In [29]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.simplefilter("ignore")
%matplotlib inline
```

In [3]:
```
vehicle = pd.read_csv("auto-mpg.csv")
vehicle
```

Out[3]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin	car name
0	18.0	8	307.0	130	3504	12.0	70	1	chevrolet chevelle malibu
1	15.0	8	350.0	165	3693	11.5	70	1	buick skylark 320
2	18.0	8	318.0	150	3436	11.0	70	1	plymouth satellite
3	16.0	8	304.0	150	3433	12.0	70	1	amc rebel sst
4	17.0	8	302.0	140	3449	10.5	70	1	ford torino
...	...	...	...	...	...	...	...	...	...
393	27.0	4	140.0	86	2790	15.6	82	1	ford mustang gl
394	44.0	4	97.0	52	2130	24.6	82	2	vw pickup
395	32.0	4	135.0	84	2295	11.6	82	1	dodge rampage
396	28.0	4	120.0	79	2625	18.6	82	1	ford ranger
397	31.0	4	119.0	82	2720	19.4	82	1	chevy s-10

398 rows × 9 columns

In [4]:
```
vehicle.describe()
```

Out[4]:

	mpg	cylinders	displacement	weight	acceleration	model year	origin
count	398.000000	398.000000	398.000000	398.000000	398.000000	398.000000	398.000000
mean	23.514573	5.454774	193.425879	2970.424623	15.568090	76.010050	1.572864
std	7.815984	1.701004	104.269838	846.841774	2.757689	3.697627	0.802055
min	9.000000	3.000000	68.000000	1613.000000	8.000000	70.000000	1.000000
25%	17.500000	4.000000	104.250000	2223.750000	13.825000	73.000000	1.000000
50%	23.000000	4.000000	148.500000	2803.500000	15.500000	76.000000	1.000000
75%	29.000000	8.000000	262.000000	3608.000000	17.175000	79.000000	2.000000
max	46.600000	8.000000	455.000000	5140.000000	24.800000	82.000000	3.000000

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

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 398 entries, 0 to 397
Data columns (total 9 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   mpg           398 non-null    float64
 1   cylinders     398 non-null    int64  
 2   displacement  398 non-null    float64
 3   horsepower    398 non-null    object 
 4   weight        398 non-null    int64  
 5   acceleration  398 non-null    float64
 6   model year    398 non-null    int64  
 7   origin        398 non-null    int64  
 8   car name      398 non-null    object 
dtypes: float64(3), int64(4), object(2)
memory usage: 28.1+ KB
```

In [6]:
```
vehicle = vehicle.drop('car name',axis=1)
```

In [7]:
```
vehicle
```

Out[7]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin
0	18.0	8	307.0	130	3504	12.0	70	1
1	15.0	8	350.0	165	3693	11.5	70	1
2	18.0	8	318.0	150	3436	11.0	70	1
3	16.0	8	304.0	150	3433	12.0	70	1
4	17.0	8	302.0	140	3449	10.5	70	1
...	...	...	...	...	...	...	...	...
393	27.0	4	140.0	86	2790	15.6	82	1
394	44.0	4	97.0	52	2130	24.6	82	2
395	32.0	4	135.0	84	2295	11.6	82	1
396	28.0	4	120.0	79	2625	18.6	82	1
397	31.0	4	119.0	82	2720	19.4	82	1

398 rows × 8 columns

In [9]:
```
vehicle = vehicle.drop_duplicates()
vehicle
```

Out[9]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin
0	18.0	8	307.0	130	3504	12.0	70	1
1	15.0	8	350.0	165	3693	11.5	70	1
2	18.0	8	318.0	150	3436	11.0	70	1
3	16.0	8	304.0	150	3433	12.0	70	1
4	17.0	8	302.0	140	3449	10.5	70	1
...	...	...	...	...	...	...	...	...
393	27.0	4	140.0	86	2790	15.6	82	1
394	44.0	4	97.0	52	2130	24.6	82	2
395	32.0	4	135.0	84	2295	11.6	82	1
396	28.0	4	120.0	79	2625	18.6	82	1
397	31.0	4	119.0	82	2720	19.4	82	1

398 rows × 8 columns

In [10]:
```
vehicle.cylinders.unique()
```

Out[10]:
```
array([8, 4, 6, 3, 5], dtype=int64)
```

In [11]:
```
vehicle['horsepower'].unique()
```

Out[11]:
```
array(['130', '165', '150', '140', '198', '220', '215', '225', '190',
       '170', '160', '95', '97', '85', '88', '46', '87', '90', '113',
       '200', '210', '193', '?', '100', '105', '175', '153', '180', '110',
       '72', '86', '70', '76', '65', '69', '60', '80', '54', '208', '155',
       '112', '92', '145', '137', '158', '167', '94', '107', '230', '49',
       '75', '91', '122', '67', '83', '78', '52', '61', '93', '148',
       '129', '96', '71', '98', '115', '53', '81', '79', '120', '152',
       '102', '108', '68', '58', '149', '89', '63', '48', '66', '139',
       '103', '125', '133', '138', '135', '142', '77', '62', '132', '84',
       '64', '74', '116', '82'], dtype=object)
```

In [13]:
```
vehicle = vehicle.loc[vehicle['horsepower'] != '?',:]
vehicle
```

Out[13]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin
0	18.0	8	307.0	130	3504	12.0	70	1
1	15.0	8	350.0	165	3693	11.5	70	1
2	18.0	8	318.0	150	3436	11.0	70	1
3	16.0	8	304.0	150	3433	12.0	70	1
4	17.0	8	302.0	140	3449	10.5	70	1
...	...	...	...	...	...	...	...	...
393	27.0	4	140.0	86	2790	15.6	82	1
394	44.0	4	97.0	52	2130	24.6	82	2
395	32.0	4	135.0	84	2295	11.6	82	1
396	28.0	4	120.0	79	2625	18.6	82	1
397	31.0	4	119.0	82	2720	19.4	82	1

