entrega_1

October 25, 2025

1 Análisis de la base de datos

1.1 1. Propuesta de Negocio

La idea de negocio es …. desarrollar esta idea mejor

alcance

descripción de la base

1.2 2. Análisis descriptivo de los datos

[22]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

[6]:
```
data = pd.read_csv('BMW sales data (2010-2024) (1).csv',header=0)
data.info()
data.describe()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 50000 entries, 0 to 49999
Data columns (total 11 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   Model                 50000 non-null  object
 1   Year                  50000 non-null  int64
 2   Region                50000 non-null  object
 3   Color                 50000 non-null  object
 4   Fuel_Type             50000 non-null  object
 5   Transmission          50000 non-null  object
 6   Engine_Size_L         50000 non-null  float64
 7   Mileage_KM            50000 non-null  int64
 8   Price_USD             50000 non-null  int64
 9   Sales_Volume          50000 non-null  int64
 10  Sales_Classification  50000 non-null  object
dtypes: float64(1), int64(4), object(6)
memory usage: 4.2+ MB
```

[6]:
```
                Year  Engine_Size_L     Mileage_KM      Price_USD  Sales_Volume
count  50000.000000   50000.000000   50000.000000   50000.000000  50000.000000
mean    2017.015700       3.247180  100307.203140   75034.600900   5067.514680
std        4.324459       1.009078   57941.509344   25998.248882   2856.767125
min     2010.000000       1.500000       3.000000   30000.000000    100.000000
25%     2013.000000       2.400000   50178.000000   52434.750000   2588.000000
50%     2017.000000       3.200000  100388.500000   75011.500000   5087.000000
75%     2021.000000       4.100000  150630.250000   97628.250000   7537.250000
max     2024.000000       5.000000  199996.000000  119998.000000   9999.000000
```

[7]:
```
data.head()
```

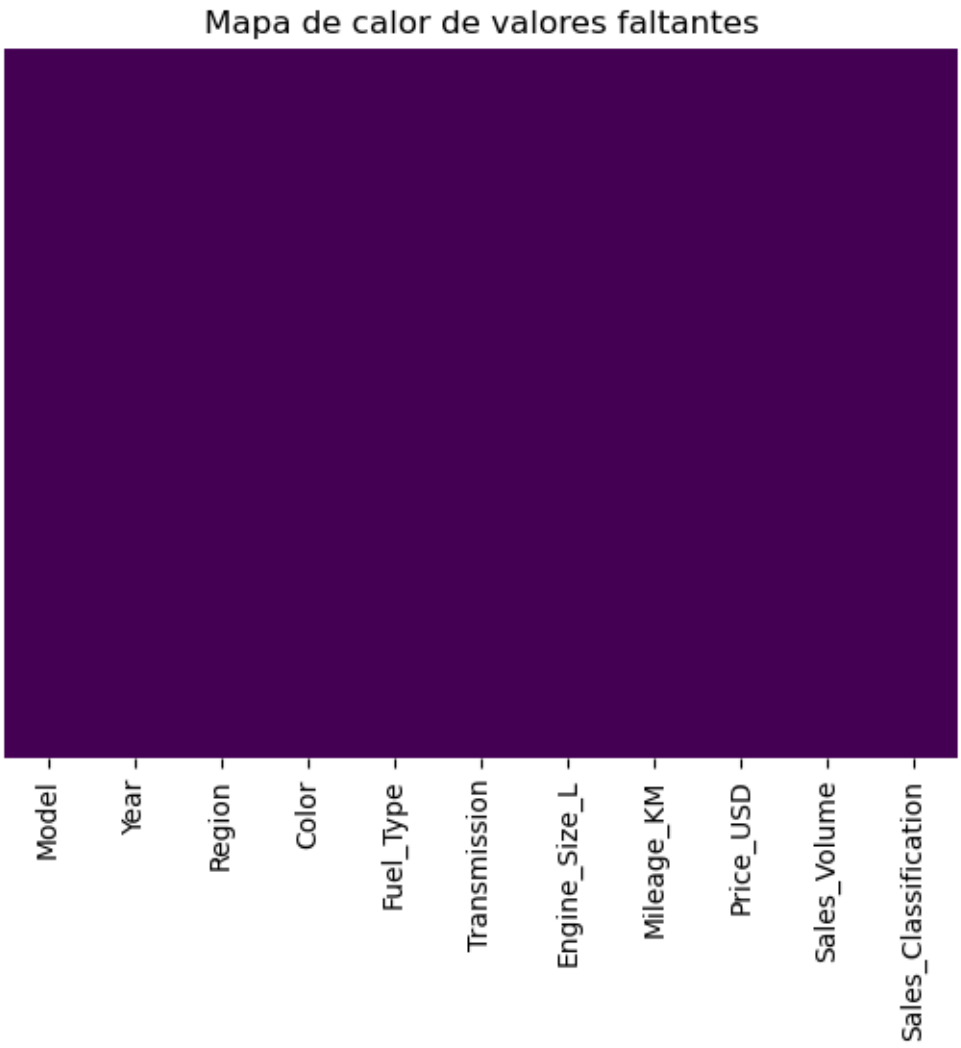
[7]:
```
      Model  Year         Region  Color Fuel_Type Transmission  Engine_Size_L  \
0  5 Series  2016           Asia    Red    Petrol       Manual            3.5
1        i8  2013  North America    Red    Hybrid    Automatic            1.6
2  5 Series  2022  North America   Blue    Petrol    Automatic            4.5
3        X3  2024    Middle East   Blue    Petrol    Automatic            1.7
4  7 Series  2020  South America  Black    Diesel       Manual            2.1

   Mileage_KM  Price_USD  Sales_Volume Sales_Classification
0      151748      98740          8300                 High
1      121671      79219          3428                  Low
2       10991     113265          6994                  Low
3       27255      60971          4047                  Low
4      122131      49898          3080                  Low
```

