EDA and PREDICTION OF CAR PRICE

SACHIN JAYARAMAN

DATA 603 FINAL PROJECT

SPRING ‘22

DATA ACQUISITION

- Data is gathered from Kaggle(https://www.kaggle.com/CooperUnion/cardataset).
- It has data about the model, make, year of make, Horsepower, type of vehicle, mpg, popularity , and price of the vehicle.
- Data has 11914 records with 16 parameters.
- Data ranges from the year 1990 – 2017
- Software used: Google Colab

Understanding dataset

```
df.head()
```

	Make	Model	Year	Engine Fuel Type	Engine HP	Engine Cylinders	Transmission Type	Driven_Wheels	Number of Doors	Market Category	Vehicle Size	Vehicle Style	highway MPG	city mpg	Popularity	MSRP
0	BMW	1 Series M	2011	premium unleaded (required)	335.0	6.0	MANUAL	rear wheel drive	2.0	Factory Tuner,Luxury,High-Performance	Compact	Coupe	26	19	3916	46135
1	BMW	1 Series	2011	premium unleaded (required)	300.0	6.0	MANUAL	rear wheel drive	2.0	Luxury,Performance	Compact	Convertible	28	19	3916	40650
2	BMW	1 Series	2011	premium unleaded (required)	300.0	6.0	MANUAL	rear wheel drive	2.0	Luxury,High-Performance	Compact	Coupe	28	20	3916	36350
3	BMW	1 Series	2011	premium unleaded (required)	230.0	6.0	MANUAL	rear wheel drive	2.0	Luxury,Performance	Compact	Coupe	28	18	3916	29450
4	BMW	1 Series	2011	premium unleaded (required)	230.0	6.0	MANUAL	rear wheel drive	2.0	Luxury	Compact	Convertible	28	18	3916	34500

Understanding dataset

```
[ ]  df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11914 entries, 0 to 11913
Data columns (total 16 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   Make               11914 non-null  object
 1   Model              11914 non-null  object
 2   Year               11914 non-null  int64
 3   Engine Fuel Type   11911 non-null  object
 4   Engine HP          11845 non-null  float64
 5   Engine Cylinders   11884 non-null  float64
 6   Transmission Type  11914 non-null  object
 7   Driven_Wheels      11914 non-null  object
 8   Number of Doors    11908 non-null  float64
 9   Market Category    8172 non-null   object
 10  Vehicle Size       11914 non-null  object
 11  Vehicle Style      11914 non-null  object
 12  highway MPG        11914 non-null  int64
 13  city mpg           11914 non-null  int64
 14  Popularity         11914 non-null  int64
 15  MSRP               11914 non-null  int64
```

```
[144] sales_df.summary().show()

+-------+-----+-----------------+----------------+---------------+-----------------+-----------------+--------------
|summary| Make|            Model|            Year|Engine Fuel Type|        Engine HP|  Engine Cylinders|Transmission Ty
+-------+-----+-----------------+----------------+---------------+-----------------+-----------------+--------------
|  count|11826|            11826|           11826|          11823|            11826|            11803|            11
|   mean| null| 745.5822222222222|2010.350752579063|           null|249.54058853373922| 5.656273828687622|             nu
| stddev| null|1490.8280590623795|7.593794497283969|           null|  109.205971004154|1.7432296450086233|             nu
|    min|Acura|         1 Series|            1990|         diesel|               55|                3| AUTOMATED_MANU
|    25%| null|             86.0|            2007|           null|              170|                4|             nu
|    50%| null|            500.0|            2015|           null|              227|                6|             nu
|    75%| null|            900.0|            2016|           null|              300|                6|             nu
|    max|Volvo|               xD|            2017|regular unleaded|             1001|               16|          UNKNO
+-------+-----+-----------------+----------------+---------------+-----------------+-----------------+--------------
```

Cleaning dataset

```
Make                   0
Model                  0
Year                   0
Engine Fuel Type       3
Engine HP             69
Engine Cylinders      30
Transmission Type      0
Driven_Wheels          0
Number of Doors        6
Vehicle Size           0
Vehicle Style          0
highway MPG            0
city mpg               0
Popularity             0
MSRP                   0
dtype: int64
```

```
df2 = df1.dropna()
df2.isnull().sum()
```

