BY CHAU NGUYEN

CAR PRICE PREDICTION

MINDX _ DA54

Overview

A Chinese car company Geely Auto, aims to enter the US market by establishing its manufacturing unit there and producing car to compete with other US and European counterparts.

Therefore, they have contracted an automotive consulting firm to understand the factors affecting car pricing in US market, as these factors may significantly from those in Chinese market.

Overview dataset

25 columns and 205 rows

- CarID: Identification Number for Each Car
- SafetyRating: Car's Safety Rating
- CarName: Name of the Car Model
- FuelType: Type of Fuel Used (Gasoline, Diesel, Electric, etc.)
- Aspiration: Type of Aspiration (Standard or Turbocharged)
- NumDoors: Number of Doors on the Car
- BodyStyle: Style of the Car's Body (Sedan, Coupe, SUV, etc.)
- DriveWheelType: Type of Drive Wheels (Front, Rear, All)
- EngineLocation: Location of the Car's Engine (Front or Rear)
- Wheelbase: Length of the Car's Wheelbase
- CarLength: Overall Length of the Car
- CarWidth: Width of the Car
- CarHeight: Height of the Car
- CurbWeight: Weight of the Car without Passengers or Cargo
- EngineType: Type of Engine (Gas, Diesel, Electric, etc.)
- NumCylinders: Number of Cylinders in the Engine
- EngineSize: Size of the Car's Engine
- FuelSystem: Type of Fuel Delivery System
- BoreRatio: Bore-to-Stroke Ratio of the Engine
- Stroke: Stroke Length of the Engine
- CompressionRatio: Compression Ratio of the Engine
- Horsepower: Car's Engine Horsepower
- PeakRPM: Engine's Peak RPM (Revolutions Per Minute)
- CityMPG: Miles Per Gallon (MPG) in City Driving
- HighwayMPG: MPG on the Highway
- CarPrice: Price of the Car

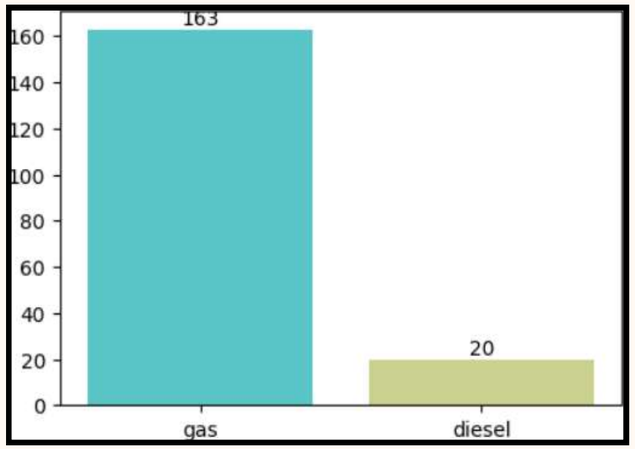
EXPLORING DATA

UNIVARIATE ANALYSIS

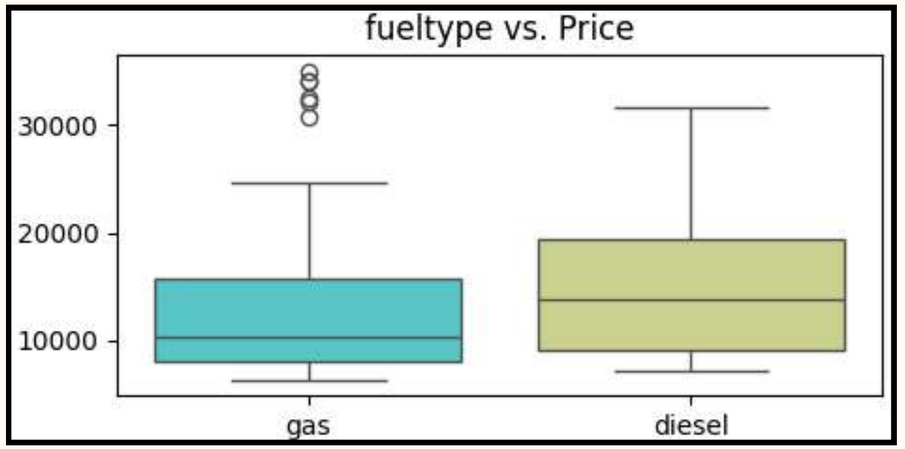
UNIVARIATE ANALYSIS

FUEL TYPE

- Diesel cars **have a higher median price** compared to gas cars.
- Diesel cars also **show less variability** in price compared to gas cars.
- Gas cars have **a higher sales volume** than diesel cars.



Sales volumns

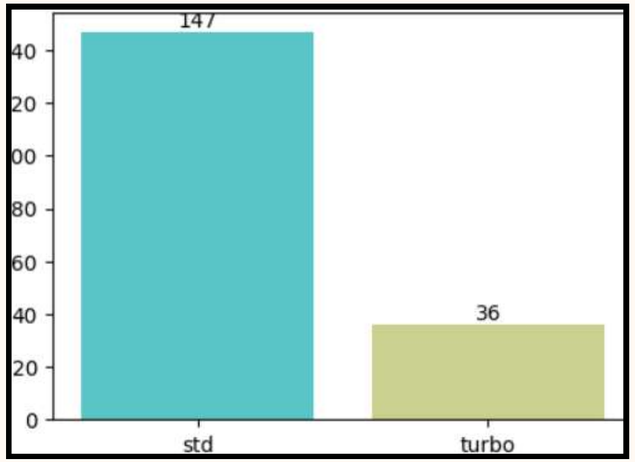


Price distribution

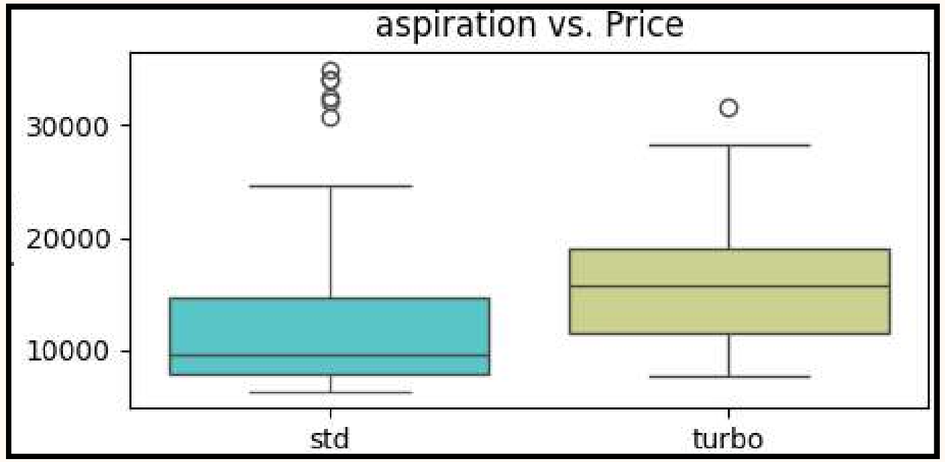
UNIVARIATE ANALYSIS

ASPIRATION

- Turbocharged (“turbo”) cars have a higher median price compared to standard (“std”) (naturally aspirated) cars.
- There is greater variability in the price of turbocharged cars compared to standard cars.
- Standard cars have **a higher sales volume** than turbocharged cars.



Sales volumns

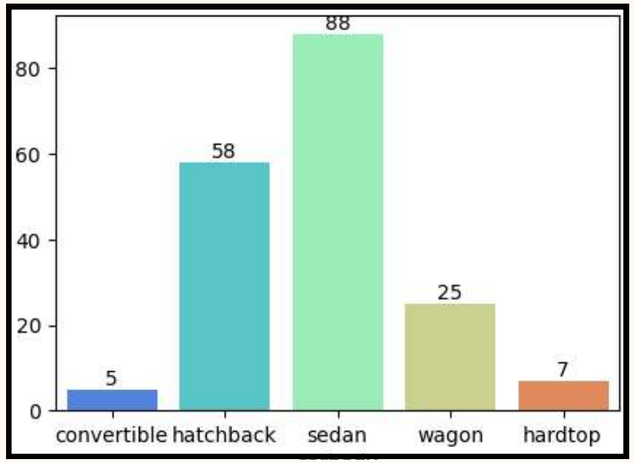


Price distribution

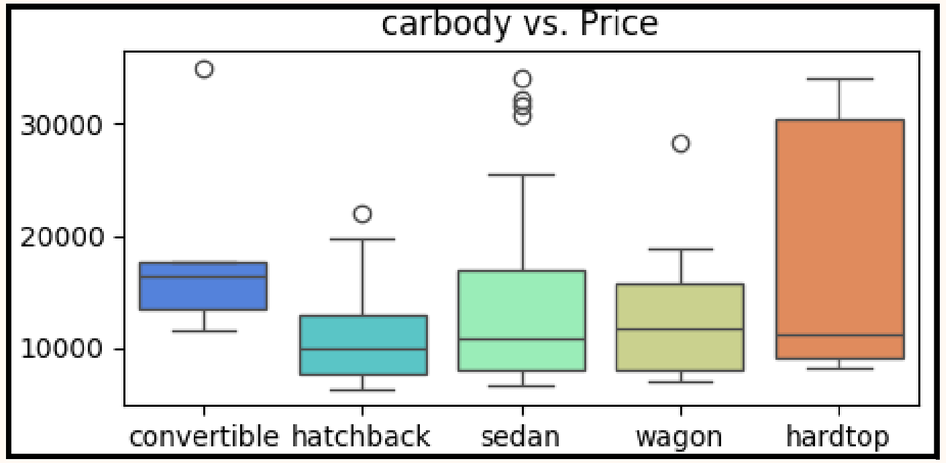
UNIVARIATE ANALYSIS

CARBODY

- Convertible cars have the highest median price.
- Hatchback, sedan, and wagon cars have lower median prices, with wagons showing the least variability.
- Sedan has **a higher sales volume** than other cars.



Sales volumns

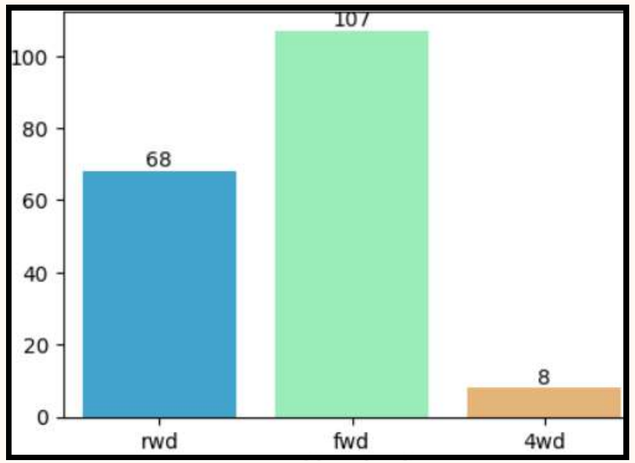


Price distribution

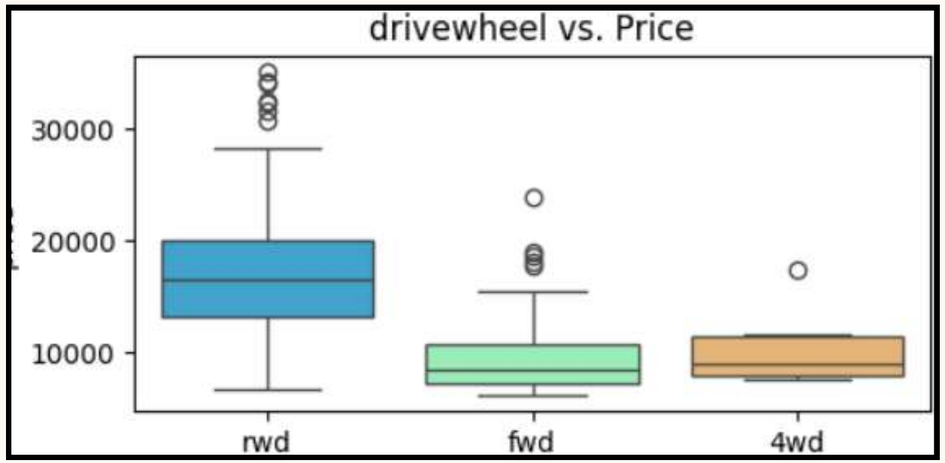
UNIVARIATE ANALYSIS

DRIVE WHEEL

- Rear-wheel-drive (RWD) cars have the highest median price, followed by four-wheel-drive (4WD) cars.
- Front-wheel-drive (FWD) cars have the lowest median price and show less variability compared to RWD and 4WD cars
- FWD has **a higher sales volume** than other drive wheel car.



Sales volumns

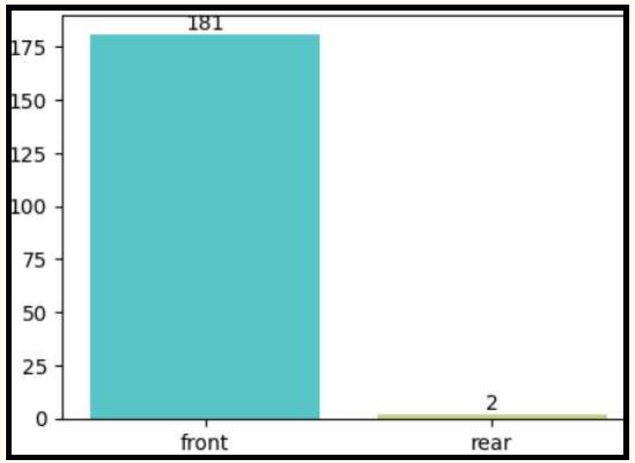


Price distribution

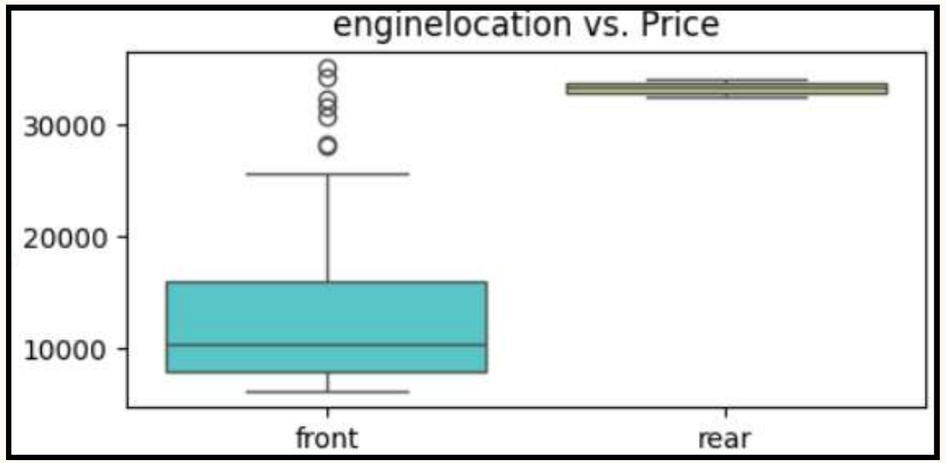
UNIVARIATE ANALYSIS

ENGINE LOCATION

- Cars with rear engine locations have significantly higher prices compared to those with front engine locations.
- Rear-engine cars show minimal variability in price, whereas front-engine cars show substantial variability.
- Far with front location has **a higher sales volume** than others.



Sales volumns

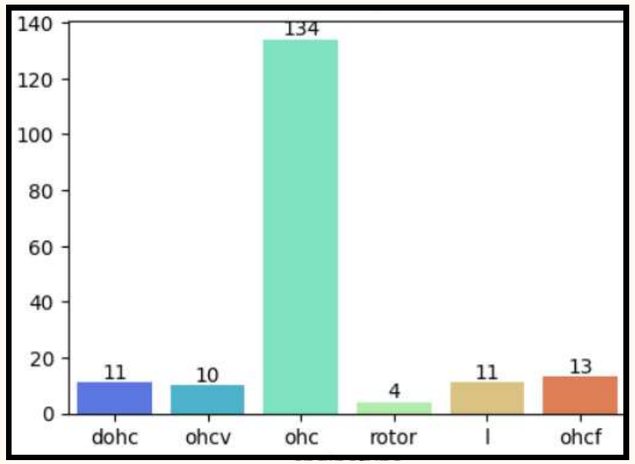


Price distribution

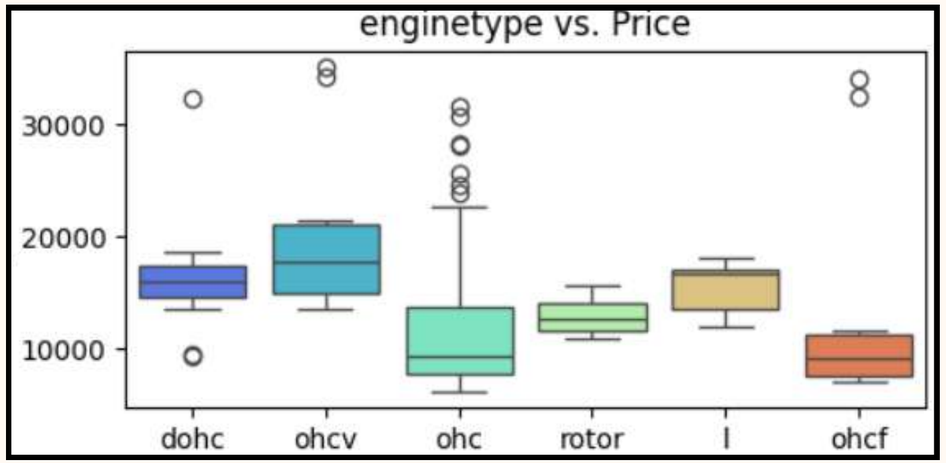
UNIVARIATE ANALYSIS

ENGINE TYPE

- DOHC and OHCV engine types have higher median prices compared to other engine types.
- OHCF engine types have the lowest median price.
- There is considerable variability in prices for all engine types, with OHCV showing the most variability
- OHC has **a higher sales volume** than others.



Sales volumns

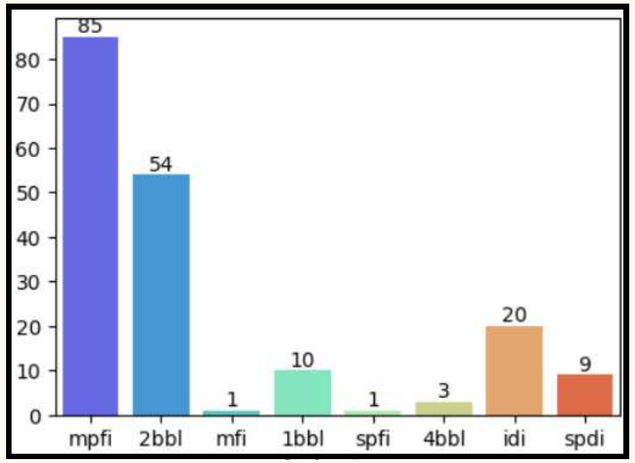


Price distribution

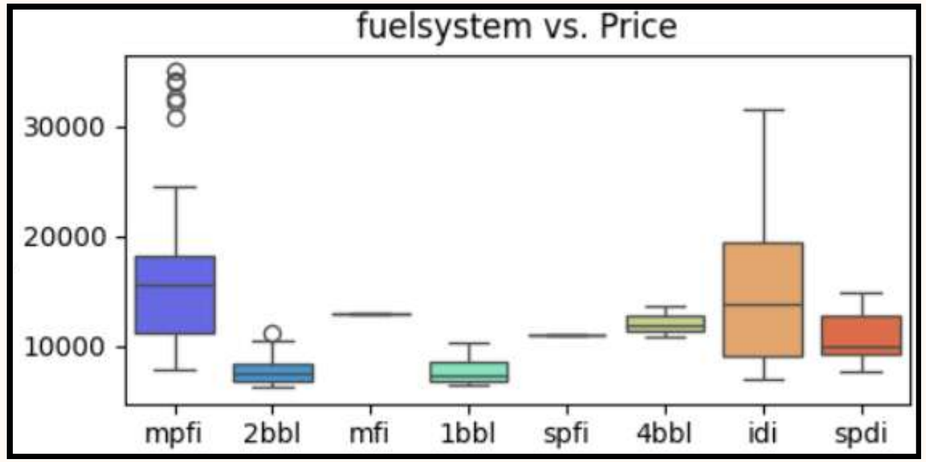
UNIVARIATE ANALYSIS

FUEL SYSTEM

- Cars with 'mpfi' (multi-point fuel injection) fuel systems have the highest median price.
- '2bbl' and '1bbl' systems have the lowest median prices and show less variability.
- There is significant variability in prices across different fuel systems, with 'mpfi' showing the most variability.
- MPFI have **a higher sales volume** than others.



Sales volumns



Price distribution

CONCLUSION

The majority of sales volume comes from the lowest median price across all features, except for the fuel system.

Due to the widespread popularity of Multi-Point Fuel Injection (MPFI) systems, most mass-market cars now use this technology.

DATA PREPROCESSING (EDA)

Data preprocessing

CHECK MISSING VALUES

CHECK DUPLICATE

TRANSFORM DATATYPE
OBJECT TO INTERGER

DESCRIBE DATA

```
Chuyển tên cột thành chữ viết thường
.columns = df.columns.str.lower()

Delete the columns not means to analyst
.drop(columns=['car_id'], inplace= True)

Check missing value
int(f'Số dòng bị null là {df.isnull().sum()}') # => Không có null

Check Duplicates
int(f'Số dòng bị lặp lại là {df.duplicated().sum()}')

TRANSFORM Doornumber 'Object' to 'Int'
.price = df.price.astype('int64')

Transform column "Doornumber" to numberic (2,4) and column "cylindernumber" to numbe
.doornumber = df.doornumber.map({'two':2,'four':4})
.cylindernumber = df.cylindernumber.map({'two':2,'three':3,'four':4,
                                         'five':5,'six':6,'eight':8,'twelve':12})
```

01

Data preprocessing

INSIGHTS FROM STATISTIC

CAR VARIETY :

- MIN PRICE: 5,118 USD
- 75% PERCENTILE : 16,558 USD
- MAX PRICE: 17,859,167 USD
- MEAN PRICE : 102,468 USD

Remove outlier

```
# Calculate the 95% percentile of the price column
percentile_95 = df['price'].quantile(0.95)
percentile_05 = df['price'].quantile(0.05)
print(f'Giá trị phân vị đến 95 của cột price là {percentile_95}')
print(f'Giá trị phân vị đến 05 của cột price là {percentile_05}')
# If the values outside 95% percentile , we delete such rows with the price values upper 'percentile_90'
df = df[(df['price'] <= percentile_95) & (df['price'] >= percentile_05)]
```

✓ 0.0s

CAR BRAND :

QUANTITY OF CAR BRAND: 26

('ALFA-ROMERO', 'AUDI', 'BMW', 'CHEVROLET', 'DODGE', 'HONDA', 'ISUZU', 'JAGUAR', 'MAXDA', 'MAZDA', 'BUICK', 'MERCURY', 'MITSUBISHI', 'NISSAN', 'NISSAN', 'PEUGEOT', 'PLYMOUTH', 'PORSCHE', 'RENAULT', 'SAAB', 'SUBARU', 'TOYOTA', 'VOKSWAGEN', 'VOLKSWAGEN', 'VW', 'VOLVO')

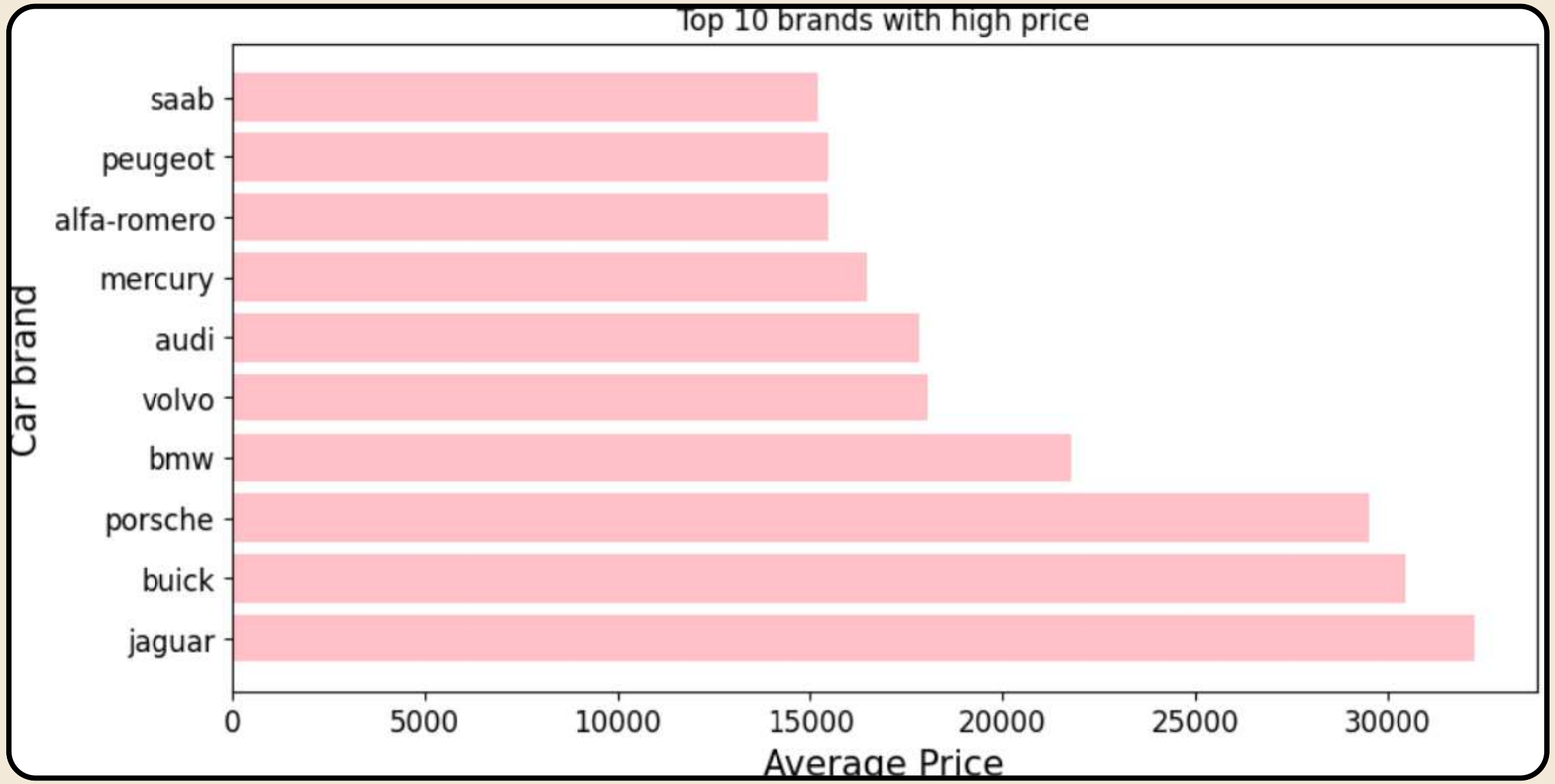
```
df['brand_name'] = df['carname'].str.split(" ",expand = True)[0]
df['brand_name'] = df['brand_name'].replace({"porcshce":"porsche","toyouta":"toyota"})
df['brand_name'].unique()
```

✓ 0.0s

```
array(['alfa-romero', 'audi', 'bmw', 'chevrolet', 'dodge', 'honda',
       'isuzu', 'jaguar', 'maxda', 'mazda', 'buick', 'mercury',
       'mitsubishi', 'Nissan', 'nissan', 'peugeot', 'plymouth', 'porsche',
       'renault', 'saab', 'subaru', 'toyota', 'vokswagen', 'volkswagen',
       'vw', 'volvo'], dtype=object)
```

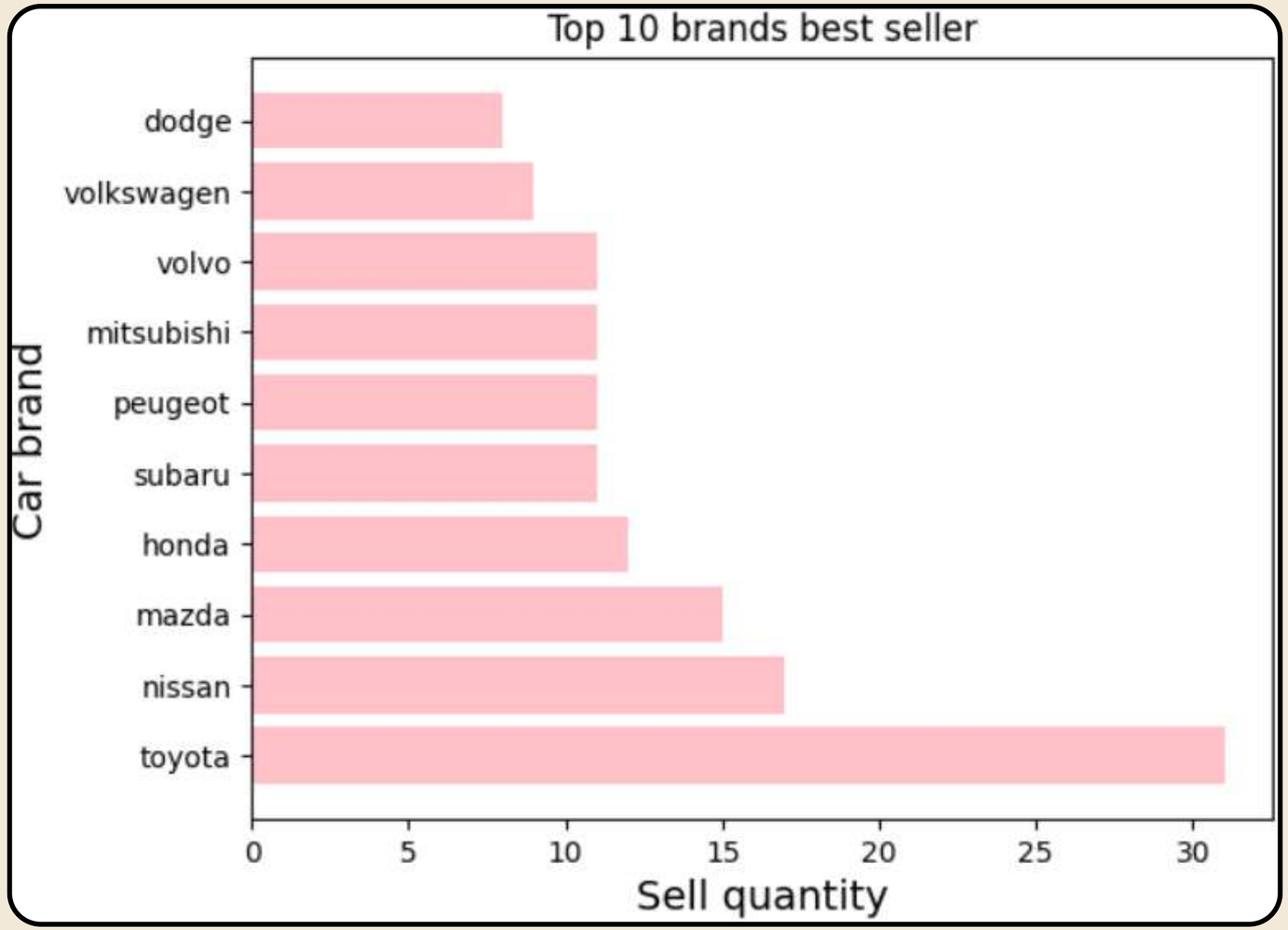
DATA PREPROCESSING

TOP 10 CAR BRANDS WITH HIGH PRICE



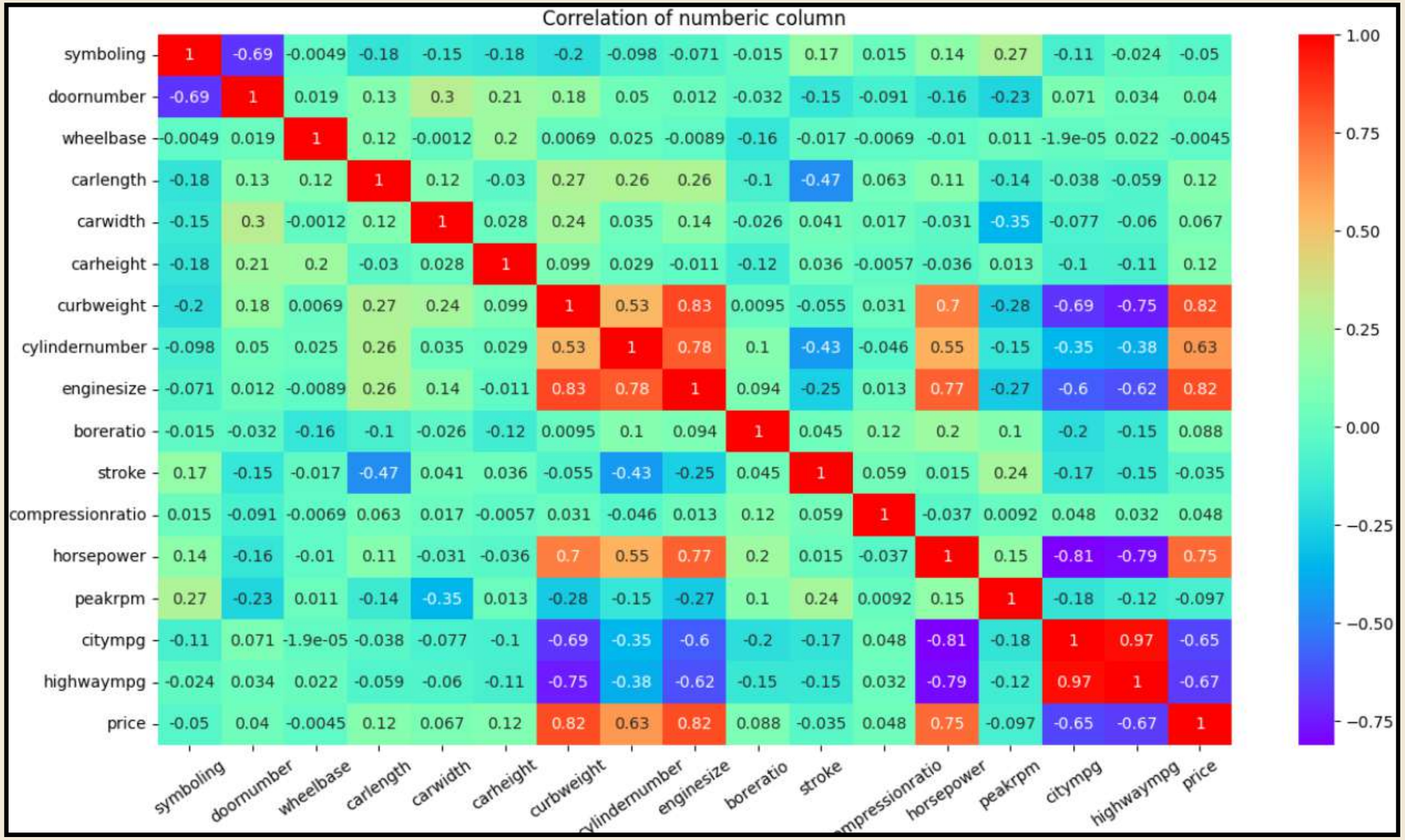
DATA PREPROCESSING

TOP 10 BEST SELLING CAR BRANDS



CORRELATION MAP

BIVARIATE ANALYSIS



CONCLUSION

POSITIVE CORRELATION:

Curb weight, cylinder number, engine size, and compression ratio all have **a significant positive relationship** with the price of a car.

NEGATIVE CORRELATION:

City MPG and highway MPG indices (indicating fuel consumption savings) both have **a negative relationship** with the price of a car.

CHOOSING A MODEL

Linear Regression

TRANSFORM DATA

ENCODING

```
# ## One hot encoding
df_copy = pd.get_dummies (df_copy, drop_first=False, dtype= int)
```

SCALE DATA

Train - Test (80 - 20)

```
ss = StandardScaler()
df_copy[['wheelbase', 'carlength', 'carwidth', 'carheight','curbweight', 'enginesize',
       'boreratio', 'stroke', 'compressionratio','horsepower', 'peakrpm', 'citympg',
       'cylindernumber','doornumber']] = ss.fit_transform(df_copy[['wheelbase',
       'carlength', 'carwidth', 'carheight', 'curbweight', 'enginesize',
       'boreratio', 'stroke', 'compressionratio','horsepower', 'peakrpm',
       'citympg','cylindernumber','doornumber']])
```

AFTER TRANSFORM DATA **37 columns and 183 rows**

TRAIN MODEL

Linear Regression

BUILD MODEL

```
lr = LinearRegression()
accuracy = []
for i in range(30,50):
    X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=0.2, random_state= i)
    lr.fit(X_train,y_train)
    y_pred = lr.predict(X_test)
    r2 = r2_score(y_test, y_pred)
    mse = mean_squared_error(y_test, y_pred)
    accuracy.append({'random_state':i, 'r2_score_1':r2,'mse_1':mse})
df_accurancy = pd.DataFrame(accuracy)
```

Notes:

- Train - test (80-20)
- Set random_state from 30 - 49

Linear Regression

RESULT

	random_state	r2_score_1	mse_1
0	30	0.44	10,559,207.51
1	31	0.79	4,846,155.65
2	32	0.82	11,192,228.46
3	33	0.82	6,361,892.48
4	34	0.87	5,183,347.77
5	35	0.85	5,579,087.56
6	36	0.92	4,763,525.01
7	37	0.83	7,549,816.18
8	38	0.86	6,500,921.64
9	39	0.72	11,999,663.15
10	40	0.89	4,563,367.53
11	41	0.74	6,330,725.02
12	42	0.87	5,193,780.54
13	43	0.89	4,771,337.75
14	44	0.55	10,316,896.64
15	45	0.86	5,486,432.60
16	46	0.86	4,807,984.10
17	47	0.67	8,553,587.06
18	48	0.83	5,933,242.56
19	49	0.64	14,042,367.28

- The best model, Linear Regression, achieved an accuracy of 92% with a train-test split and a random state of 36.

```
sns.scatterplot(x=y_pred,y=y_test)
plt.xlabel('Predictions')
plt.title('Evaluation of our LM model')
plt.show()
```

✓ 0.2s