In [9]:
```
import pandas as pd
import numpy as np
import warnings
import math
import scipy.stats
import statsmodels.api as sm
warnings.filterwarnings("ignore")

import os
os.chdir("/Users/Lenovo/Desktop/EBAC")
```

In [66]:
```
data = pd.read_csv("kc_house_data.csv")
data
```

Out[66]:

| | id | date | price | bedrooms | bathrooms | sqft_living | sqft_lot | floors | waterfront | view | ... | grade | sq |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 7129300520 | 20141013T000000 | 221900.0 | 3 | 1.00 | 1180 | 5650 | 1.0 | 0 | 0 | ... | 7 | |
| **1** | 6414100192 | 20141209T000000 | 538000.0 | 3 | 2.25 | 2570 | 7242 | 2.0 | 0 | 0 | ... | 7 | |
| **2** | 5631500400 | 20150225T000000 | 180000.0 | 2 | 1.00 | 770 | 10000 | 1.0 | 0 | 0 | ... | 6 | |
| **3** | 2487200875 | 20141209T000000 | 604000.0 | 4 | 3.00 | 1960 | 5000 | 1.0 | 0 | 0 | ... | 7 | |
| **4** | 1954400510 | 20150218T000000 | 510000.0 | 3 | 2.00 | 1680 | 8080 | 1.0 | 0 | 0 | ... | 8 | |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | |
| **21608** | 263000018 | 20140521T000000 | 360000.0 | 3 | 2.50 | 1530 | 1131 | 3.0 | 0 | 0 | ... | 8 | |
| **21609** | 6600060120 | 20150223T000000 | 400000.0 | 4 | 2.50 | 2310 | 5813 | 2.0 | 0 | 0 | ... | 8 | |
| **21610** | 1523300141 | 20140623T000000 | 402101.0 | 2 | 0.75 | 1020 | 1350 | 2.0 | 0 | 0 | ... | 7 | |
| **21611** | 291310100 | 20150116T000000 | 400000.0 | 3 | 2.50 | 1600 | 2388 | 2.0 | 0 | 0 | ... | 8 | |
| **21612** | 1523300157 | 20141015T000000 | 325000.0 | 2 | 0.75 | 1020 | 1076 | 2.0 | 0 | 0 | ... | 7 | |

21613 rows × 21 columns

In [108…
```
Y = data['price'].values.reshape(-1,1)
features = ['bedrooms', 'bathrooms', 'sqft_living', 'sqft_lot', 'floors', 'waterfront', 'view', 'condition', 'g
X = data[features].copy()
X = sm.add_constant(X)
X = X.values
```

In [110…
```
XT_X = np.matmul(np.matrix.transpose(X), X)
XT_X_inv = np.linalg.inv(XT_X)
XT_Y = np.matmul(np.matrix.transpose(X), Y)
betas = np.matmul(XT_X_inv, XT_Y)

# Calculo de los pronosticos para Y de acuerdo a los coeficientes de regresion
Y_pred = np.matmul(X, betas)

# Calculo de residuales
Resid = Y - Y_pred

# Calculo de suma de residuales al cuadrado
RSS = float(np.matmul(np.matrix.transpose(Resid), Resid))

# Calculo de suma total de cuadrados
TSS = float(np.matmul(np.matrix.transpose(Y), Y) - len(Y)*(Y.mean()**2))

# Calculo de coeficiente de determinacion
R_cuad = float(1 - RSS/TSS)

# Calculo de coeficiente de determinacion ajustado
RSqAj = float(1 - (RSS / (X.shape[0] - X.shape[1])) / (TSS / X.shape[0] - 1))

# Calculo de varianza de error de regresion
s_cuad = RSS / (len(Y) - X.shape[1])

# Desviacion estandar del error de regresion
s = math.sqrt(s_cuad)

# Calculo de las t's estadisticas para cada coeficiente de regresion
result_t = []
for i in range(0, X.shape[1]):
    t = float(betas[i] / (s*math.sqrt(XT_X_inv[i][i])))
    result_t.append(t)
result_t

# Obtener valor critico de la t de Student de tablas
```

```
import scipy.stats

grados_libertad = len(Y) - X.shape[1]
# La t critica se obtendra a un nivel de confianza del 95% (Alfa = 5%)
t_critico = abs(scipy.stats.t.ppf(q = 0.025, df = grados_libertad))

## Criterio 1
for i in range(0, X.shape[1]):
    if abs(result_t[i] > t_critico):
        print("Beta", i, "es significativa") # Aqui se rechaza H0
    else:
        print("Beta", i, "NO es significativa") # Aqui NO se rechaza H0

## Criterio 2
# Calculo de valores p
for i in range(0, X.shape[1]):
    print("Valor p de Beta", i, ":", scipy.stats.t.sf(abs(result_t[i]), df = grados_libertad) * 2)

## Criterio 3
# Calculo de intervalos de confianza del 95% para el verdadero valor del coeficiente de cada Beta
for i in range(0, X.shape[1]):
    print("El valor de Beta", i, "se encuentra entre", float(betas[i]) - t_critico * s * math.sqrt(XT_X_inv[i][
          "y", float(betas[i]) + t_critico * s * math.sqrt(XT_X_inv[i][i]))
```

```
Beta 0 NO es significativa
Beta 1 NO es significativa
Beta 2 NO es significativa
Beta 3 es significativa
Beta 4 NO es significativa
Beta 5 NO es significativa
Beta 6 es significativa
Beta 7 es significativa
Beta 8 es significativa
Beta 9 es significativa
Beta 10 es significativa
Beta 11 NO es significativa
Valor p de Beta 0 : 0.0
Valor p de Beta 1 : 1.5683673954192252e-53
Valor p de Beta 2 : 0.6212899800230343
Valor p de Beta 3 : 0.0
Valor p de Beta 4 : 0.6112292011242967
Valor p de Beta 5 : 5.883625993167733e-06
Valor p de Beta 6 : 1.3208697103460223e-232
Valor p de Beta 7 : 1.052456612158509e-156
Valor p de Beta 8 : 2.661064654662993e-106
Valor p de Beta 9 : 0.0
Valor p de Beta 10 : 0.0
Valor p de Beta 11 : 1.859913078707342e-90
El valor de Beta 0 se encuentra entre -60790720.642469816 y -55436267.03843839
El valor de Beta 1 se encuentra entre -34232.77028290146 y -26522.662267968422
El valor de Beta 2 se encuentra entre -4615.425207769238 y 7726.05370390948
El valor de Beta 3 se encuentra entre 194.28130141299087 y 206.66016121529057
El valor de Beta 4 se encuentra entre -0.052478153130359154 y 0.08922845247165186
El valor de Beta 5 se encuentra entre -21131.11805911055 y -8370.716158573792
El valor de Beta 6 se encuentra entre 560512.4853101289 y 631374.0615071738
El valor de Beta 7 se encuentra entre 53893.43334950097 y 62367.56623079292
El valor de Beta 8 se encuentra entre 46178.5984238194 y 55202.33416949644
El valor de Beta 9 se encuentra entre 83558.2960651409 y 91573.11703605912
El valor de Beta 10 se encuentra entre 615622.8900325744 y 657090.5005113633
El valor de Beta 11 se encuentra entre -244365.854245274 y -201261.368039043
```

