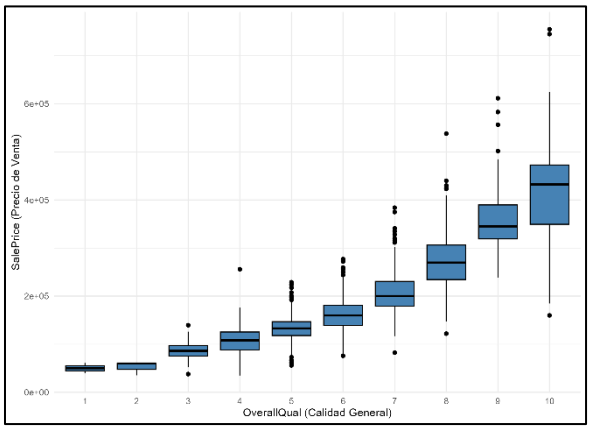

Ilustración 11 Relación entre OverallQual y SalePrice

Finalmente, se realizó una evaluación de multicolinealidad utilizando el Factor de Inflación de la Varianza (VIF). La Tabla 4 resume estos resultados, mostrando que no se detectaron problemas significativos de multicolinealidad, ya que todos los valores de VIF son menores a 10.

Tabla 4 Valores de VIF para Detectar Multicolinealidad

Variable	VIF
GrLivArea	1.86
LotArea	1.13
OverallQual	2.52
GarageArea	1.74
TotalBsmtSF	1.63
YearBuilt	1.72

En conclusión, el análisis confirmó la relevancia de variables clave para el modelado y la ausencia de problemas significativos de multicolinealidad, sentando las bases para un modelo robusto.

5. Análisis de Modelos de Regresión

El análisis de regresión tuvo como objetivo modelar la relación entre *SalePrice* y las variables independientes *GrLivArea* y *OverallQual*, evaluando su capacidad predictiva y cumpliendo los supuestos estadísticos básicos. Inicialmente, se construyeron tres modelos: un modelo completo que incluye ambas variables independientes, y dos modelos simplificados que utilizan únicamente una de las variables independientes. Estos modelos fueron evaluados con base en el Criterio de Información de Akaike (AIC) y el R^2 ajustado, indicadores que permiten comparar la calidad del ajuste y la complejidad de los modelos.

El modelo completo, que incluye $GrLivArea$ y $OverallQual$, mostró un AIC de 35,267.58 y un R^2 ajustado de 0.71, lo que indica un buen equilibrio entre ajuste y complejidad. En contraste, el modelo con solo $GrLivArea$ presentó un AIC más alto de 36,075.76 y un R^2 ajustado de 0.50, evidenciando una capacidad predictiva menor. Por otro lado, el modelo con solo $OverallQual$ obtuvo un AIC de 35,659.49 y un R^2 ajustado de 0.63, demostrando un rendimiento intermedio. Los resultados cuantitativos de esta comparación se presentan en la Tabla 5.

Tabla 5 Comparación de modelos por R^2 ajustado y AIC

Modelo	R^2 Ajustado	AIC
Completo	0.71	35267.58
Solo GrLivArea	0.50	36075.76
Solo OverallQual	0.63	35659.49

Además de los resultados numéricos, se realizaron diagnósticos gráficos de los tres modelos para evaluar el cumplimiento de los supuestos de linealidad, homocedasticidad e independencia de los errores. aunque los tres modelos presentan diferencias visuales en los gráficos de diagnóstico, todos comparten problemas similares en términos de heterocedasticidad y linealidad. En particular, los gráficos *Residuals vs Fitted* de los tres modelos evidencian una dispersión de los residuos no constante, lo cual sugiere la presencia de heterocedasticidad. Además, en los gráficos *Q-Q Residuals*, se observa una desviación de los puntos respecto a la línea teórica, lo que indica posibles violaciones a la normalidad de los residuos. Por último, los gráficos *Scale-Location* confirman una tendencia ascendente en la varianza, reforzando la detección de heterocedasticidad.

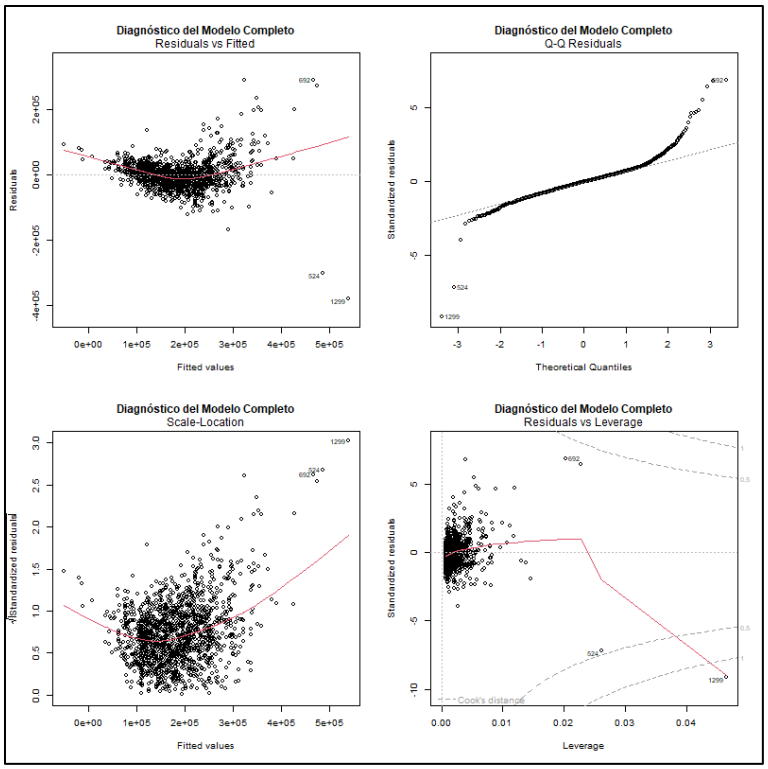

Ilustración 12 Diagnóstico del modelo completo.

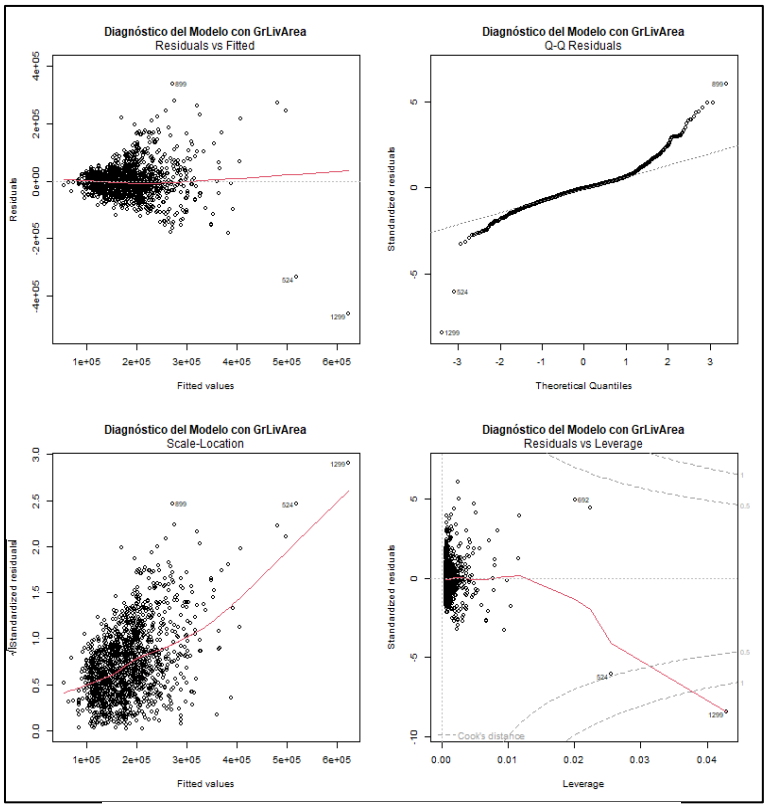

Ilustración 13 Diagnóstico del modelo con

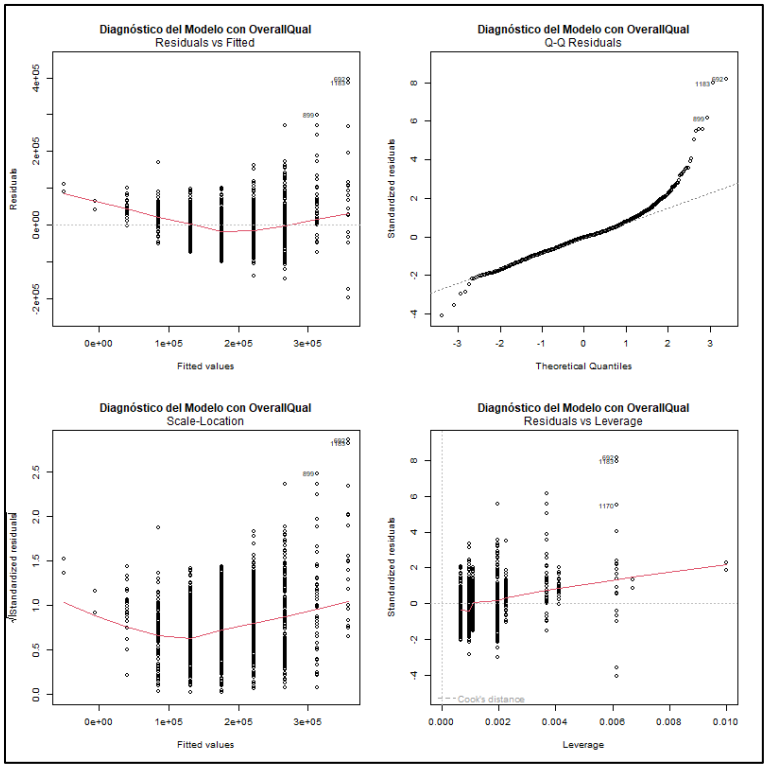

Ilustración 14 Diagnóstico del modelo con OverallQual.

El análisis comparativo destacó al modelo completo como el más adecuado debido a su mejor rendimiento en términos de AIC y R^2 ajustado. Sin embargo, los diagnósticos revelaron problemas menores de no linealidad y heterocedasticidad, que podrían afectar la validez del modelo.

Para abordar los problemas detectados en los diagnósticos del modelo completo, como la falta de linealidad y la heteroscedasticidad, se implementaron transformaciones logarítmicas en las variables *SalePrice* y *GrLivArea*. La transformación se aplicó únicamente a *GrLivArea*, debido a su naturaleza numérica y continua, lo cual permite estabilizar varianzas y mejorar la linealidad en las relaciones con la variable dependiente. Por otro lado, *OverallQual*, al ser una variable ordinal que mide la calidad general de la propiedad, ya representa adecuadamente su impacto sobre el precio de venta sin necesidad de transformaciones adicionales.

El modelo transformado mostró una mejora significativa respecto al modelo completo original, aumentando el R^2 ajustado de 0.71 a 0.75 y reduciendo el AIC de 35267.58 a -562.10. Estos resultados reflejan una mayor capacidad explicativa y eficiencia del modelo transformado para balancear la complejidad con el ajuste a los datos. Las gráficas de diagnóstico del modelo transformado evidencian mejoras claras en la linealidad y la homocedasticidad de los residuos. En comparación con el modelo completo, estas gráficas confirman que las transformaciones logarítmicas no solo optimizaron el ajuste del modelo, sino también minimizaron patrones problemáticos observados previamente en las gráficas de residuos. Dichas gráficas se presentan en la ilustración 15:

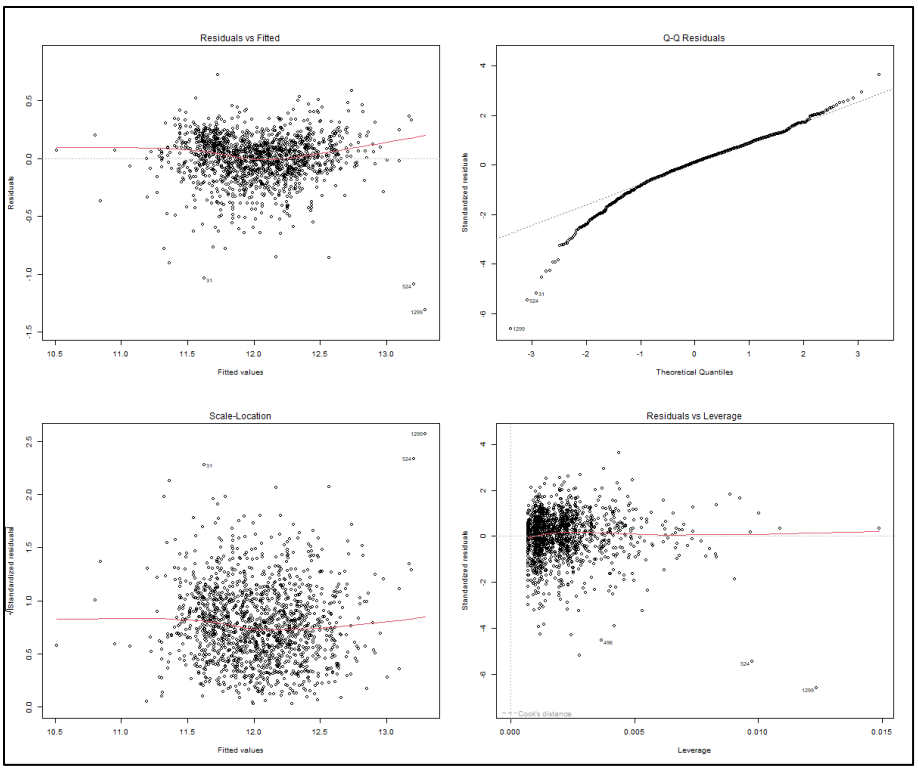

Ilustración 15 Diagnóstico del Modelo Transformado con Transformaciones Logarítmicas.

En conclusión, el modelo completo inicial fue importante para identificar la importancia relativa de ambas variables, pero el modelo transformado ofreció un ajuste más robusto y estadísticamente adecuado. Este proceso permitió obtener un modelo final que combina precisión predictiva y cumplimiento de los supuestos estadísticos esenciales.

6. Evaluación del modelo

El análisis del desempeño del modelo se llevó a cabo mediante el cálculo de métricas clave en los conjuntos de datos de entrenamiento y prueba. Estas métricas incluyeron el Error Absoluto Medio (MAE), la Raíz del Error Cuadrático Medio ($RMSE$), el Error Porcentual Absoluto Medio ($MAPE$) y el coeficiente de determinación ajustado (R^2). Estas métricas proporcionan una evaluación completa sobre la precisión y la capacidad del modelo para generalizar a nuevos datos.

En el conjunto de entrenamiento, el modelo transformado presentó un MAE de 0.15, lo que indica un error promedio reducido entre los valores predichos y los valores observados, mientras que el $RMSE$ fue de 0.21, evidenciando un buen desempeño en términos de precisión general. Por otro lado, el MAPE obtenido fue de 0.01, lo que representa un bajo porcentaje de error promedio respecto a los valores observados, y el R^2 alcanzó un valor de 0.74, sugiriendo que el modelo explica el 74% de la variabilidad de los datos de entrenamiento.

En el conjunto de prueba, los resultados fueron consistentes con los obtenidos en el entrenamiento, con un MAE de 0.15 y un $RMSE$ de 0.19, valores que confirman la estabilidad del modelo. El $MAPE$ fue de 0.01, mientras que el R^2 fue de 0.78, indicando que el modelo mantiene una alta capacidad de explicación y predicción en datos no utilizados durante el ajuste.

Estos resultados destacan la robustez del modelo transformado, evidenciando una baja discrepancia entre los conjuntos de entrenamiento y prueba. La validación del modelo demuestra su capacidad de generalizar adecuadamente y mantener un alto nivel de precisión, lo cual es esencial para aplicaciones predictivas confiables.

7. Interpretación y conclusiones

Los resultados obtenidos del modelo final proporcionan un mejor entendimiento de los factores más relevantes que influyen en el precio de venta de las propiedades. La variable $OverallQual$, que refleja la calidad general del inmueble, se identificó como el principal determinante del precio. Esto subraya que propiedades con mejores acabados, materiales de mayor calidad y una percepción superior de valor tienden a alcanzar precios significativamente más altos. De hecho, los resultados muestran que un incremento en un punto de esta variable está asociado con un aumento considerable en el precio de venta, incluso después de aplicar transformaciones logarítmicas para optimizar el modelo.

Por otro lado, GrLivArea, que mide el área habitable sobre el nivel del suelo, también tiene un impacto positivo y significativo en el precio de las propiedades. Este hallazgo refleja que, en general, los compradores valoran los espacios amplios y están dispuestos a pagar un precio mayor por ellos. No obstante, su efecto en el precio es menor en comparación con OverallQual, lo que resalta la importancia de la percepción de calidad más allá del tamaño físico del inmueble.

Las métricas calculadas para el modelo transformado, incluyendo R^2 ajustado, *MAE*, *RMSE* y *MAPE*, indicaron un excelente ajuste y capacidad predictiva. Esto sugiere que el modelo no solo explica una alta proporción de la variabilidad en los precios de venta, sino que también es capaz de generalizar adecuadamente a datos no utilizados durante el entrenamiento. La consistencia observada entre los conjuntos de entrenamiento y prueba refuerza esta conclusión.

En un contexto práctico, estos hallazgos tienen implicaciones significativas para el mercado inmobiliario. Los desarrolladores y agentes inmobiliarios pueden enfocar sus estrategias de inversión y promoción en mejorar los aspectos relacionados con la calidad percibida de las propiedades, dado su fuerte impacto en el precio. Además, la relación positiva entre $GrLivArea$ y el precio destaca la importancia de diseñar viviendas con espacios habitables bien distribuidos para maximizar el valor percibido por los compradores.

En conclusión, el modelo final no solo ofrece una herramienta robusta para la predicción de precios, sino que también proporciona información clave para la toma de decisiones en el ámbito inmobiliario. Estas conclusiones pueden ser utilizadas para guiar tanto a los profesionales del sector como a los compradores, optimizando la valoración y selección de propiedades en función de las características que realmente aportan valor.

Referencias

- Fondo Monetario Internacional (2018). *Fundamental Drivers of House Prices in Advanced Economies*. Disponible en: IMF Publications.
- Springer (2007). *Determinants of House Prices: A Quantile Regression Approach*. Disponible en: https://link.springer.com/article/10.1007/s11146-007-9053-7.
- Springer (2013). *House Price Determinants: Fundamentals and Underlying Factors*. Disponible en: https://link.springer.com/article/10.1057/ces.2013.3.

Apéndice

(1) Código R Completo

Script 1: Analisis_Inicial_Limpieza_Datos

```
# Cargar las librerías necesarias
library(dplyr)
library(ggplot2)

# Paso 1: Cargar el conjunto de datos
# Reemplazar 'house_prices.csv' con el nombre o ruta del archivo
datos <- read.csv("house_prices.csv", stringsAsFactors = FALSE)

# Paso 2: Seleccionar las variables clave justificadas en el
análisis teórico
variables_clave <- c("SalePrice", "Neighborhood", "MSZoning",
"GrLivArea", "LotArea",
                     "OverallQual", "GarageArea", "TotalBsmtSF",
"YearBuilt")
datos_seleccionados <- datos[, variables_clave]

# Paso 3: Detectar el tipo de cada variable
# Clasificar las variables en numéricas o categóricas
tipo_variable <- sapply(datos_seleccionados, function(x) {
  if (is.character(x) || is.factor(x)) {
    "Categorical"
  } else if (is.numeric(x)) {
    "Numerical"
```

```
  } else {
    "Other"
  }
})

# Mostrar el tipo de cada variable
cat("\nTipos de variables seleccionadas:\n")
print(tipo_variable)

# Paso 4: Análisis de valores faltantes
# Calcular valores faltantes por columna
valores_faltantes <- colSums(is.na(datos_seleccionados))
datos_faltantes <- data.frame(Variable = names(valores_faltantes),
                              Faltantes = valores_faltantes,
                              Tipo = tipo_variable)
cat("\nResumen de valores faltantes:\n")
print(datos_faltantes)

# Paso 5: Análisis descriptivo ajustado al tipo de variable

# Crear carpeta para guardar las gráficas
if (!dir.exists("graficas")) {
  dir.create("graficas")
}

# Variables numéricas
numerical_vars <-
names(datos_seleccionados)[tipo_variable[names(datos_seleccionados
)] == "Numerical"]
if (length(numerical_vars) > 0) {
  cat("\nResumen estadístico de variables numéricas:\n")
  print(summary(datos_seleccionados[numerical_vars]))

  # Guardar histogramas de variables numéricas
  for (var in numerical_vars) {
    p <- ggplot(datos_seleccionados, aes(x = !!sym(var))) +
      geom_histogram(fill = "steelblue", color = "black", bins =
30) +
      labs(title = paste("Distribución de", var), x = var, y =
"Frecuencia") +
      theme_minimal()
```

```
    ggsave(filename = paste0("graficas/Distribucion_", var,
".png"), plot = p, width = 8, height = 6)
  }
}

# Variables categóricas
categorical_vars <-
names(datos_seleccionados)[tipo_variable[names(datos_seleccionados
)] == "Categorical"]
if (length(categorical_vars) > 0) {
  cat("\nTablas de frecuencia para variables categóricas:\n")
  for (var in categorical_vars) {
    cat("\nVariable:", var, "\n")
    print(table(datos_seleccionados[[var]]))
  }

  # Guardar gráficos de barras de variables categóricas
  for (var in categorical_vars) {
    p <- ggplot(datos_seleccionados, aes(x = !!sym(var))) +
      geom_bar(fill = "steelblue", color = "black") +
      labs(title = paste("Frecuencia de", var), x = var, y =
"Cantidad") +
      theme(axis.text.x = element_text(angle = 45, hjust = 1))
    ggsave(filename = paste0("graficas/Frecuencia_", var, ".png"),
plot = p, width = 8, height = 6)
  }
}

# Paso 6: Visualización de relaciones clave
# Relación entre variables numéricas y la variable dependiente
(SalePrice)
if ("SalePrice" %in% numerical_vars) {
  for (var in numerical_vars) {
    if (var != "SalePrice") {
      p <- ggplot(datos_seleccionados, aes(x = !!sym(var), y =
SalePrice)) +
        geom_point(color = "blue") +
        geom_smooth(method = "lm", color = "red") +
        labs(title = paste("Relación entre", var, "y Precio de
Venta"), x = var, y = "Precio de Venta")
      ggsave(filename = paste0("graficas/Relacion_", var,
"_SalePrice.png"), plot = p, width = 8, height = 6)
```

```
    }
  }
}

# Paso final: Confirmar que todas las variables fueron procesadas
cat("\nTodas las variables seleccionadas fueron analizadas sin
problemas.\n")
```

Script 2: analisis_correlacion_multicolinealidadF

```
# Cargar las librerías necesarias
library(dplyr)
library(GGally)
library(car)

# Paso 1: Seleccionar solo las variables numéricas para el
análisis de correlación
numerical_vars <- datos_seleccionados %>%
  select_if(is.numeric)

# Paso 2: Calcular la matriz de correlación
correlation_matrix <- cor(numerical_vars, use = "complete.obs")

# Paso 3: Visualizar la matriz de correlación con formato
"pairplot"
pairplot <- ggpairs(
  numerical_vars,
  title = "Relaciones entre Variables Numéricas",
  upper = list(continuous = wrap("cor", size = 3)),
  diag = list(continuous = wrap("densityDiag")),
  lower = list(continuous = wrap("smooth", alpha = 0.3))
)

# Guardar la visualización
ggsave(filename = "matriz_correlacion_pairplot.png", plot =
pairplot, width = 10, height = 10)

# Paso 4: Identificar correlaciones significativas (>|0.7|)
# Transformar la matriz de correlación en una tabla para
identificar valores altos
significant_correlations <- which(abs(correlation_matrix) > 0.7 &
correlation_matrix != 1,
```

```
                                    arr.ind = TRUE)
correlation_table <- data.frame(
  Variable1 =
rownames(correlation_matrix)[significant_correlations[, 1]],
  Variable2 =
colnames(correlation_matrix)[significant_correlations[, 2]],
  Correlation = correlation_matrix[significant_correlations]
)

# Eliminar duplicados (pares repetidos de correlaciones)
correlation_table <-
correlation_table[!duplicated(t(apply(correlation_table, 1,
sort))), ]

# Mostrar correlaciones significativas
cat("\nCorrelaciones significativas (>|0.7|):\n")
print(correlation_table)

# Paso 5: Visualización de relaciones significativas
if (nrow(correlation_table) > 0) {
  for (i in 1:nrow(correlation_table)) {
    var1 <- correlation_table$Variable1[i]
    var2 <- correlation_table$Variable2[i]

    if (var1 == "OverallQual" || var2 == "OverallQual") {
      # Usar un gráfico de cajas para OverallQual
      ggplot(numerical_vars, aes(x =
as.factor(.data[["OverallQual"]]), y = .data[["SalePrice"]])) +
        geom_boxplot(fill = "steelblue", color = "black") +
        labs(title = "Relación entre OverallQual y SalePrice",
             x = "OverallQual (Calidad General)", y = "SalePrice
(Precio de Venta)") +
        theme_minimal() -> plot
    } else {
      # Usar un gráfico de dispersión para otras relaciones
      ggplot(numerical_vars, aes(x = .data[[var1]], y =
.data[[var2]])) +
        geom_point(color = "blue") +
        geom_smooth(method = "lm", color = "red") +
        labs(title = paste("Relación entre", var1, "y", var2),
             x = var1, y = var2) +
        theme_minimal() -> plot
```

```
    }

    # Guardar las visualizaciones
    ggsave(filename = paste0("graficas/Relacion_", var1, "_",
var2, ".png"),
           plot = plot, width = 8, height = 6)
  }
} else {
  cat("\nNo se encontraron correlaciones significativas mayores a
0.7.\n")
}

# Paso 6: Evaluación de multicolinealidad con VIF
vif_model <- lm(SalePrice ~ ., data = numerical_vars)
vif_values <- vif(vif_model)

# Mostrar los valores de VIF
vif_table <- data.frame(Variable = names(vif_values), VIF =
vif_values)
cat("\nValores de VIF para detectar multicolinealidad:\n")
print(vif_table)

# Identificar variables con VIF > 10
high_vif <- vif_table %>% filter(VIF > 10)
cat("\nVariables con alta multicolinealidad (VIF > 10):\n")
print(high_vif)

# Recomendación para manejo de multicolinealidad
if (nrow(high_vif) > 0) {
  cat("\nSugerencia: Considerar eliminar una de las variables con
alta multicolinealidad o realizar transformaciones.\n")
} else {
  cat("\nNo se detectaron problemas significativos de
multicolinealidad (VIF <= 10).\n")
}
```

Script 3: modelado_y_seleccion_modelo

```
# Cargar las librerías necesarias
library(MASS)
library(dplyr)
library(car)
```

```
# Paso 1: Preparar los datos para el modelado
# Seleccionar las variables significativas según el análisis
previo
selected_vars <- c("SalePrice", "GrLivArea", "OverallQual")
model_data <- numerical_vars[, selected_vars]

# Verificar si hay valores faltantes en los datos seleccionados
if (any(is.na(model_data))) {
  stop("Existen valores faltantes en los datos seleccionados para
el modelado.")
}

# Paso 2: Construir los modelos
# Modelo completo (ambas variables independientes)
full_model <- lm(SalePrice ~ GrLivArea + OverallQual, data =
model_data)

# Modelo con solo GrLivArea
model_grlivarea <- lm(SalePrice ~ GrLivArea, data = model_data)

# Modelo con solo OverallQual
model_overallqual <- lm(SalePrice ~ OverallQual, data =
model_data)

# Paso 3: Resumen de los modelos
cat("\nResumen del modelo completo:\n")
print(summary(full_model))

cat("\nResumen del modelo con GrLivArea:\n")
print(summary(model_grlivarea))

cat("\nResumen del modelo con OverallQual:\n")
print(summary(model_overallqual))

# Paso 4: Comparación de modelos usando AIC
aic_full <- AIC(full_model)
aic_grlivarea <- AIC(model_grlivarea)
aic_overallqual <- AIC(model_overallqual)

cat("\nComparación de AIC:\n")
cat("Modelo completo AIC:", aic_full, "\n")
cat("Modelo con GrLivArea AIC:", aic_grlivarea, "\n")
```

```
cat("Modelo con OverallQual AIC:", aic_overallqual, "\n")

# Paso 5: Evaluación de modelos (R^2 ajustado y AIC)
cat("\nComparación de modelos (R^2 ajustado y AIC):\n")
comparison_table <- data.frame(
  Modelo = c("Completo", "Solo GrLivArea", "Solo OverallQual"),
  R2_Ajustado = c(summary(full_model)$adj.r.squared,
                  summary(model_grlivarea)$adj.r.squared,
                  summary(model_overallqual)$adj.r.squared),
  AIC = c(aic_full, aic_grlivarea, aic_overallqual)
)
print(comparison_table)

# Paso 6: Guardar los resultados de comparación
write.csv(comparison_table, "comparacion_modelos.csv", row.names =
FALSE)

# Paso 7: Guardar diagnósticos del modelo completo
png("diagnostico_modelo_completo.png", width = 800, height = 800)
par(mfrow = c(2, 2))  # Configurar para mostrar múltiples gráficas
plot(full_model, main = "Diagnóstico del Modelo Completo")
dev.off()

# Paso 8: Guardar diagnósticos del modelo con GrLivArea
png("diagnostico_modelo_grlivarea.png", width = 800, height = 800)
par(mfrow = c(2, 2))  # Configurar para mostrar múltiples gráficas
plot(model_grlivarea, main = "Diagnóstico del Modelo con
GrLivArea")
dev.off()

# Paso 9: Guardar diagnósticos del modelo con OverallQual
png("diagnostico_modelo_overallqual.png", width = 800, height =
800)
par(mfrow = c(2, 2))  # Configurar para mostrar múltiples gráficas
plot(model_overallqual, main = "Diagnóstico del Modelo con
OverallQual")
dev.off()

# Paso 10: Comparación adicional y selección del modelo completo
cat("\nEl modelo completo se selecciona debido a un balance entre
menor AIC y mayor R^2 ajustado.\n")
```

```
# Paso 11: Aplicar transformaciones (etapa posterior)
# Aplicar log-transformación para abordar no linealidad y
heteroscedasticidad
model_data$LogSalePrice <- log(model_data$SalePrice)
model_data$LogGrLivArea <- log(model_data$GrLivArea)

# Modelo con transformaciones
transformed_model <- lm(LogSalePrice ~ LogGrLivArea + OverallQual,
data = model_data)
cat("\nResumen del modelo con transformaciones:\n")
print(summary(transformed_model))

# Diagnósticos del modelo transformado
cat("\nDiagnósticos del modelo transformado:\n")
png("diagnosticos_transformed_model.png", width = 1200, height =
1000)
par(mfrow = c(2, 2))  # Configurar para mostrar múltiples gráficas
plot(transformed_model)
dev.off()

# Paso 12: Comparación final de AIC
aic_transformed <- AIC(transformed_model)
cat("\nComparación final de AIC:\n")
cat("Modelo completo AIC:", aic_full, "\n")
cat("Modelo transformado AIC:", aic_transformed, "\n")

# Evaluación final
if (aic_transformed < aic_full) {
  cat("\nEl modelo transformado es preferible debido a un menor
AIC.\n")
} else {
  cat("\nEl modelo completo se mantiene como preferido.\n")
}

# Guardar el modelo transformado
saveRDS(transformed_model, file = "modelo_transformado.rds")
cat("\nEl modelo transformado ha sido guardado como
'modelo_transformado.rds'.\n")
```