Limpieza de datos

[21]:
```
data.isna().sum()
```

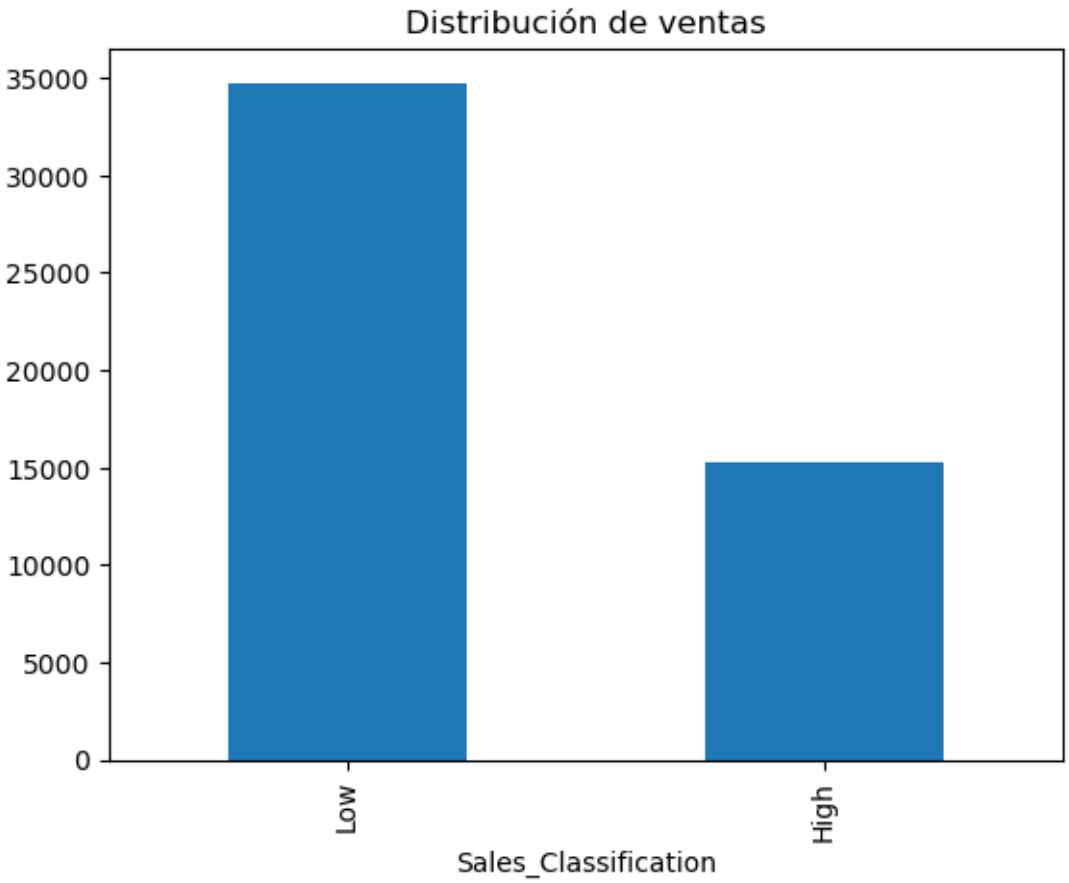
[21]:
```
Model                  0
Year                   0
Region                 0
Color                  0
Fuel_Type              0
Transmission           0
Engine_Size_L          0
Mileage_KM             0
Price_USD              0
Sales_Volume           0
Sales_Classification   0
dtype: int64
```

[23]:
```
sns.heatmap(data.isna(), cbar=False, yticklabels=False, cmap='viridis')
plt.title('Mapa de calor de valores faltantes')
plt.show()
```



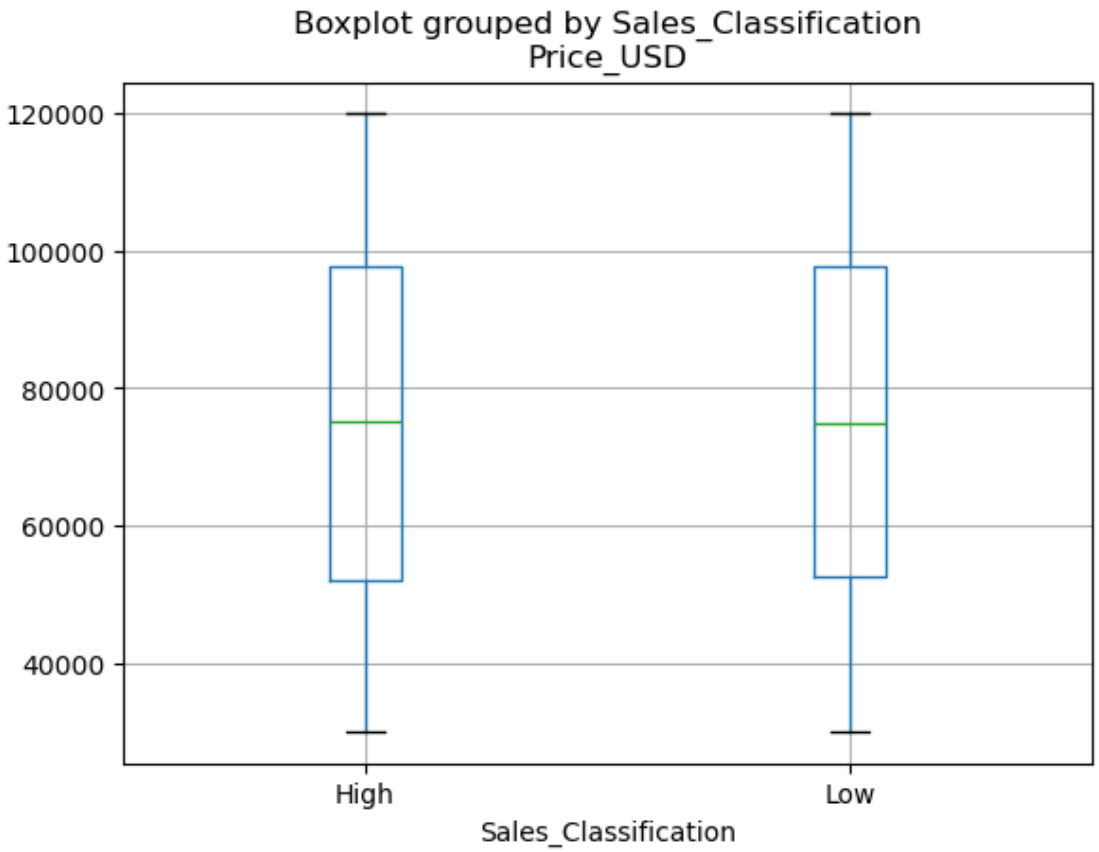
[8]:
```
data['Sales_Classification'].value_counts().plot(kind='bar')
plt.title('Distribución de ventas')
```

[8]: Text(0.5, 1.0, 'Distribución de ventas')

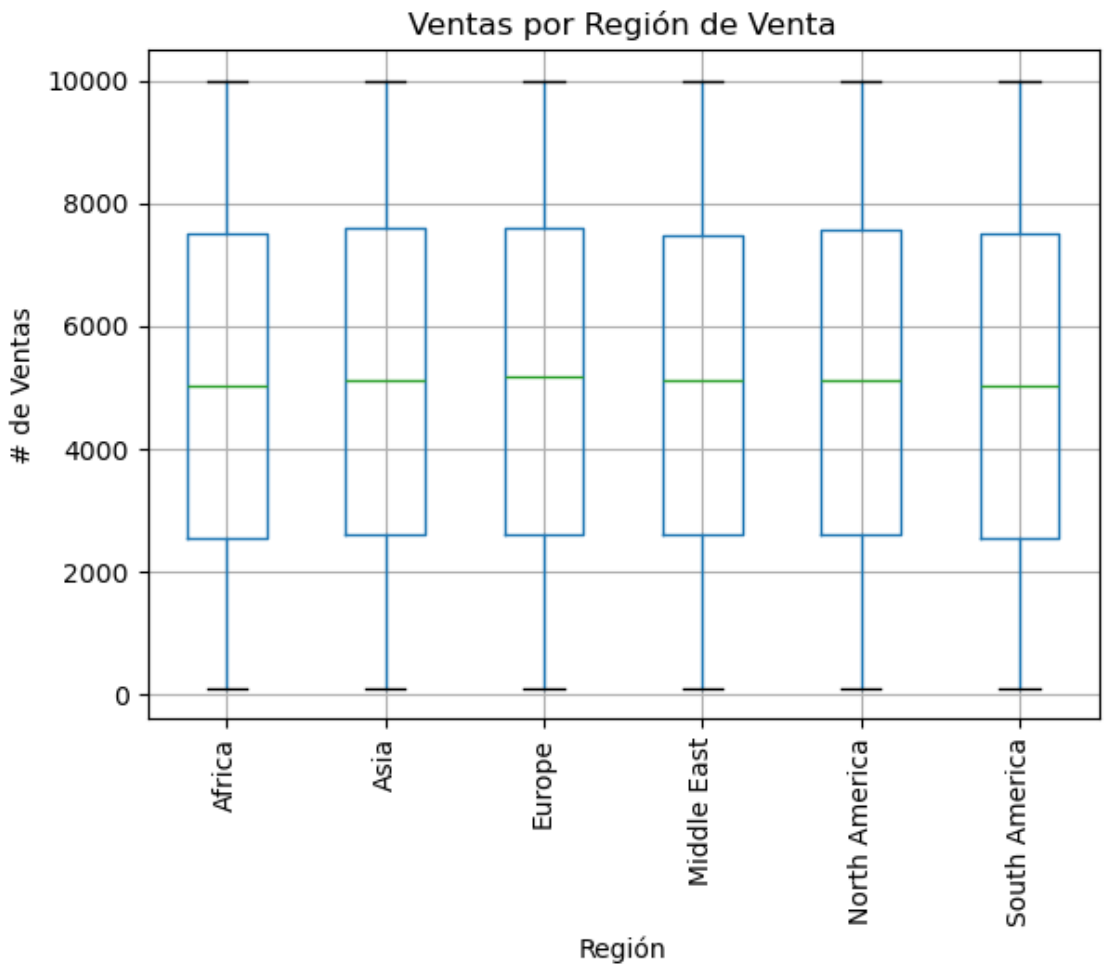


[12]:
```
data.boxplot(column='Price_USD', by='Sales_Classification')
```

[12]: <Axes: title={'center': 'Price_USD'}, xlabel='Sales_Classification'>

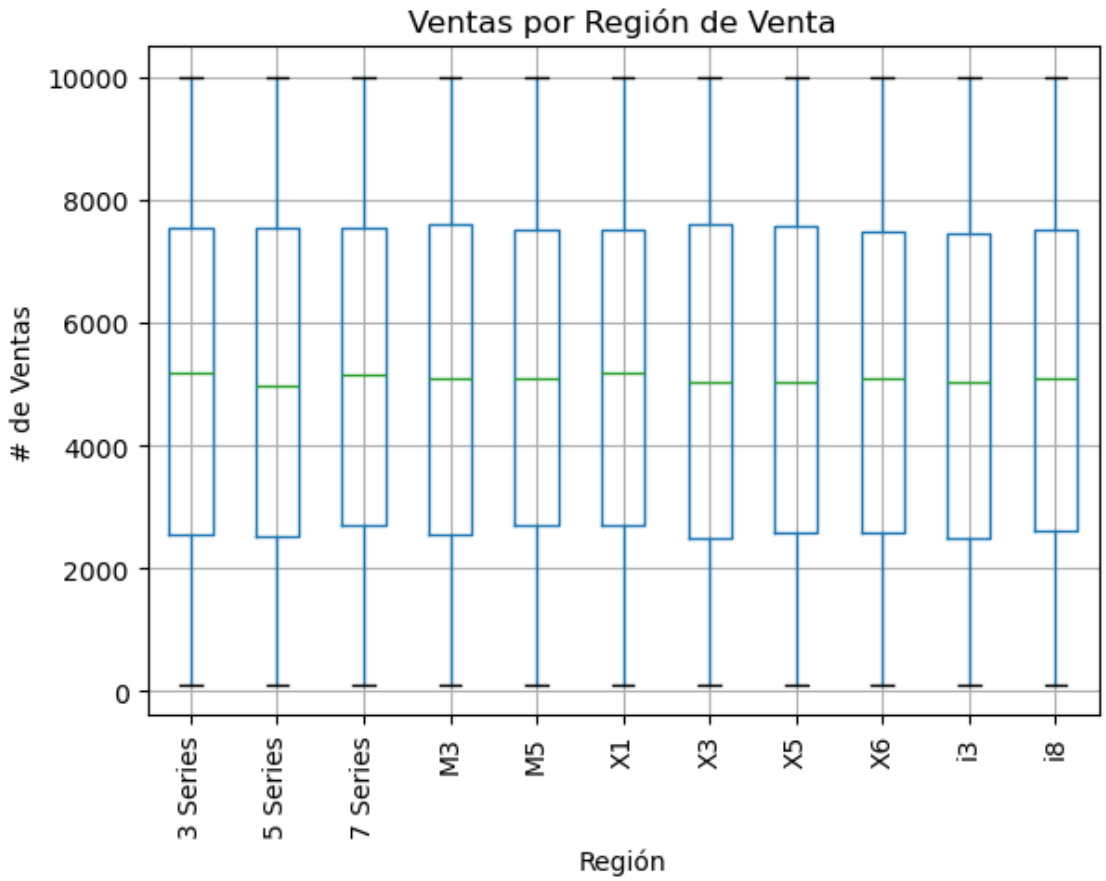


[ ]:
```
data.boxplot(column='Sales_Volume', by='Region')
plt.title('Ventas por Región de Venta')
plt.suptitle('')
plt.xlabel('Región')
plt.ylabel('# de Ventas')
plt.xticks(rotation=90)
plt.show()
```



[ ]:

[19]:
```
data.boxplot(column='Sales_Volume', by='Model')
plt.title('Ventas por Región de Venta')
plt.suptitle('')
plt.xlabel('Región')
plt.ylabel('# de Ventas')
plt.xticks(rotation=90)
plt.show()
```


Ventas por Región de Venta
10000
8000
6000
4000
2000
0
de Ventas
3 Series
5 Series
7 Series
M3
M5
X1
X3
X5
X6
i3
i8
Región

[ ]: