
GRAND LEGIT AUTO
AN ANALYSIS OF VALUE
IN AUSTRALIAN VEHICLE PRICES


GRAND LEGIT AUTO
AN ANALYSIS OF VALUE
IN AUSTRALIAN VEHICLE PRICES


GRAND LEGIT AUTO


GRAND LEGIT AUTO

kaggle

GRAND LEGIT AUTO

kaggle

GRAND LEGIT AUTO


GRAND LEGIT AUTO

In [3]:
```
Supra_options = avp.loc[avp["Model"] == 'Supra']
Supra_options[['Year', 'Model', 'Title', 'Price']]
```

Out[3]:

	Year	Model	Title	Price
7588	2022.0	Supra	2022 Toyota Supra GTS +alcantara Seats	95888
8661	2021.0	Supra	2021 Toyota Supra GTS	97990
12246	2020.0	Supra	2020 Toyota Supra GTS	87990
12261	2021.0	Supra	2021 Toyota Supra GTS +alcantara Seats	94995
15491	2023.0	Supra	2023 Toyota Supra GTS +alcant Seats +matte Paint	102000
15670	1993.0	Supra	1993 Toyota Supra	69880
15971	1994.0	Supra	1994 Toyota Supra TURBO 1994	77888

In [5]:
```
avp.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 16734 entries, 0 to 16733
Data columns (total 19 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   Brand              16733 non-null  object 
 1   Year               16733 non-null  float64
 2   Model              16733 non-null  object 
 3   Car/Suv            16706 non-null  object 
 4   Title              16733 non-null  object 
 5   UsedOrNew          16733 non-null  object 
 6   Transmission       16733 non-null  object 
 7   Engine             16733 non-null  object 
 8   DriveType          16733 non-null  object 
 9   FuelType           16733 non-null  object 
 10  FuelConsumption    16733 non-null  object 
 11  Kilometres         16733 non-null  object 
 12  ColourExtInt       16733 non-null  object 
 13  Location           16284 non-null  object 
 14  CylindersinEngine  16733 non-null  object 
 15  BodyType           16452 non-null  object 
 16  Doors              15130 non-null  object 
 17  Seats              15029 non-null  object 
 18  Price              16731 non-null  object 
dtypes: float64(1), object(18)
memory usage: 2.4+ MB
```

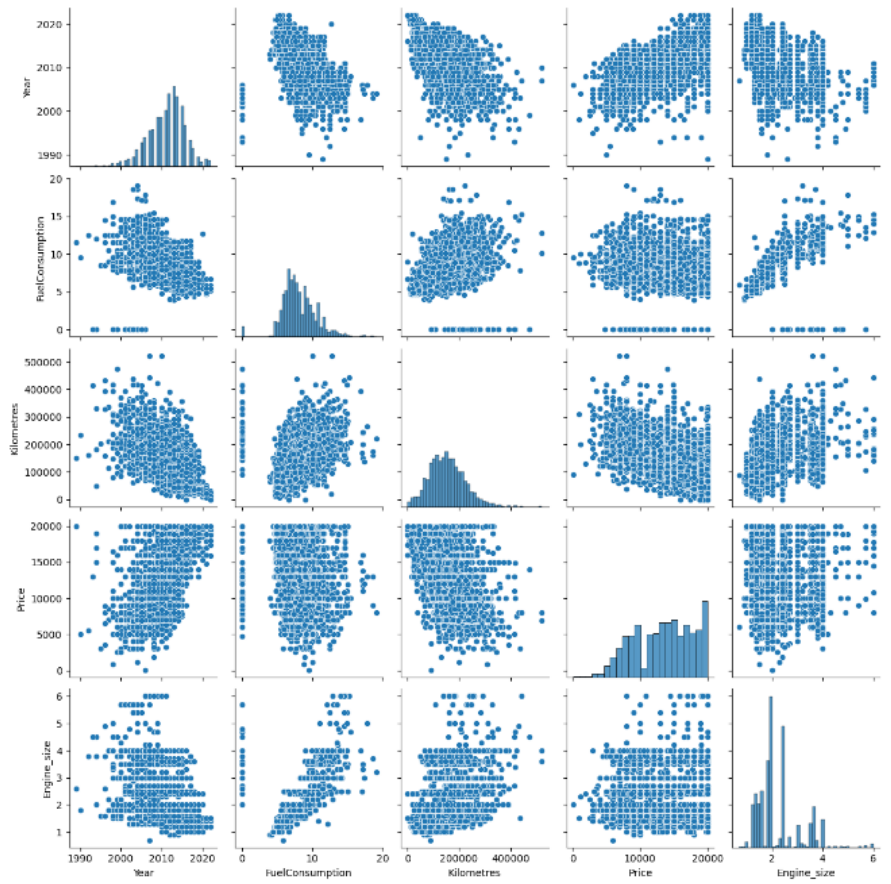
Value for Money

$$= \frac{\text{Price}}{\text{Fuel Consumption}}$$

GRAND LEGIT AUTO

```
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

```
sns.pairplot(avp)
plt.show()
```



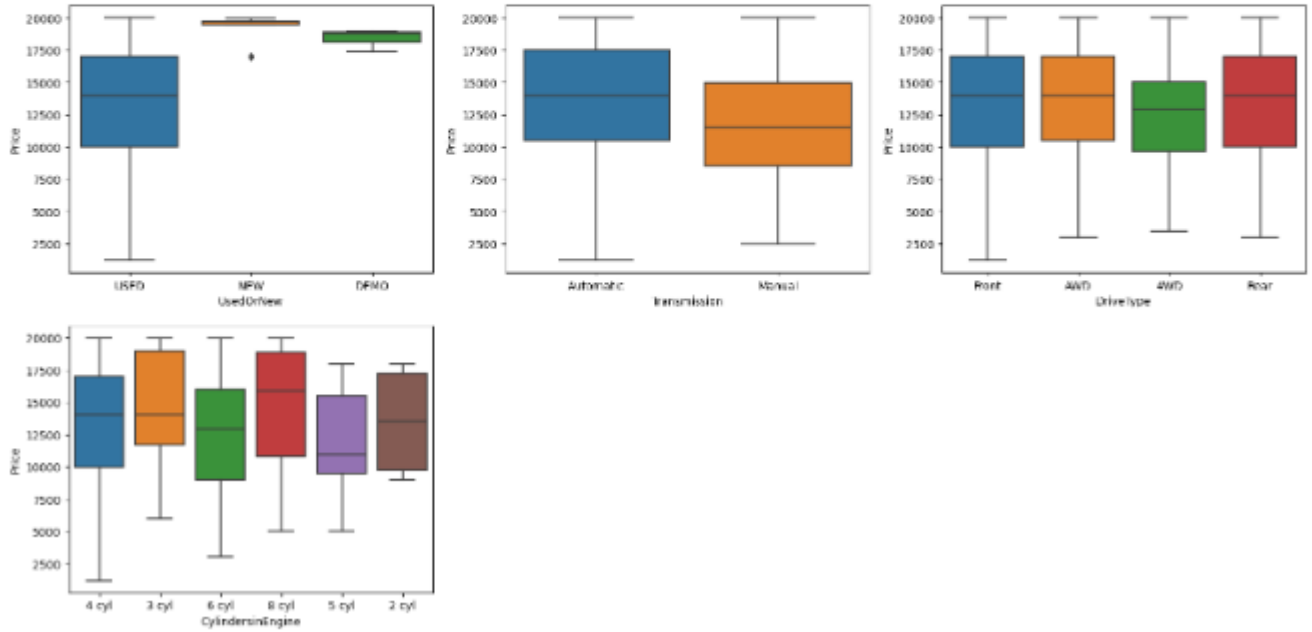
Grand Legit Auto

In [29]:
```
avp = avp.drop(avp[avp['Price'] < 1000].index)
```

In [30]:
```
plt.figure(figsize=(20, 20))

plt.subplot(4,3,1)
sns.boxplot(x = 'UsedOrNew', y = 'Price', data = avp)
plt.subplot(4,3,2)
sns.boxplot(x = 'Transmission', y = 'Price', data = avp)
plt.subplot(4,3,3)
sns.boxplot(x = 'DriveType', y = 'Price', data = avp)
plt.subplot(4,3,4)
sns.boxplot(x = 'CylindersinEngine', y = 'Price', data = avp)

plt.show()
```



Grand Legit Auto

1st Iteration

Fit OLS Regression Model

```
In [48]: import statsmodels.api as
```

```
In [49]: # Add constant term to th
         X_train_model1 = sm.add_c

         # Fit OLS model
         ols_model1 = sm.OLS(y_tra
         ols_results1 = ols_model1

         # Print summary of the OL
         print(ols_results1.summar
```

```
==========================
Dep. Variable:
Model:
Method:                 L
Date:                Wed,
Time:
No. Observations:
Df Residuals:
Df Model:
Covariance Type:
==========================

--------------------------
const                    .
log_Year
log_Kilometres
log_Engine_size
Transmission_Manual
DriveType_AWD
DriveType_Front
DriveType_Rear
CylindersinEngine_3 cyl
CylindersinEngine_4 cyl
CylindersinEngine_5 cyl
CylindersinEngine_6 cyl
CylindersinEngine_8 cyl
==========================
Omnibus:
Prob(Omnibus):
Skew:
Kurtosis:
==========================

Notes:
[1] Standard Errors assum
[2] The condition number
strong multicollinearity
```

2nd Iteration

```
In [53]: # Fit OLS model
         ols_model2 = sm
         ols_results2 = (

         # Print summary
         print(ols_resul
```

```
==========
Dep. Variable:
Model:
Method:
Date:
Time:
No. Observation:
Df Residuals:
Df Model:
Covariance Type
==========

----------
const
log_Year
log_Kilometres
log_Engine_size
Transmission_Ma
DriveType_AWD
DriveType_Front
DriveType_Rear
CylindersEngi
CylindersEngi
CylindersEngi
CylindersEngi
==========
Omnibus:
Prob(Omnibus):
Skew:
Kurtosis:
==========

Notes:
[1] Standard Er
[2] The conditi
strong multicol
```

3rd Iteration

```
In [55]: # Fit OLS model
         ols_model3 = sm.OLS(y_train, X_train_model3)
         ols_results3 = ols_model3.fit()

         # Print summary of the OLS regression results
         print(ols_results3.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:              log_Value   R-squared:                       0.700
Model:                            OLS   Adj. R-squared:                  0.699
Method:                 Least Squares   F-statistic:                     761.4
Date:                Wed, 01 May 2024   Prob (F-statistic):               0.00
Time:                        09:05:47   Log-Likelihood:                -213.18
No. Observations:                2291   AIC:                             442.4
Df Residuals:                    2283   BIC:                             488.2
Df Model:                           7
Covariance Type:            nonrobust
===========================================================================================
                              coef    std err          t      P>|t|      [0.025      0.975]
-------------------------------------------------------------------------------------------
const                   -1125.9484     24.227    -46.475      0.000   -1173.458   -1078.439
log_Year                  149.2425      3.178     46.965      0.000     143.011     155.474
log_Kilometres             -0.1407      0.012    -12.215      0.000      -0.163      -0.118
log_Engine_size            -0.2922      0.027    -10.840      0.000      -0.345      -0.239
Transmission_Manual        -0.0735      0.014     -5.093      0.000      -0.102      -0.045
DriveType_Front            -0.0806      0.015     -5.212      0.000      -0.111      -0.050
CylindersinEngine_4 cyl     0.0405      0.018      2.244      0.025       0.005       0.076
CylindersinEngine_8 cyl     0.2408      0.053      4.515      0.000       0.136       0.345
==============================================================================
Omnibus:                      267.973   Durbin-Watson:                   2.010
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              781.348
Skew:                          -0.616   Prob(JB):                    2.15e-170
Kurtosis:                       5.583   Cond. No.                     6.22e+04
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 6.22e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Variance Inflation Factor

Out[50]:

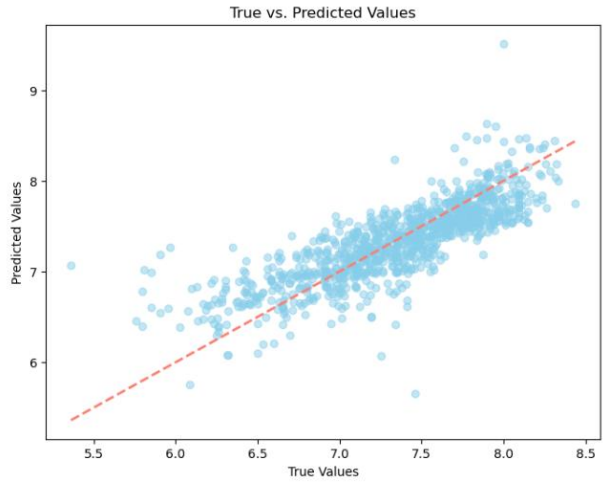
	Features	VIF
0	const	19404227.17
9	CylindersinEngine_4 cyl	399.38
11	CylindersinEngine_6 cyl	350.37
8	CylindersinEngine_3 cyl	50.24
12	CylindersinEngine_8 cyl	29.45
10	CylindersinEngine_5 cyl	9.09
3	log_Engine_size	4.08
6	DriveType_Front	3.88
7	DriveType_Rear	2.49
5	DriveType_AWD	2.05
2	log_Kilometres	1.50
1	log_Year	1.48
4	Transmission_Manual	1.05

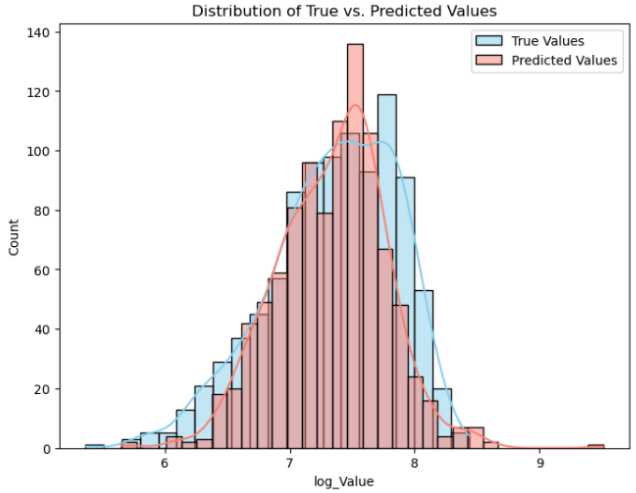
Correlation Coefficient

Out[51]:

pairs	cc
(CylindersinEngine_4 cyl, CylindersinEngine_6 cyl)	0.888923
(log_Engine_size, CylindersinEngine_6 cyl)	0.715344
(CylindersinEngine_4 cyl, log_Engine_size)	0.658813
(DriveType_Front, log_Engine_size)	0.615525
(DriveType_Front, DriveType_Rear)	0.609671
(log_Kilometres, log_Year)	0.506714

Train Model Summary:
Mean Squared Error (MSE) on training data: 0.07052568163676846
R-squared on training data: 0.700096565076077





Test Model Summary:
Mean Squared Error on testing data: 0.2995668811992725
R-squared: 0.6596185666202541

Grand Legit Auto

Findings

1. **Year of Manufacture**

2. **Engine Size**

Further findings

- Avoid manual
- Avoid Front Wheel Drive
- 4 cylinder vehicles

3rd Iteration

In [55]:
```
# Fit OLS model
ols_model3 = sm.OLS(y_train, X_train_model3)
ols_results3 = ols_model3.fit()

# Print summary of the OLS regression results
print(ols_results3.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:              log_Value   R-squared:                       0.700
Model:                            OLS   Adj. R-squared:                  0.699
Method:                 Least Squares   F-statistic:                     761.4
Date:                Wed, 01 May 2024   Prob (F-statistic):               0.00
Time:                        09:05:47   Log-Likelihood:                -213.18
No. Observations:                2291   AIC:                             442.4
Df Residuals:                    2283   BIC:                             488.2
Df Model:                           7
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
const                        -1125.9484     24.227    -46.475      0.000   -1173.458   -1078.439
log_Year                       149.2425      3.178     46.965      0.000     143.011     155.474
log_Kilometres                  -0.1407      0.012    -12.215      0.000      -0.163      -0.118
log_Engine_size                 -0.2922      0.027    -10.840      0.000      -0.345      -0.239
Transmission_Manual             -0.0735      0.014     -5.093      0.000      -0.102      -0.045
DriveType_Front                 -0.0806      0.015     -5.212      0.000      -0.111      -0.050
CylindersinEngine_4 cyl          0.0405      0.018      2.244      0.025       0.005       0.076
CylindersinEngine_8 cyl          0.2408      0.053      4.515      0.000       0.136       0.345
==============================================================================
Omnibus:                      267.973   Durbin-Watson:                   2.010
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              781.348
Skew:                          -0.616   Prob(JB):                    2.15e-170
Kurtosis:                       5.583   Cond. No.                     6.22e+04
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 6.22e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Grand Legit Auto


GRAND LEGIT AUTO