392 rows × 8 columns

In [14]:
```
vehicle.describe()
```

Out[14]:

	mpg	cylinders	displacement	weight	acceleration	model year	origin
count	392.000000	392.000000	392.000000	392.000000	392.000000	392.000000	392.000000
mean	23.445918	5.471939	194.411990	2977.584184	15.541327	75.979592	1.576531
std	7.805007	1.705783	104.644004	849.402560	2.758864	3.683737	0.805518
min	9.000000	3.000000	68.000000	1613.000000	8.000000	70.000000	1.000000
25%	17.000000	4.000000	105.000000	2225.250000	13.775000	73.000000	1.000000
50%	22.750000	4.000000	151.000000	2803.500000	15.500000	76.000000	1.000000
75%	29.000000	8.000000	275.750000	3614.750000	17.025000	79.000000	2.000000
max	46.600000	8.000000	455.000000	5140.000000	24.800000	82.000000	3.000000

In [15]:
```
vehicle['horsepower'] = vehicle['horsepower'].replace('?',np.nan)
vehicle['horsepower'] = vehicle['horsepower'].ffill()
```

In [16]:
```
vehicle['horsepower'].unique()
```

Out[16]:
```
array(['130', '165', '150', '140', '198', '220', '215', '225', '190',
       '170', '160', '95', '97', '85', '88', '46', '87', '90', '113',
       '200', '210', '193', '100', '105', '175', '153', '180', '110',
       '72', '86', '70', '76', '65', '69', '60', '80', '54', '208', '155',
       '112', '92', '145', '137', '158', '167', '94', '107', '230', '49',
       '75', '91', '122', '67', '83', '78', '52', '61', '93', '148',
       '129', '96', '71', '98', '115', '53', '81', '79', '120', '152',
       '102', '108', '68', '58', '149', '89', '63', '48', '66', '139',
       '103', '125', '133', '138', '135', '142', '77', '62', '132', '84',
       '64', '74', '116', '82'], dtype=object)
```

In [17]:
```
vehicle.horsepower = vehicle.horsepower.astype(int)
```

In [18]:
```
vehicle.dtypes
```

Out[18]:
```
mpg             float64
cylinders         int64
displacement    float64
horsepower        int32
weight            int64
acceleration    float64
model year        int64
origin            int64
dtype: object
```

In [19]:
```
vehicle.isna().sum()
```

Out[19]:
```
mpg             0
cylinders       0
displacement    0
horsepower      0
weight          0
acceleration    0
model year      0
origin          0
dtype: int64
```

Outlier Detection

In [20]:
```
# column - mpg
q3 = vehicle.mpg.quantile(0.75)
q1 = vehicle.mpg.quantile(0.25)
iqr = q3-q1
upper_cut = q3 + (1.5*iqr)
lower_cut = q1 - (1.5*iqr)
upper_cut,lower_cut
```

Out[20]:
```
(47.0, -1.0)
```

In [21]:
```
# column - cylinders
q3 = vehicle.cylinders.quantile(0.75)
q1 = vehicle.cylinders.quantile(0.25)
iqr = q3-q1
upper_cut = q3 + (1.5*iqr)
lower_cut = q1 - (1.5*iqr)
upper_cut,lower_cut
```

Out[21]:
```
(14.0, -2.0)
```

In [22]:
```
# column - displacement
q3 = vehicle.displacement.quantile(0.75)
q1 = vehicle.displacement.quantile(0.25)
iqr = q3-q1
upper_cut = q3 + (1.5*iqr)
lower_cut = q1 - (1.5*iqr)
upper_cut,lower_cut
```

Out[22]: (531.875, -151.125)

In [23]:
```
# column - horsepower
q3 = vehicle.horsepower.quantile(0.75)
q1 = vehicle.horsepower.quantile(0.25)
iqr = q3-q1
upper_cut = q3 + (1.5*iqr)
lower_cut = q1 - (1.5*iqr)
upper_cut,lower_cut
vehicle['horsepower'] = vehicle['horsepower'].clip(lower_cut,upper_cut)
```

In [24]:
```
# column - weight
q3 = vehicle.weight.quantile(0.75)
q1 = vehicle.weight.quantile(0.25)
iqr = q3-q1
upper_cut = q3 + (1.5*iqr)
lower_cut = q1 - (1.5*iqr)
upper_cut,lower_cut
```

Out[24]: (5699.0, 141.0)

In [25]:
```
# column - acceleration
q3 = vehicle.acceleration.quantile(0.75)
q1 = vehicle.acceleration.quantile(0.25)
iqr = q3-q1
upper_cut = q3 + (1.5*iqr)
lower_cut = q1 - (1.5*iqr)
upper_cut,lower_cut
vehicle['acceleration'] = vehicle['acceleration'].clip(lower_cut,upper_cut)
```

In [26]:
```
vehicle = vehicle.rename(columns = {'model year':'model_year'})
```

In [27]:
```
# column - model year
q3 = vehicle.model_year.quantile(0.75)
q1 = vehicle.model_year.quantile(0.25)
iqr = q3-q1
upper_cut = q3 + (1.5*iqr)
lower_cut = q1 - (1.5*iqr)
upper_cut,lower_cut
```

Out[27]:
```
(88.0, 64.0)
```

In [28]:
```
vehicle.columns
```

Out[28]:
```
Index(['mpg', 'cylinders', 'displacement', 'horsepower', 'weight',
       'acceleration', 'model_year', 'origin'],
      dtype='object')
```

In [ ]: