Seatwork 11.1 Exploratory Data Analysis for Machine Learning

Name: Xander Sam E. Galapia

Section: CPE22S3

```
!pip install hvplot
```

```
Requirement already satisfied: hvplot in /usr/local/lib/python3.10/dist-packages (0.9.2)
Requirement already satisfied: bokeh>=1.0.0 in /usr/local/lib/python3.10/dist-packages (from hvplot) (3.3.4)
Requirement already satisfied: colorcet>=2 in /usr/local/lib/python3.10/dist-packages (from hvplot) (3.1.0)
Requirement already satisfied: holoviews>=1.11.0 in /usr/local/lib/python3.10/dist-packages (from hvplot) (1.17.1)
Requirement already satisfied: pandas in /usr/local/lib/python3.10/dist-packages (from hvplot) (2.0.3)
Requirement already satisfied: numpy>=1.15 in /usr/local/lib/python3.10/dist-packages (from hvplot) (1.25.2)
Requirement already satisfied: packaging in /usr/local/lib/python3.10/dist-packages (from hvplot) (24.0)
Requirement already satisfied: panel>=0.11.0 in /usr/local/lib/python3.10/dist-packages (from hvplot) (1.3.8)
Requirement already satisfied: param<3.0,>=1.12.0 in /usr/local/lib/python3.10/dist-packages (from hvplot) (2.1.0)
Requirement already satisfied: Jinja2>=2.9 in /usr/local/lib/python3.10/dist-packages (from bokeh>=1.0.0->hvplot) (3.1.3)
Requirement already satisfied: contourpy>=1 in /usr/local/lib/python3.10/dist-packages (from bokeh>=1.0.0->hvplot) (1.2.1)
Requirement already satisfied: pillow>=7.1.0 in /usr/local/lib/python3.10/dist-packages (from bokeh>=1.0.0->hvplot) (9.4.0)
Requirement already satisfied: PyYAML>=3.10 in /usr/local/lib/python3.10/dist-packages (from bokeh>=1.0.0->hvplot) (6.0.1)
Requirement already satisfied: tornado>=5.1 in /usr/local/lib/python3.10/dist-packages (from bokeh>=1.0.0->hvplot) (6.3.3)
Requirement already satisfied: xyzservices>=2021.09.1 in /usr/local/lib/python3.10/dist-packages (from bokeh>=1.0.0->hvplot) (2024.4.0)
Requirement already satisfied: pyviz-comms>=0.7.4 in /usr/local/lib/python3.10/dist-packages (from holoviews>=1.11.0->hvplot) (3.0.2)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.10/dist-packages (from pandas->hvplot) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas->hvplot) (2023.4)
Requirement already satisfied: tzdata>=2022.1 in /usr/local/lib/python3.10/dist-packages (from pandas->hvplot) (2024.1)
Requirement already satisfied: markdown in /usr/local/lib/python3.10/dist-packages (from panel>=0.11.0->hvplot) (3.6)
Requirement already satisfied: markdown-it-py in /usr/local/lib/python3.10/dist-packages (from panel>=0.11.0->hvplot) (3.0.0)
Requirement already satisfied: linkify-it-py in /usr/local/lib/python3.10/dist-packages (from panel>=0.11.0->hvplot) (2.0.3)
Requirement already satisfied: mdit-py-plugins in /usr/local/lib/python3.10/dist-packages (from panel>=0.11.0->hvplot) (0.4.0)
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-packages (from panel>=0.11.0->hvplot) (2.31.0)
Requirement already satisfied: tqdm>=4.48.0 in /usr/local/lib/python3.10/dist-packages (from panel>=0.11.0->hvplot) (4.66.2)
Requirement already satisfied: bleach in /usr/local/lib/python3.10/dist-packages (from panel>=0.11.0->hvplot) (6.1.0)
Requirement already satisfied: typing-extensions in /usr/local/lib/python3.10/dist-packages (from panel>=0.11.0->hvplot) (4.11.0)
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.10/dist-packages (from Jinja2>=2.9->bokeh>=1.0.0->hvplot) (2.1.5)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.8.2->pandas->hvplot) (1.16.0)
Requirement already satisfied: webencodings in /usr/local/lib/python3.10/dist-packages (from bleach->panel>=0.11.0->hvplot) (0.5.1)
Requirement already satisfied: uc-micro-py in /usr/local/lib/python3.10/dist-packages (from linkify-it-py->panel>=0.11.0->hvplot) (1.0.3)
Requirement already satisfied: mdurl~=0.1 in /usr/local/lib/python3.10/dist-packages (from markdown-it-py->panel>=0.11.0->hvplot) (0.1.2)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests->panel>=0.11.0->hvplot) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests->panel>=0.11.0->hvplot) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests->panel>=0.11.0->hvplot) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests->panel>=0.11.0->hvplot) (2024.2.2)
```

```
pip install ucimlrepo
```

```
Requirement already satisfied: ucimlrepo in /usr/local/lib/python3.10/dist-packages (0.0.6)
```

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
%matplotlib inline

from sklearn.model_selection import train_test_split
from sklearn import metrics
from sklearn.linear_model import LinearRegression
```

```
from ucimlrepo import fetch_ucirepo

# fetch dataset
automobile = fetch_ucirepo(id=10)

# data (as pandas dataframes)
X = automobile.data.features
y = automobile.data.targets

# metadata
print(automobile.metadata)

# variable information
print(automobile.variables)
```

```
. body-style:               hardtop, wagon, sedan, hatchback, convertible.\r\n  8. drive-wheels:             4wd, fwd, rwd.\r\n  9. engine-location:          front, rear.\r\n 10. wheel-base:               continuous from 86.6 120.9.\r\n 11. length:
```

```python
from ucimlrepo import fetch_ucirepo

# fetch dataset
wine = fetch_ucirepo(id=109)

# data (as pandas dataframes)
Xx = wine.data.features
yy = wine.data.targets

# metadata
print(wine.metadata)

# variable information
print(wine.variables)
```

```
{'uci_id': 109, 'name': 'Wine', 'repository_url': 'https://archive.ics.uci.edu/dataset/109/wine', 'data_url': 'https://archive.ics.uci.edu/static/public/109/data.csv', 'abstract': 'Using chemical analysis to determine the origin of wines', 'area': 'Physics
                    name     role         type demographic  \
0                  class   Target  Categorical        None
1                Alcohol  Feature   Continuous        None
2              Malicacid  Feature   Continuous        None
3                    Ash  Feature   Continuous        None
4      Alcalinity_of_ash  Feature   Continuous        None
5              Magnesium  Feature      Integer        None
6          Total_phenols  Feature   Continuous        None
7             Flavanoids  Feature   Continuous        None
8   Nonflavanoid_phenols  Feature   Continuous        None
9        Proanthocyanins  Feature   Continuous        None
10       Color_intensity  Feature   Continuous        None
11                   Hue  Feature   Continuous        None
12  0D280_0D315_of_diluted_wines  Feature   Continuous        None
13               Proline  Feature      Integer        None

   description units missing_values
0         None  None             no
1         None  None             no
2         None  None             no
3         None  None             no
4         None  None             no
5         None  None             no
6         None  None             no
7         None  None             no
8         None  None             no
9         None  None             no
10        None  None             no
11        None  None             no
12        None  None             no
13        None  None             no
```

Double-click (or enter) to edit

```python
AutoMobile = pd.concat([X,y], axis = 1)
AutoMobile
```

	price	highway-mpg	city-mpg	peak-rpm	horsepower	compression-ratio	stroke	bore	fuel-system	engine-size	...	wheel-base	engine-location	drive-wheels	body-style	num-of-doors	aspiration	fuel-type	make
0	13495.0	27	21	5000.0	111.0	9.0	2.68	3.47	mpfi	130	...	88.6	front	rwd	convertible	2.0	std	gas	alfa-romero
1	16500.0	27	21	5000.0	111.0	9.0	2.68	3.47	mpfi	130	...	88.6	front	rwd	convertible	2.0	std	gas	alfa-romero
2	16500.0	26	19	5000.0	154.0	9.0	3.47	2.68	mpfi	152	...	94.5	front	rwd	hatchback	2.0	std	gas	alfa-romero
3	13950.0	30	24	5500.0	102.0	10.0	3.40	3.19	mpfi	109	...	99.8	front	fwd	sedan	4.0	std	gas	audi
4	17450.0	22	18	5500.0	115.0	8.0	3.40	3.19	mpfi	136	...	99.4	front	4wd	sedan	4.0	std	gas	audi
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
200	16845.0	28	23	5400.0	114.0	9.5	3.15	3.78	mpfi	141	...	109.1	front	rwd	sedan	4.0	std	gas	volvo
201	19045.0	25	19	5300.0	160.0	8.7	3.15	3.78	mpfi	141	...	109.1	front	rwd	sedan	4.0	turbo	gas	volvo
202	21485.0	23	18	5500.0	134.0	8.8	2.87	3.58	mpfi	173	...	109.1	front	rwd	sedan	4.0	std	gas	volvo

```
Wine = pd.concat([Xx, yy], axis = 1)
Wine
```

	Alcohol	Malicacid	Ash	Alcalinity_of_ash	Magnesium	Total_phenols	Flavanoids	Nonflavanoid_phenols	Proanthocyanins	Color_intensity	Hue	0D280_0D315_of_diluted_wines	Proline	class
0	14.23	1.71	2.43	15.6	127	2.80	3.06	0.28	2.29	5.64	1.04	3.92	1065	1
1	13.20	1.78	2.14	11.2	100	2.65	2.76	0.26	1.28	4.38	1.05	3.40	1050	1
2	13.16	2.36	2.67	18.6	101	2.80	3.24	0.30	2.81	5.68	1.03	3.17	1185	1
3	14.37	1.95	2.50	16.8	113	3.85	3.49	0.24	2.18	7.80	0.86	3.45	1480	1
4	13.24	2.59	2.87	21.0	118	2.80	2.69	0.39	1.82	4.32	1.04	2.93	735	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
173	13.71	5.65	2.45	20.5	95	1.68	0.61	0.52	1.06	7.70	0.64	1.74	740	3
174	13.40	3.91	2.48	23.0	102	1.80	0.75	0.43	1.41	7.30	0.70	1.56	750	3
175	13.27	4.28	2.26	20.0	120	1.59	0.69	0.43	1.35	10.20	0.59	1.56	835	3
176	13.17	2.59	2.37	20.0	120	1.65	0.68	0.53	1.46	9.30	0.60	1.62	840	3
177	14.13	4.10	2.74	24.5	96	2.05	0.76	0.56	1.35	9.20	0.61	1.60	560	3

178 rows × 14 columns

Next steps: View recommended plots

Finding where and how many missing values are there in all columns of AutoMobile

```
AutoMobile.isna().sum()
```

```
price                 4
highway-mpg           0
city-mpg              0
peak-rpm              2
horsepower            2
compression-ratio     0
stroke                4
bore                  4
fuel-system           0
engine-size           0
num-of-cylinders      0
engine-type           0
curb-weight           0
height                0
width                 0
length                0
wheel-base            0
engine-location       0
drive-wheels          0
body-style            0
num-of-doors          2
aspiration            0
fuel-type             0
make                  0
normalized-losses    41
symboling             0
dtype: int64
```

```
AutoMobile.describe()
```

	price	highway-mpg	city-mpg	peak-rpm	horsepower	compression-ratio	stroke	bore	engine-size	num-of-cylinders	curb-weight	height	width	length	wheel-base	num-of-doors	normalized-losses	symboling
count	201.000000	205.000000	205.000000	203.000000	203.000000	205.000000	201.000000	201.000000	205.000000	205.000000	205.000000	205.000000	205.000000	205.000000	205.000000	203.000000	164.000000	205.000000
mean	13207.129353	30.751220	25.219512	5125.369458	104.256158	10.142537	3.255423	3.329751	126.907317	4.380488	2555.565854	53.724878	65.907805	174.049268	98.756585	3.123153	122.000000	0.834146
std	7947.066342	6.886443	6.542142	479.334560	39.714369	3.972040	0.316717	0.273539	41.642693	1.080854	520.680204	2.443522	2.145204	12.337289	6.021776	0.994841	35.442168	1.245307
min	5118.000000	16.000000	13.000000	4150.000000	48.000000	7.000000	2.070000	2.540000	61.000000	2.000000	1488.000000	47.800000	60.300000	141.100000	86.600000	2.000000	65.000000	-2.000000
25%	7775.000000	25.000000	19.000000	4800.000000	70.000000	8.600000	3.110000	3.150000	97.000000	4.000000	2145.000000	52.000000	64.100000	166.300000	94.500000	2.000000	94.000000	0.000000
50%	10295.000000	30.000000	24.000000	5200.000000	95.000000	9.000000	3.290000	3.310000	120.000000	4.000000	2414.000000	54.100000	65.500000	173.200000	97.000000	4.000000	115.000000	1.000000
75%	16500.000000	34.000000	30.000000	5500.000000	116.000000	9.400000	3.410000	3.590000	141.000000	4.000000	2935.000000	55.500000	66.900000	183.100000	102.400000	4.000000	150.000000	2.000000
max	45400.000000	54.000000	49.000000	6600.000000	288.000000	23.000000	4.170000	3.940000	326.000000	12.000000	4066.000000	59.800000	72.300000	208.100000	120.900000	4.000000	256.000000	3.000000

As there is a missing values in some rows in the columns we will use the mean of their specific column and add its mean to the values with missing value

```
Missing_val = ['price', 'peak-rpm','horsepower','stroke','bore','num-of-doors','normalized-losses']
```

```
for col in Missing_val:
  AutoMobile[col].fillna(AutoMobile[col].mean(), inplace=True)
```

Recheking if there is null/missing values

```
AutoMobile.isna().sum()
```

```
price                0
highway-mpg          0
city-mpg             0
peak-rpm             0
horsepower           0
compression-ratio    0
stroke               0
bore                 0
fuel-system          0
engine-size          0
num-of-cylinders     0
engine-type          0
curb-weight          0
height               0
width                0
length               0
wheel-base           0
engine-location      0
drive-wheels         0
body-style           0
num-of-doors         0
aspiration           0
fuel-type            0
make                 0
normalized-losses    0
symboling            0
dtype: int64
```

Checking the datatypes

```
AutoMobile.dtypes
```

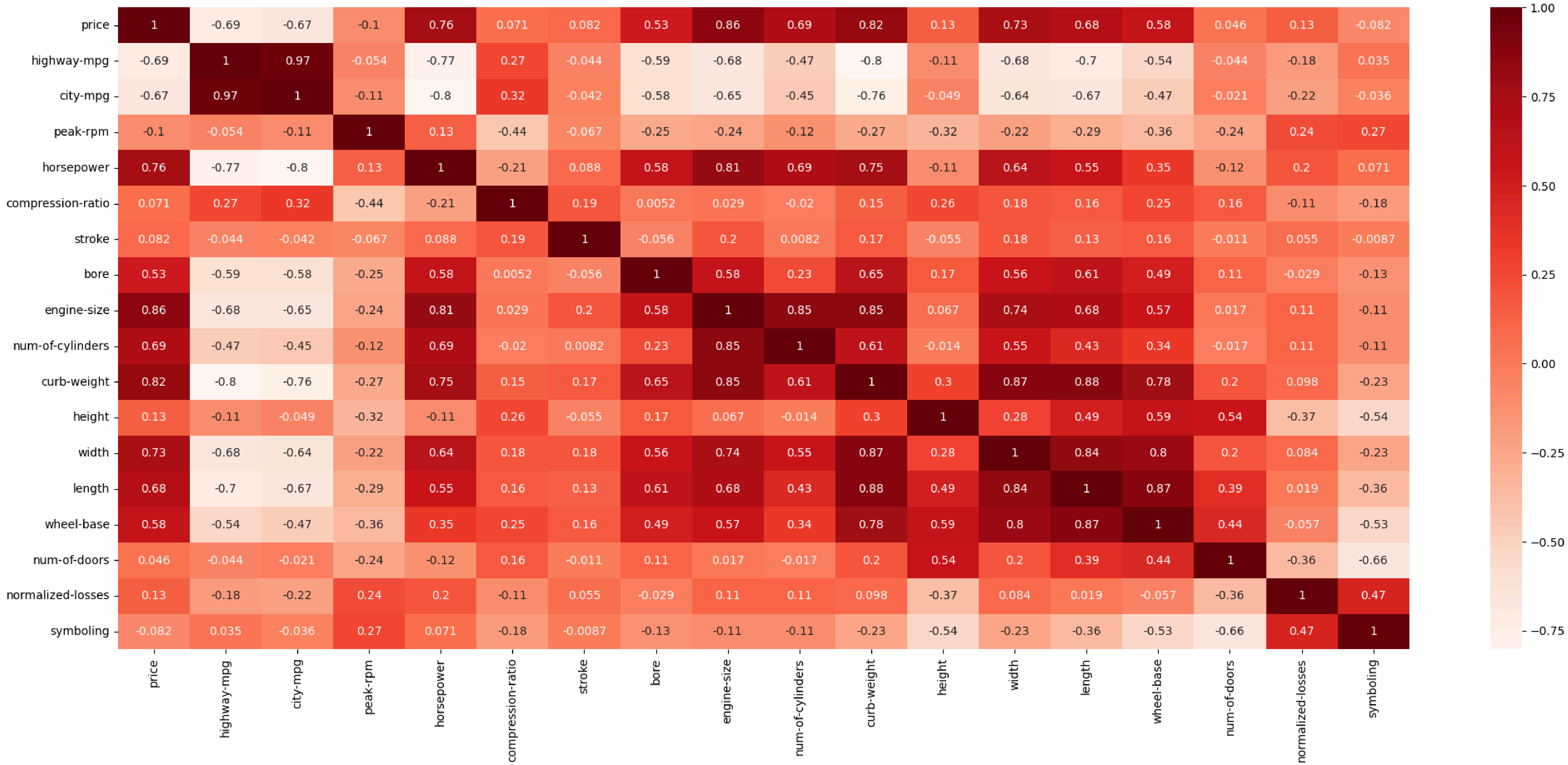
```
price                float64
highway-mpg            int64
city-mpg               int64
peak-rpm             float64
horsepower           float64
compression-ratio    float64
stroke               float64
bore                 float64
fuel-system           object
engine-size            int64
```

```
num-of-cylinders       int64
engine-type           object
curb-weight            int64
height               float64
width                float64
length               float64
wheel-base           float64
engine-location       object
drive-wheels          object
body-style            object
num-of-doors         float64
aspiration            object
fuel-type             object
make                  object
normalized-losses    float64
symboling              int64
dtype: object
```

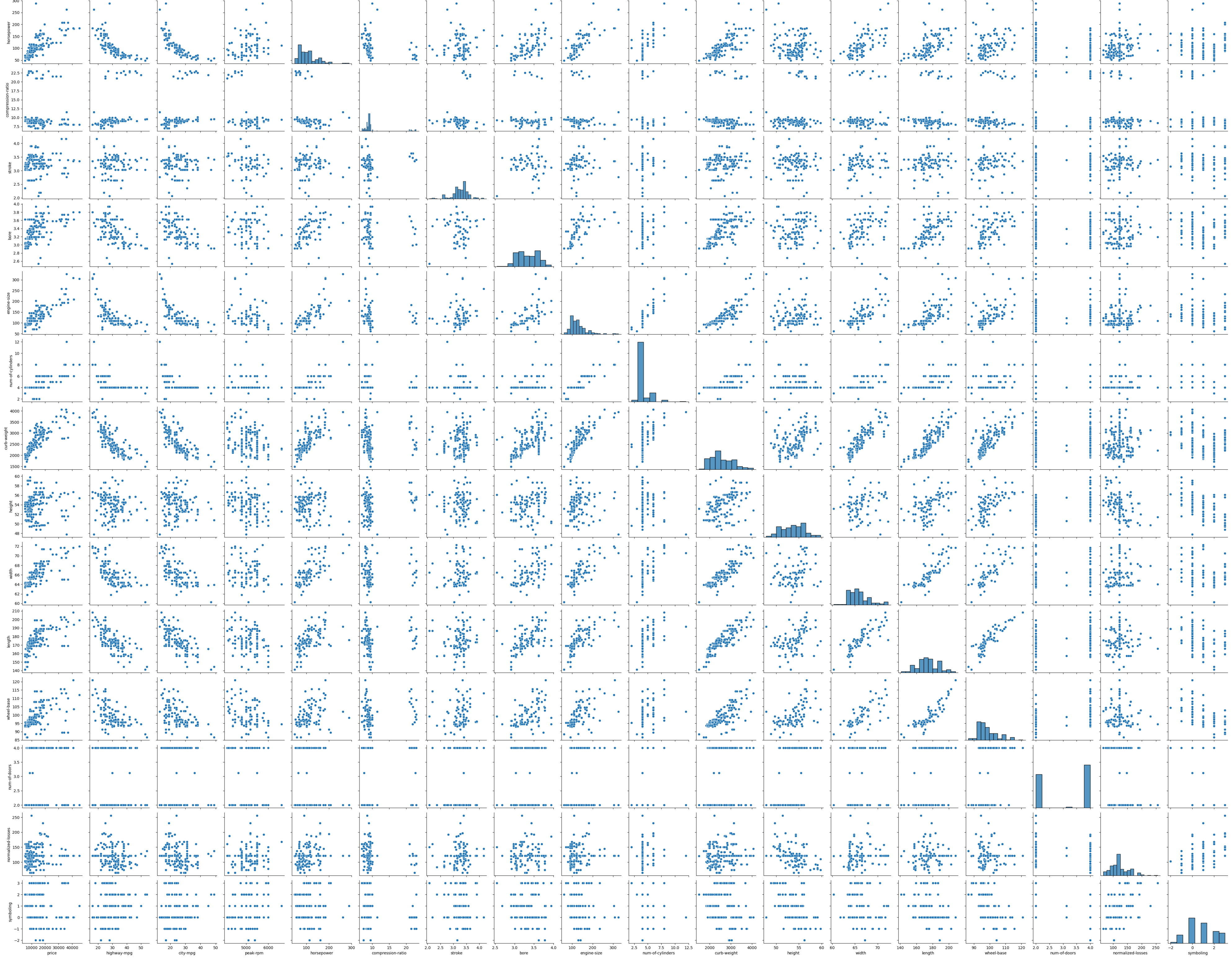
Removing the columns that have object datatype

```
AutoMob = AutoMobile.drop(['fuel-system',
                   'engine-type',
                   'engine-location',
                   'drive-wheels',
                   'body-style',
                   'aspiration',
                   'fuel-type',
                   'make'], axis = 1)
```

```
plt.figure(figsize =(25,10))
ax = sns.heatmap(AutoMob.corr(), annot = True, cmap = 'Reds')
```



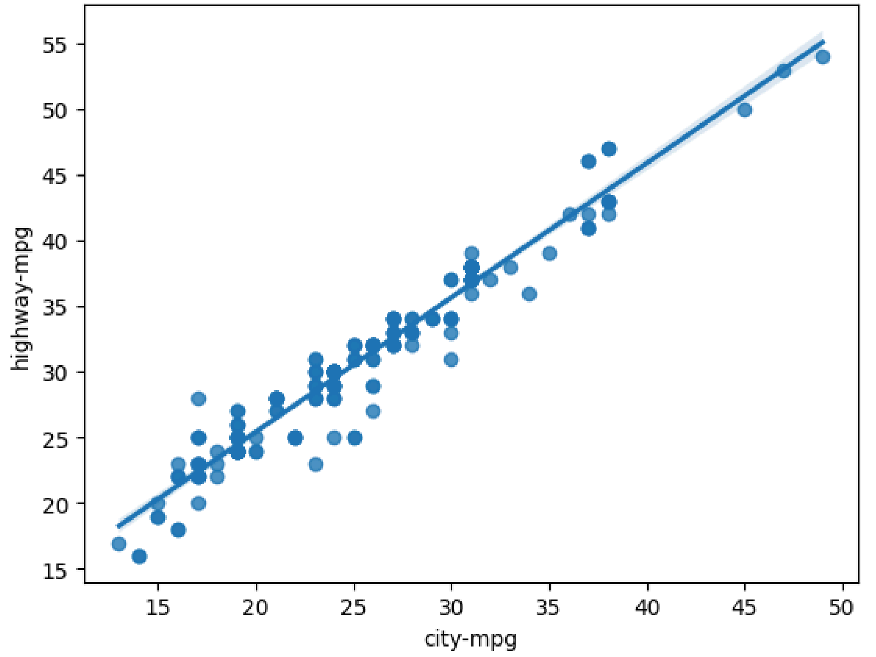

```
sns.pairplot(AutoMob)
```


horsepower
compression-ratio
stroke
bore
engine-size
num-of-cylinders
curb-weight
height
width
length
wheel-base
num-of-doors
normalized-losses
symboling
price
highway-mpg
city-mpg
peak-rpm

We will use city-mpg and the highway-mpg since it have the highest correlation and we can see that if city-mpg increases the highway-mpg also increases since it both cover distance

```
sns.regplot(x = AutoMob['city-mpg'], y = AutoMob['highway-mpg'])
```

<Axes: xlabel='city-mpg', ylabel='highway-mpg'>



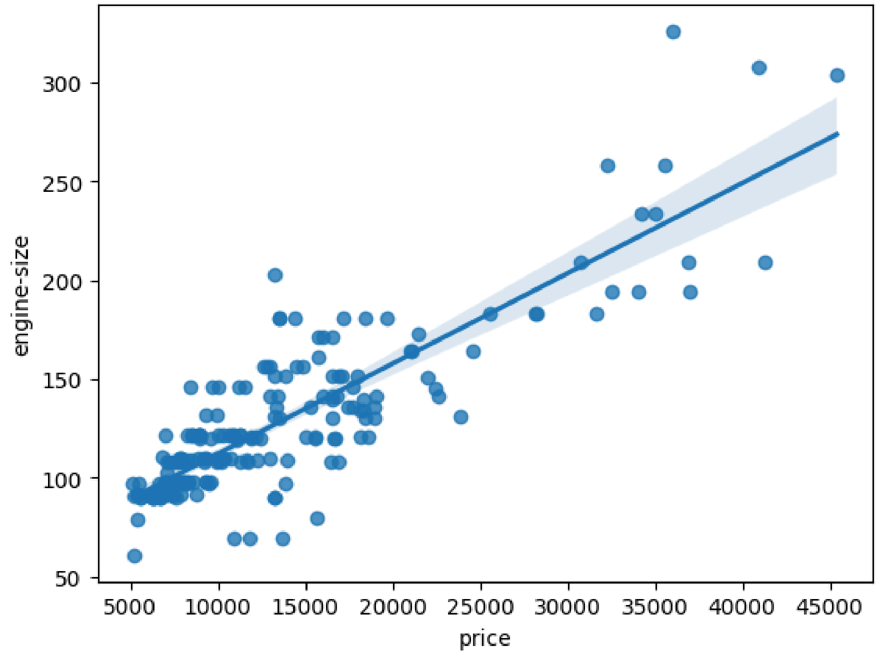

```
AutoMob['highway-mpg'].corr(AutoMob['city-mpg'])
```

0.9713370423425061

We will be using another sample

```
sns.regplot(x = AutoMob['price'], y = AutoMob['engine-size'])
```

<Axes: xlabel='price', ylabel='engine-size'>



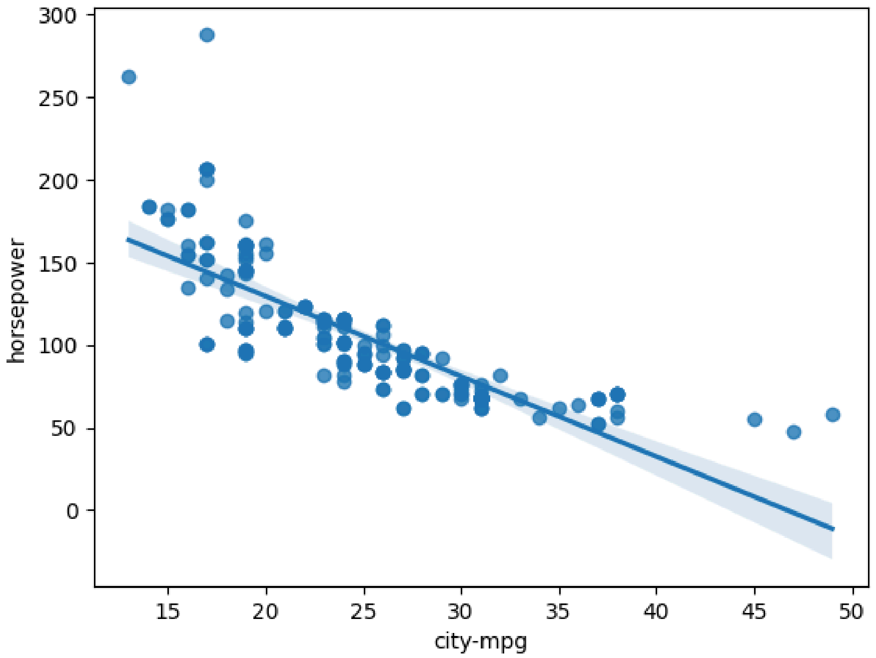

```
AutoMob['price'].corr(AutoMob['engine-size'])
```

0.8617522436859719

Using city-mpg and horsepower we can see that both have low correlation where if horsepower increases the city-mpg doesn't increase

```
sns.regplot(x = AutoMob['city-mpg'], y = AutoMob['horsepower'])
```

```
<Axes: xlabel='city-mpg', ylabel='horsepower'>
```



Double-click (or enter) to edit

```
AutoMob['horsepower'].corr(AutoMob['city-mpg'])
```

```
-0.8031621465372332
```

Wine

```
Wine
```

	Alcohol	Malicacid	Ash	Alcalinity_of_ash	Magnesium	Total_phenols	Flavanoids	Nonflavanoid_phenols	Proanthocyanins	Color_intensity	Hue	0D280_0D315_of_dilut
0	14.23	1.71	2.43	15.6	127	2.80	3.06	0.28	2.29	5.64	1.04	
1	13.20	1.78	2.14	11.2	100	2.65	2.76	0.26	1.28	4.38	1.05	
2	13.16	2.36	2.67	18.6	101	2.80	3.24	0.30	2.81	5.68	1.03	
3	14.37	1.95	2.50	16.8	113	3.85	3.49	0.24	2.18	7.80	0.86	
4	13.24	2.59	2.87	21.0	118	2.80	2.69	0.39	1.82	4.32	1.04	
...	...	...	...	...	...	...	...	...	...	...	...	
173	13.71	5.65	2.45	20.5	95	1.68	0.61	0.52	1.06	7.70	0.64	
174	13.40	3.91	2.48	23.0	102	1.80	0.75	0.43	1.41	7.30	0.70	
175	13.27	4.28	2.26	20.0	120	1.59	0.69	0.43	1.35	10.20	0.59	
176	13.17	2.59	2.37	20.0	120	1.65	0.68	0.53	1.46	9.30	0.60	
177	14.13	4.10	2.74	24.5	96	2.05	0.76	0.56	1.35	9.20	0.61	

178 rows × 14 columns

Next steps: View recommended plots

```
Wine.describe()
```

	Alcohol	Malicacid	Ash	Alcalinity_of_ash	Magnesium	Total_phenols	Flavanoids	Nonflavanoid_phenols	Proanthocyanins	Color_intensity	Hue	0D280_0D315_of_diluted_wines	Proline	class
count	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000
mean	13.000618	2.336348	2.366517	19.494944	99.741573	2.295112	2.029270	0.361854	1.590899	5.058090	0.957449	2.611685	746.893258	1.938202
std	0.811827	1.117146	0.274344	3.339564	14.282484	0.625851	0.998859	0.124453	0.572359	2.318286	0.228572	0.709990	314.907474	0.775035
min	11.030000	0.740000	1.360000	10.600000	70.000000	0.980000	0.340000	0.130000	0.410000	1.280000	0.480000	1.270000	278.000000	1.000000
25%	12.362500	1.602500	2.210000	17.200000	88.000000	1.742500	1.205000	0.270000	1.250000	3.220000	0.782500	1.937500	500.500000	1.000000
50%	13.050000	1.865000	2.360000	19.500000	98.000000	2.355000	2.135000	0.340000	1.555000	4.690000	0.965000	2.780000	673.500000	2.000000
75%	13.677500	3.082500	2.557500	21.500000	107.000000	2.800000	2.875000	0.437500	1.950000	6.200000	1.120000	3.170000	985.000000	3.000000
max	14.830000	5.800000	3.230000	30.000000	162.000000	3.880000	5.080000	0.660000	3.580000	13.000000	1.710000	4.000000	1680.000000	3.000000

As there is no missing value we don't need to use the mean

```
Wine.isna().sum()
```

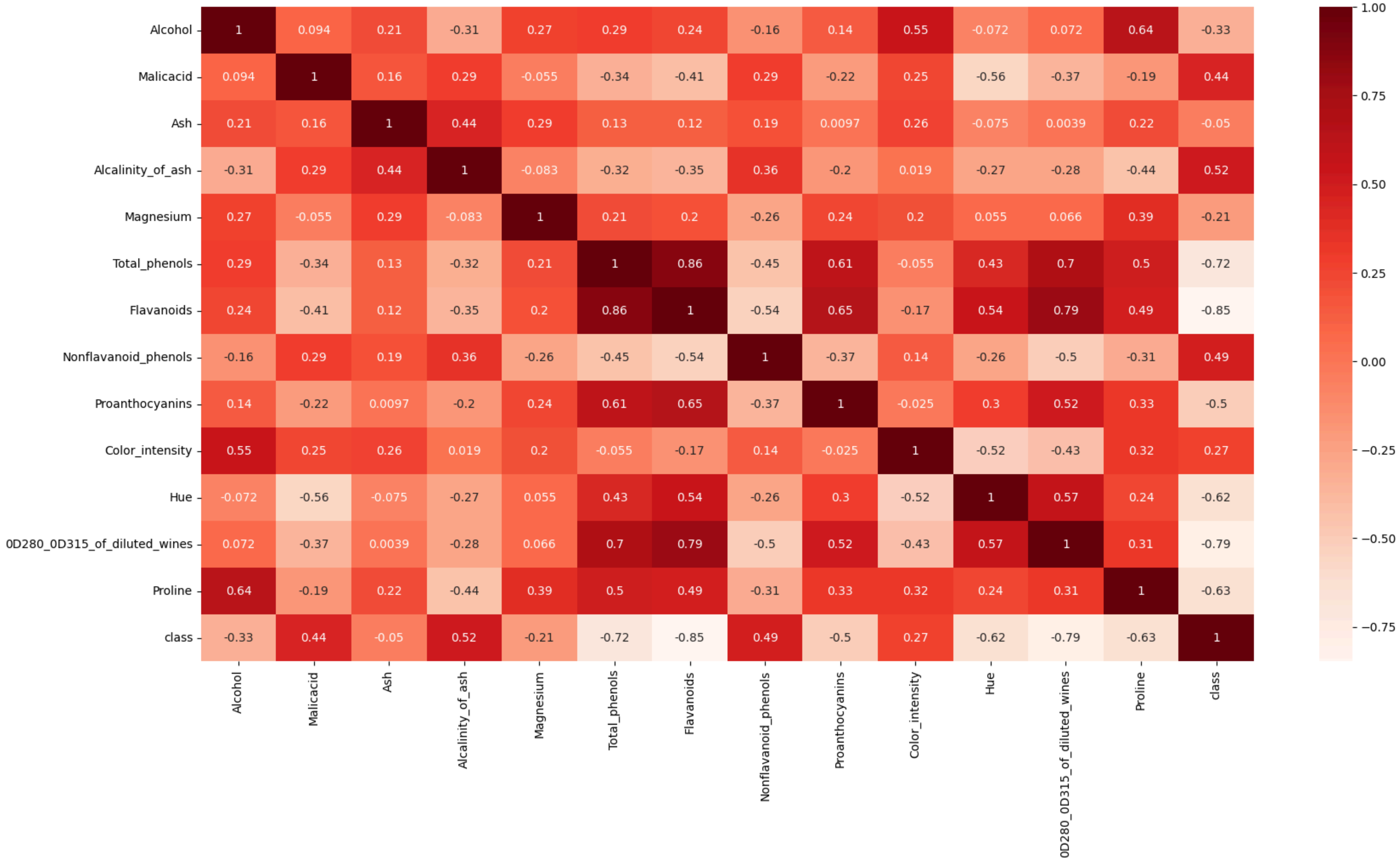
```
Alcohol                         0
Malicacid                       0
Ash                             0
Alcalinity_of_ash               0
Magnesium                       0
Total_phenols                   0
Flavanoids                      0
Nonflavanoid_phenols            0
Proanthocyanins                 0
Color_intensity                 0
Hue                             0
0D280_0D315_of_diluted_wines    0
Proline                         0
class                           0
dtype: int64
```

As there is no object datatype we won't be needing to delete columns

```
Wine.dtypes
```

```
Alcohol                         float64
Malicacid                       float64
Ash                             float64
Alcalinity_of_ash               float64
Magnesium                         int64
Total_phenols                   float64
Flavanoids                      float64
Nonflavanoid_phenols            float64
Proanthocyanins                 float64
Color_intensity                 float64
Hue                             float64
0D280_0D315_of_diluted_wines    float64
Proline                           int64
class                             int64
dtype: object
```

```
plt.figure(figsize =(20,10))
ax = sns.heatmap(Wine.corr(), annot = True, cmap = 'Reds')
```




```
sns.pairplot(Wine)
```