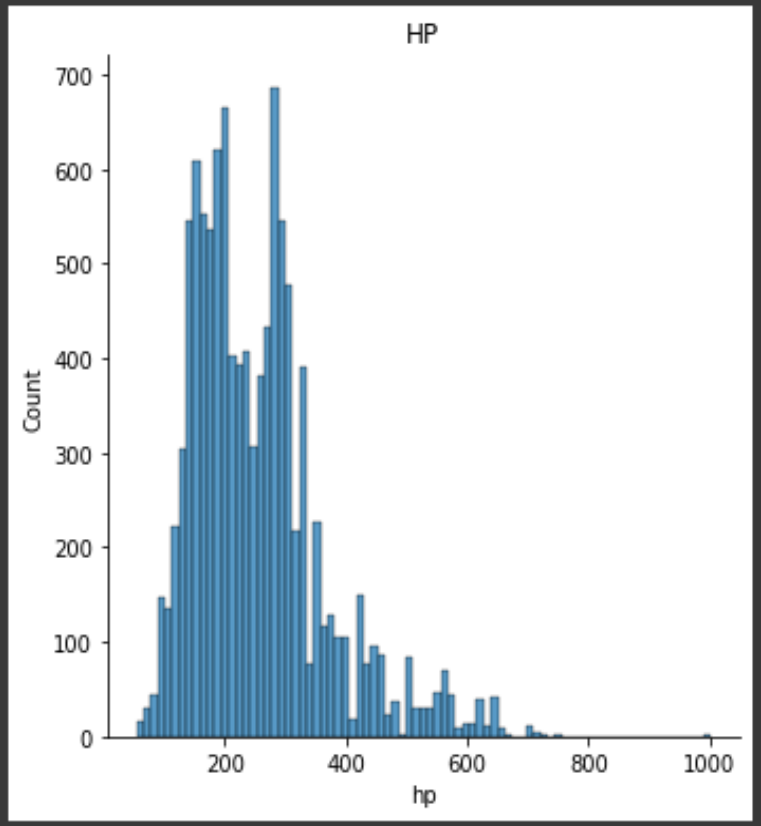
```
Make                   0
Model                  0
Year                   0
Engine Fuel Type       0
Engine HP              0
Engine Cylinders       0
Transmission Type      0
Driven_Wheels          0
Number of Doors        0
Vehicle Size           0
Vehicle Style          0
highway MPG            0
city mpg               0
Popularity             0
MSRP                   0
dtype: int64
```

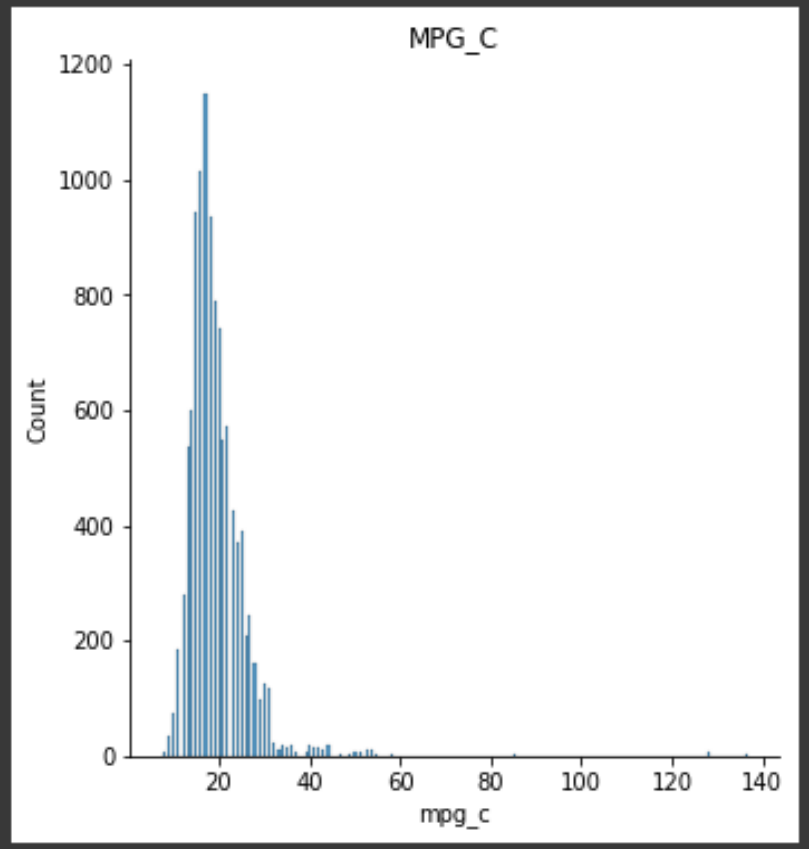
(11914, 15)

