

# Actividad IA

Juan Camilo Mendoza Robledo

Desarrollo de Software - Parquesoft Ti

## Actividad 1

<b>Título:</b>	Kaggle, House Prices - Advanced Regression Techniques.
<b>Resumen:</b>	El dataset "House Prices - Advanced Regression Techniques" de Kaggle trata sobre la predicción del precio de venta de casas
<b>Origen:</b>	<a href="https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques/data">https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques/data</a>
<b>Número total de Variables:</b>	81 (incluyendo la variable objetivo)
<b>Número total de Variables Cualitativas:</b>	43
<b>Número total de Variables Cuantitativas:</b>	38
<b>Variable por Predecir:</b>	SalePrice
<b>Algoritmo de Predicción:</b>	Regresión Lineal

## Actividad 2

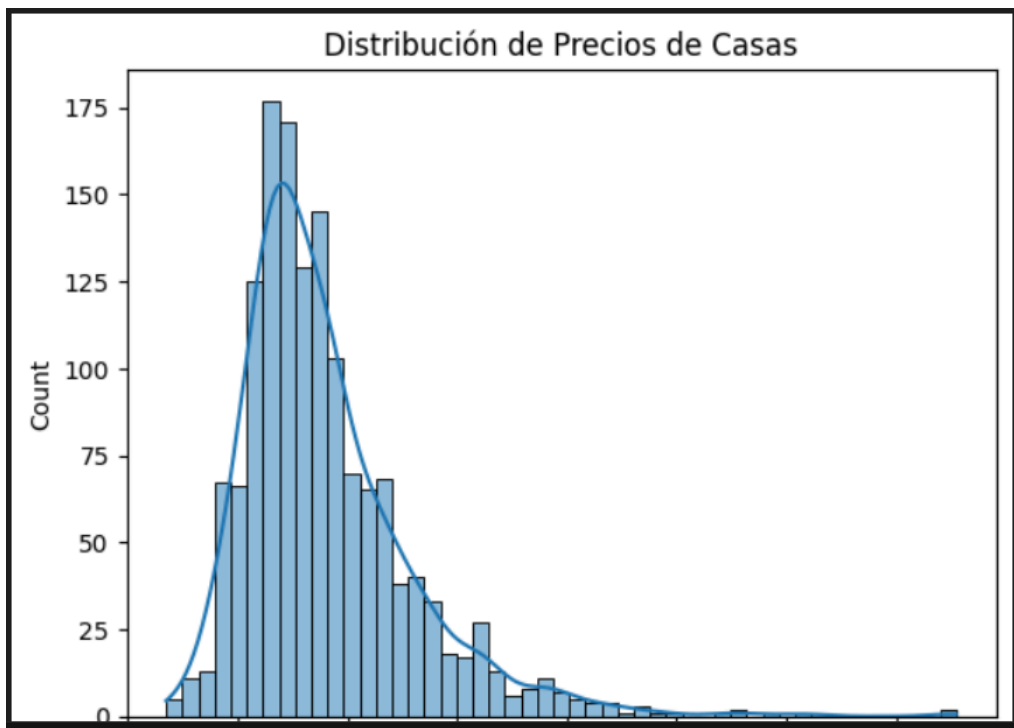
Se importaron las librerías y se procesaron los datos del CSV:

```
[1] ✓ 19.5s Python
# Importar bibliotecas necesarias
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score

[2] ✓ 0.0s Python
df = pd.read_csv('train.csv')

[3] ✓ 0.0s Python
df.head()
```

Se procesa la información para visualizar los datos y la distribución de precios



```
# Exploración de datos
# Información básica del dataset
df.info()

# Descripción estadística
df.describe()

# Visualizar la distribución de precios
sns.histplot(df['SalePrice'], kde=True)
plt.title('Distribución de Precios de Casas')
plt.show()
```

✓ 0.2s

```
# Evaluar el modelo
y_pred = model.predict(x_test)

# Métricas de evaluación
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print(f'Mean Squared Error: {mse}')
print(f'R^2 Score: {r2}')
```

[9] ✓ 0.0s

... Mean Squared Error: 1572343803.577853  
R^2 Score: 0.7950095261783572

Al final el **Gráfico de Dispersión** Muestra cómo se comparan las predicciones del modelo con los precios reales. Los puntos cercanos a la diagonal indican buenas predicciones, mientras que los puntos alejados indican errores en la predicción.

```
# Visualizar las predicciones
plt.scatter(y_test, y_pred)
plt.xlabel('Actual Prices')
plt.ylabel('Predicted Prices')
plt.title('Actual vs Predicted Prices')
plt.show()
```

✓ 0.1s