In [112…

```
# Comparacion de resultados contra reporte automatizado
import statsmodels.api as sm

regressor = sm.OLS(Y, X).fit()
print(regressor.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.673
Model:                            OLS   Adj. R-squared:                  0.672
Method:                 Least Squares   F-statistic:                     4034.
Date:                Sun, 07 Sep 2025   Prob (F-statistic):               0.00
Time:                        19:09:42   Log-Likelihood:            -2.9554e+05
No. Observations:               21613   AIC:                         5.911e+05
Df Residuals:                   21601   BIC:                         5.912e+05
Df Model:                          11
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const      -5.811e+07   1.37e+06    -42.547      0.000   -6.08e+07   -5.54e+07
x1         -3.038e+04   1966.790    -15.445      0.000   -3.42e+04   -2.65e+04
x2          1555.3143   3148.218      0.494      0.621   -4615.425    7726.054
x3           200.4707      3.158     63.485      0.000     194.281     206.660
x4             0.0184      0.036      0.508      0.611      -0.052       0.089
x5         -1.475e+04   3255.082     -4.532      0.000   -2.11e+04   -8370.716
x6          5.959e+05   1.81e+04     32.968      0.000    5.61e+05    6.31e+05
x7          5.813e+04   2161.687     26.891      0.000    5.39e+04    6.24e+04
x8          5.069e+04   2301.887     22.021      0.000    4.62e+04    5.52e+04
x9          8.757e+04   2044.520     42.829      0.000    8.36e+04    9.16e+04
x10         6.364e+05   1.06e+04     60.158      0.000    6.16e+05    6.57e+05
x11        -2.228e+05    1.1e+04    -20.264      0.000   -2.44e+05   -2.01e+05
==============================================================================
Omnibus:                    17985.706   Durbin-Watson:                   1.995
Prob(Omnibus):                  0.000   Jarque-Bera (JB):          1587750.659
Skew:                           3.485   Prob(JB):                         0.00
Kurtosis:                      44.407   Cond. No.                     4.21e+07
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 4.21e+07. This might indicate that there are
strong multicollinearity or other numerical problems.
```