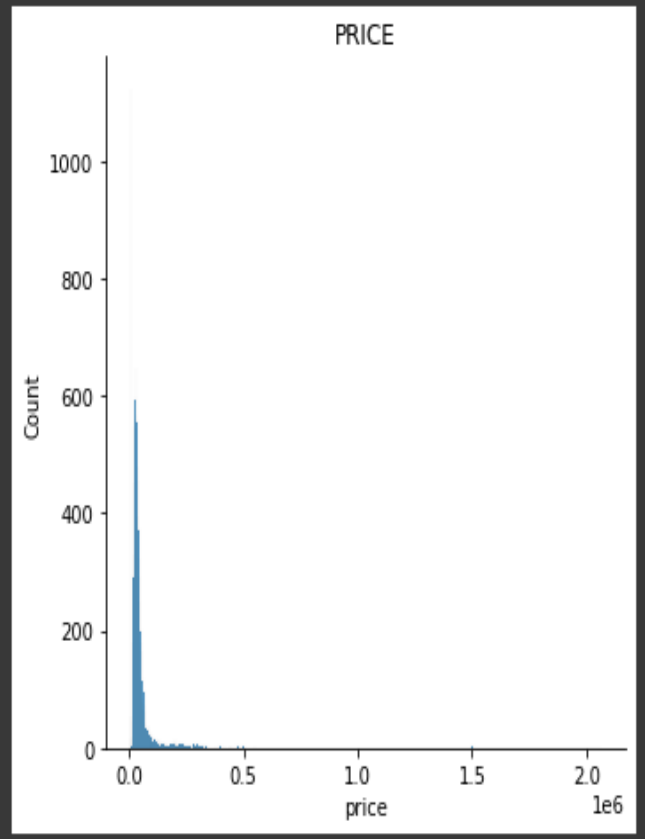
	Make	Model	Year	Engine Fuel Type	Engine HP	Engine Cylinders	Transmission Type	Driven_Wheels	Number of Doors	Vehicle Size	Vehicle Style	highway MPG
0	BMW	1 Series M	2011	premium unleaded (required)	335.0	6.0	MANUAL	rear wheel drive	2.0	Compact	Coupe	26
1	BMW	1 Series	2011	premium unleaded (required)	300.0	6.0	MANUAL	rear wheel drive	2.0	Compact	Convertible	28
2	BMW	1 Series	2011	premium unleaded (required)	300.0	6.0	MANUAL	rear wheel drive	2.0	Compact	Coupe	28
3	BMW	1 Series	2011	premium unleaded (required)	230.0	6.0	MANUAL	rear wheel drive	2.0	Compact	Coupe	28
4	BMW	1 Series	2011	premium unleaded (required)	230.0	6.0	MANUAL	rear wheel drive	2.0	Compact	Convertible	28

Heatmap – Correlation analysis

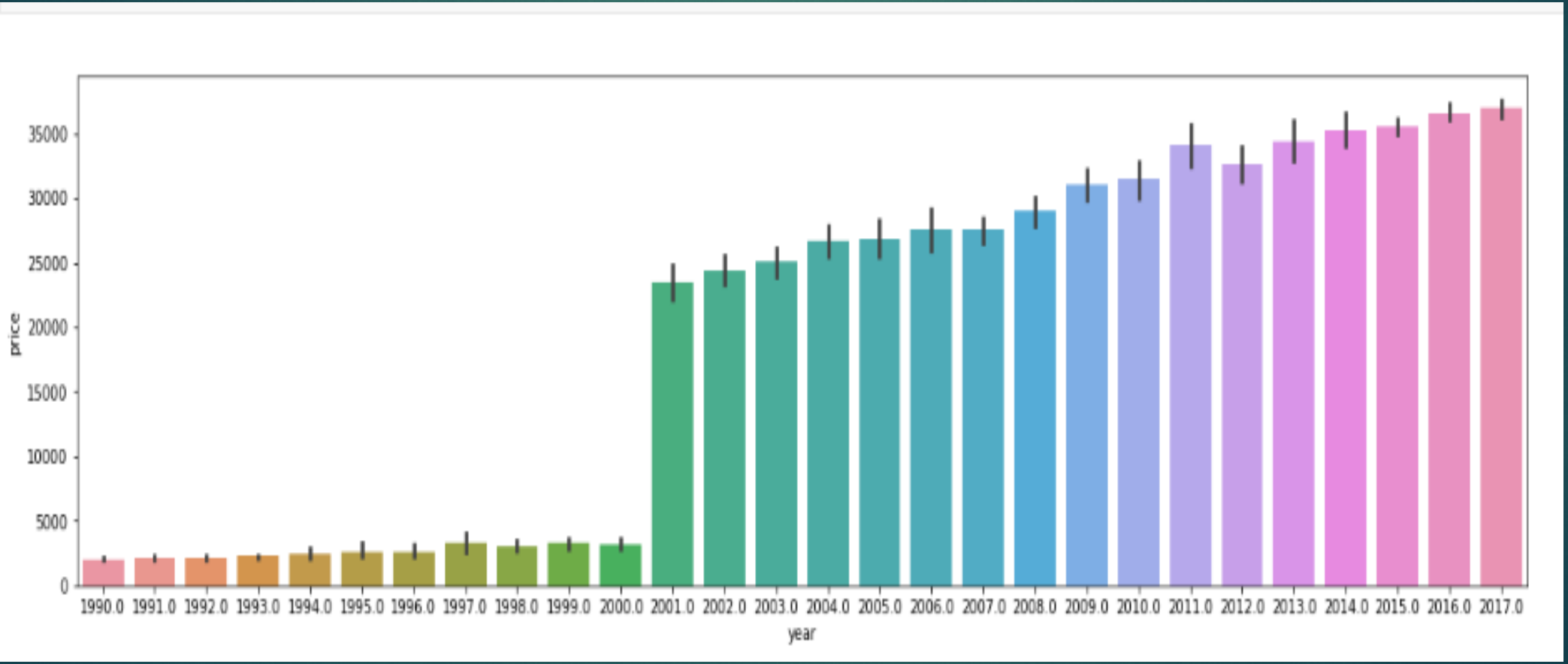
Correlating data distribution



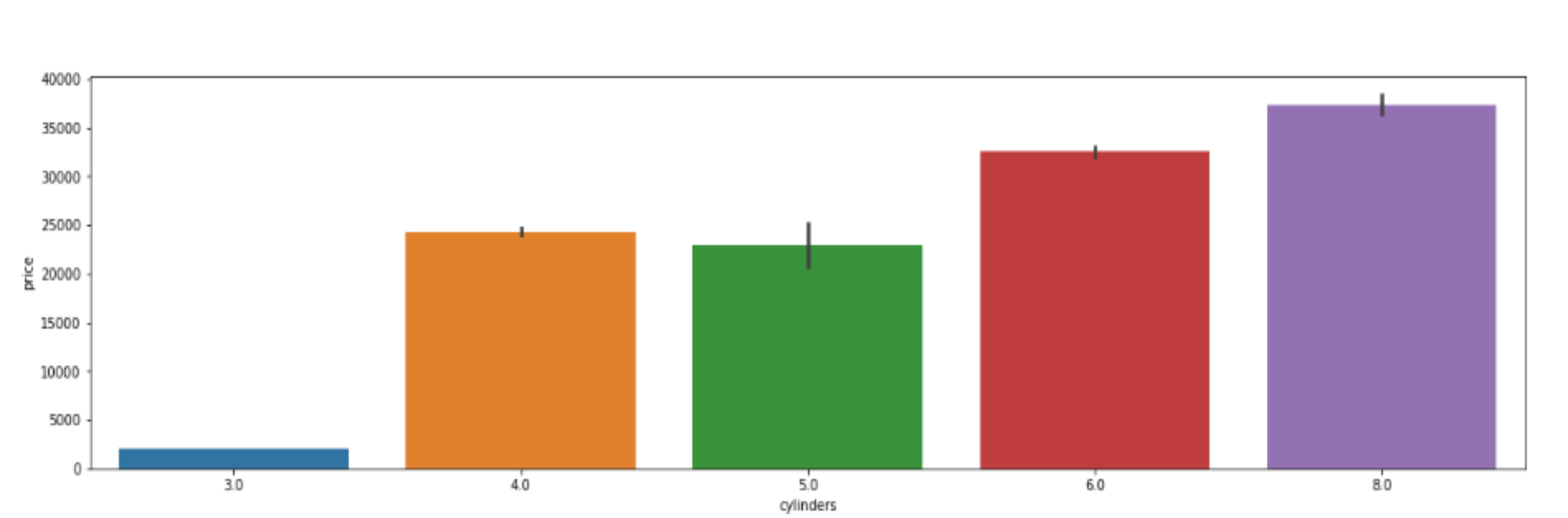




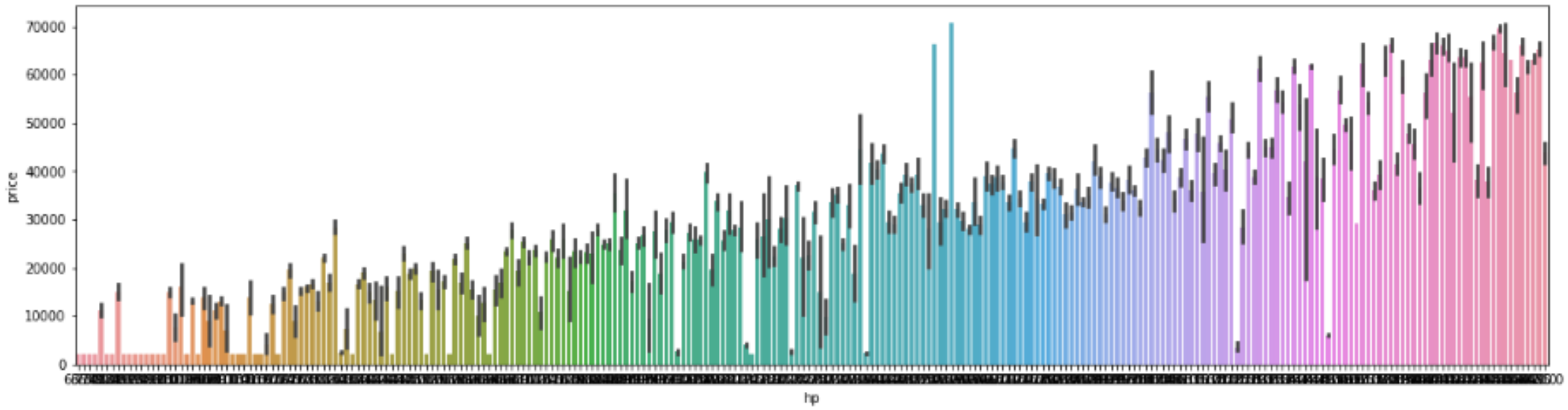
Year of Make Vs Price



Cylinder vs Price



Hp Vs Price



Basic EDA

```
[149] data_clean.groupBy("Engine Cylinders").count().show()
```

Engine Cylinders	count
12	230
null	111
6	4473
16	3
3	30
5	225
4	4743
8	2031
10	68

```
data_clean.groupBy("Engine HP").count().show()
```

Engine HP	count
148	94
540	15
580	5
137	16
451	1
251	6
255	56
296	17
133	4
322	11
78	8
321	18
362	10
597	2
375	19
155	156
108	31
193	12
520	2

```
[174] data_clean.groupBy("Number of Doors").count().show()
```

Number of Doors	count
null	89
3	395
4	8273
2	3157

Linear Regression

- Linear regression model to identify whether to be able to identify price of the car in correspondence with the Year, Cylinder, and MPG

```
model_df = scaledData.select("ScaledAttributes","MSRP")
training_df, test_df = model_df.randomSplit([0.8,0.2])
print("Count of training data: ", training_df.count())
print("Count of testing data: ", test_df.count())
```

```
Count of training data:  9491
Count of testing data:  2311
```

```
#Evaluate the results with the test data

test_results = lr_model.evaluate(test_df)

print("RMSE: {}".format(test_results.rootMeanSquaredError))
print("MSE: {}".format(test_results.meanSquaredError))
print("R2: {}".format(test_results.r2))
```

```
RMSE: 70928.52875448742
MSE: 5030856191.276148
R2: 0.28671970828763793
```