Script 4: modelo_training_testing_metricas

```
# Cargar las librerías necesarias
library(dplyr)
```

```
library(ggplot2)
library(corrplot)
library(car)
library(Metrics)

# Definir la función calculate_metrics
calculate_metrics <- function(actual, predicted) {
  mae <- mae(actual, predicted)
  rmse <- rmse(actual, predicted)
  mape <- mape(actual, predicted)
  r_squared <- 1 - sum((actual - predicted)^2) / sum((actual -
mean(actual))^2)

  metrics <- data.frame(
    MAE = mae,
    RMSE = rmse,
    MAPE = mape,
    R_squared = r_squared
  )
  return(metrics)
}

# Paso 1: Seleccionar solo las variables numéricas para el
análisis de correlación
numerical_vars <- datos_seleccionados %>%
  select_if(is.numeric)

# Paso 2: Calcular la matriz de correlación
correlation_matrix <- cor(numerical_vars, use = "complete.obs")

# Paso 3: Visualizar la matriz de correlación
corrplot(correlation_matrix, method = "color", type = "upper",
         tl.col = "black", tl.srt = 45, title = "Matriz de
Correlación")

# Paso 4: Identificar correlaciones significativas (>|0.7|)
significant_correlations <- which(abs(correlation_matrix) > 0.7 &
correlation_matrix != 1, arr.ind = TRUE)
correlation_table <- data.frame(
  Variable1 =
rownames(correlation_matrix)[significant_correlations[, 1]],
```

```
  Variable2 =
colnames(correlation_matrix)[significant_correlations[, 2]],
  Correlation = correlation_matrix[significant_correlations]
)
correlation_table <-
correlation_table[!duplicated(t(apply(correlation_table, 1,
sort))), ]
print(correlation_table)

# Paso 5: Visualización de relaciones significativas
if (nrow(correlation_table) > 0) {
  for (i in 1:nrow(correlation_table)) {
    var1 <- correlation_table$Variable1[i]
    var2 <- correlation_table$Variable2[i]

    if (var1 == "OverallQual" || var2 == "OverallQual") {
      ggplot(numerical_vars, aes(x =
as.factor(.data[["OverallQual"]]), y = .data[["SalePrice"]])) +
        geom_boxplot(fill = "steelblue", color = "black") +
        labs(title = "Relación entre OverallQual y SalePrice",
             x = "OverallQual (Calidad General)", y = "SalePrice
(Precio de Venta)") +
        theme_minimal() -> plot
    } else {
      ggplot(numerical_vars, aes(x = .data[[var1]], y =
.data[[var2]])) +
        geom_point(color = "blue") +
        geom_smooth(method = "lm", color = "red") +
        labs(title = paste("Relación entre", var1, "y", var2),
             x = var1, y = var2) +
        theme_minimal() -> plot
    }
    ggsave(filename = paste0("graficas/Relacion_", var1, "_",
var2, ".png"),
           plot = plot, width = 8, height = 6)
  }
}

# Paso 6: Evaluación de multicolinealidad con VIF
vif_model <- lm(SalePrice ~ ., data = numerical_vars)
vif_values <- vif(vif_model)
print(vif_values)
```

```
# Paso 7: Dividir los datos en conjuntos de entrenamiento y prueba
set.seed(123)
train_indices <- sample(1:nrow(numerical_vars), size = 0.7 *
nrow(numerical_vars))
train_data <- numerical_vars[train_indices, ]
test_data <- numerical_vars[-train_indices, ]

# Guardar los datos de entrenamiento y prueba como CSV
write.csv(train_data, "train_data.csv", row.names = FALSE)
write.csv(test_data, "test_data.csv", row.names = FALSE)

# Paso 8: Ajustar el modelo transformado al conjunto de
entrenamiento
train_data$LogSalePrice <- log(train_data$SalePrice)
train_data$LogGrLivArea <- log(train_data$GrLivArea)
transformed_model_train <- lm(LogSalePrice ~ LogGrLivArea +
OverallQual, data = train_data)

# Paso 9: Predicciones y métricas en el conjunto de entrenamiento
predicted_train <- predict(transformed_model_train, newdata =
train_data)
metrics_train <- calculate_metrics(train_data$LogSalePrice,
predicted_train)
print(metrics_train)

# Paso 10: Predicciones y métricas en el conjunto de prueba
test_data$LogSalePrice <- log(test_data$SalePrice)
test_data$LogGrLivArea <- log(test_data$GrLivArea)
predicted_test <- predict(transformed_model_train, newdata =
test_data)
metrics_test <- calculate_metrics(test_data$LogSalePrice,
predicted_test)
print(metrics_test)

# Paso 11: Guardar el modelo transformado entrenado
saveRDS(transformed_model_train, file =
"transformed_model_train.rds")
cat("El modelo transformado entrenado ha sido guardado como
'transformed_model_train.rds'.\n")
```

(2) Matriz de correlaciones