In [114…

```
# Mientras exista multicoordinalidad o dependencia entre variable regresoras no podemos obetener un resultado c
data = pd.DataFrame(X)
regresoras = data.iloc[:, 1:]
correlaciones = regresoras.corr()
print(correlaciones)
```

```
          1         2         3         4         5         6         7   \
1   1.000000  0.515884  0.576671  0.031703  0.175429 -0.006582  0.079532
2   0.515884  1.000000  0.754665  0.087740  0.500653  0.063744  0.187737
3   0.576671  0.754665  1.000000  0.172826  0.353949  0.103818  0.284611
4   0.031703  0.087740  0.172826  1.000000 -0.005201  0.021604  0.074710
5   0.175429  0.500653  0.353949 -0.005201  1.000000  0.023698  0.029444
6  -0.006582  0.063744  0.103818  0.021604  0.023698  1.000000  0.401857
7   0.079532  0.187737  0.284611  0.074710  0.029444  0.401857  1.000000
8   0.028472 -0.124982 -0.058753 -0.008958 -0.263768  0.016653  0.045990
9   0.356967  0.664983  0.762704  0.113621  0.458183  0.082775  0.251321
10 -0.008931  0.024573  0.052529 -0.085683  0.049614 -0.014274  0.006157
11  0.129473  0.223042  0.240223  0.229521  0.125419 -0.041910 -0.078400

          8         9         10        11
1   0.028472  0.356967 -0.008931  0.129473
2  -0.124982  0.664983  0.024573  0.223042
3  -0.058753  0.762704  0.052529  0.240223
4  -0.008958  0.113621 -0.085683  0.229521
5  -0.263768  0.458183  0.049614  0.125419
6   0.016653  0.082775 -0.014274 -0.041910
7   0.045990  0.251321  0.006157 -0.078400
8   1.000000 -0.144674 -0.014941 -0.106500
9  -0.144674  1.000000  0.114084  0.198372
10 -0.014941  0.114084  1.000000 -0.135512
11 -0.106500  0.198372 -0.135512  1.000000
```

In [116…

```
X_Nueva = np.delete(X, 2, 1)
X_Nueva
```

Out[116…
```
array([[ 1.00000e+00,  3.00000e+00,  1.18000e+03, ...,  7.00000e+00,
         4.75112e+01, -1.22257e+02],
       [ 1.00000e+00,  3.00000e+00,  2.57000e+03, ...,  7.00000e+00,
         4.77210e+01, -1.22319e+02],
       [ 1.00000e+00,  2.00000e+00,  7.70000e+02, ...,  6.00000e+00,
         4.77379e+01, -1.22233e+02],
       ...,
       [ 1.00000e+00,  2.00000e+00,  1.02000e+03, ...,  7.00000e+00,
         4.75944e+01, -1.22299e+02],
       [ 1.00000e+00,  3.00000e+00,  1.60000e+03, ...,  8.00000e+00,
         4.75345e+01, -1.22069e+02],
       [ 1.00000e+00,  2.00000e+00,  1.02000e+03, ...,  7.00000e+00,
         4.75941e+01, -1.22299e+02]])
```

In [118…
```
# Volvemos a correr el modelo
regressor = sm.OLS(Y, X_Nueva).fit()
print(regressor.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.673
Model:                            OLS   Adj. R-squared:                  0.672
Method:                 Least Squares   F-statistic:                     4437.
Date:                Sun, 07 Sep 2025   Prob (F-statistic):               0.00
Time:                        19:11:23   Log-Likelihood:            -2.9554e+05
No. Observations:               21613   AIC:                         5.911e+05
Df Residuals:                   21602   BIC:                         5.912e+05
Df Model:                          10
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const      -5.807e+07   1.36e+06    -42.607      0.000   -6.07e+07   -5.54e+07
x1         -3.019e+04   1929.441    -15.647      0.000    -3.4e+04   -2.64e+04
x2           201.0963      2.893     69.518      0.000     195.426     206.766
x3             0.0177      0.036      0.489      0.625      -0.053       0.088
x4         -1.423e+04   3077.358     -4.623      0.000   -2.03e+04   -8195.048
x5          5.959e+05   1.81e+04     32.968      0.000     5.6e+05    6.31e+05
x6          5.814e+04   2161.590     26.896      0.000    5.39e+04    6.24e+04
x7          5.066e+04   2301.028     22.016      0.000    4.61e+04    5.52e+04
x8           8.77e+04   2025.691     43.295      0.000    8.37e+04    9.17e+04
x9          6.361e+05   1.06e+04     60.187      0.000    6.15e+05    6.57e+05
x10        -2.225e+05    1.1e+04    -20.266      0.000   -2.44e+05   -2.01e+05
==============================================================================
Omnibus:                    17988.109   Durbin-Watson:                   1.995
Prob(Omnibus):                  0.000   Jarque-Bera (JB):          1589123.557
Skew:                           3.486   Prob(JB):                         0.00
Kurtosis:                      44.425   Cond. No.                     4.21e+07
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 4.21e+07. This might indicate that there are
strong multicollinearity or other numerical problems.
```

# Conclusion

Para este ejercicio tome en cuenta las variables que use en la tarea anterior, en este caso vemos que nuestra R2 es de 67.3% por lo que no es muy buena, por lo que decidi quitar la variable "bathrooms" ya que tenia un P Value muy alto en comparacion a otra variables y tenia una correlacion del 75% con la variable 3. Al eliminarla de la ecuacion encontre que nuestra R2 no se afecto, se mantuvo en el mismo resultado.

In [ ]: