```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from statsmodels.graphics.regressionplots import influence_plot
import statsmodels.formula.api as smf
import numpy as np

import pandas as pd

#step2
from google.colab import files
uploaded = files.upload()

#step3
file_name = list(uploaded.keys())[0]
cars = pd.read_csv(file_name)
```

```
<IPython.core.display.HTML object>

Saving Cars.csv to Cars.csv
```

```
cars.shape
```

```
(81, 5)
```

```
isna = cars.isna().sum()
isna
```

```
HP     0
MPG    0
VOL    0
SP     0
WT     0
dtype: int64
```

```
#correlation
cars.corr()
```


```
{"summary":"{\n  \"name\": \"cars\",\n  \"rows\": 5,\n  \"fields\": [\n    {\n      \"column\": \"HP\",\n      \"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 0.7232487955377062,\n        \"min\": -0.7250383497637914,\n        \"max\": 1.0,\n        \"num_unique_values\": 5,\n        \"samples\": [\n          -0.7250383497637914,\n          0.07651306534492211,\n          0.07745947360036072\n        ],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"MPG\",\n      \"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 0.7287325628832262,\n        \"min\": -0.7250383497637914,\n        \"max\": 1.0,\n        \"num_unique_values\": 5,\n        \"samples\": [\n          1.0,\n          -0.5267590900278917,\n          -0.5290565802560923\n        ],\n        \"semantic_type\": \"\",\n
```

\"description\": \"\"\n   }\n   },\n   {\n   \"column\":
\"VOL\",\n   \"properties\": {\n   \"dtype\": \"number\",\n
\"std\": 0.6615123170839469,\n   \"min\": -0.5290565802560923,\n
\"max\": 1.0,\n   \"num_unique_values\": 5,\n   \"samples\":
[\n   -0.5290565802560923,\n   0.999203080186856,\n
1.0\n   ],\n   \"semantic_type\": \"\",\n
\"description\": \"\"\n   }\n   },\n   {\n   \"column\":
\"SP\",\n   \"properties\": {\n   \"dtype\": \"number\",\n
\"std\": 0.7065122396814414,\n   \"min\": -0.6871246127261932,\n
\"max\": 1.0,\n   \"num_unique_values\": 5,\n   \"samples\":
[\n   -0.6871246127261932,\n   0.10243919098077894,\n
0.1021700095142569\n   ],\n   \"semantic_type\": \"\",\n
\"description\": \"\"\n   }\n   },\n   {\n   \"column\":
\"WT\",\n   \"properties\": {\n   \"dtype\": \"number\",\n
\"std\": 0.6608343322683942,\n   \"min\": -0.5267590900278917,\n
\"max\": 1.0,\n   \"num_unique_values\": 5,\n   \"samples\":
[\n   -0.5267590900278917,\n   1.0,\n
0.999203080186856\n   ],\n   \"semantic_type\": \"\",\n
\"description\": \"\"\n   }\n   }\n]\n}","type":"dataframe"}

```
#dataframes
cars_new = cars.iloc[:,1:]
cars_new.head()
```

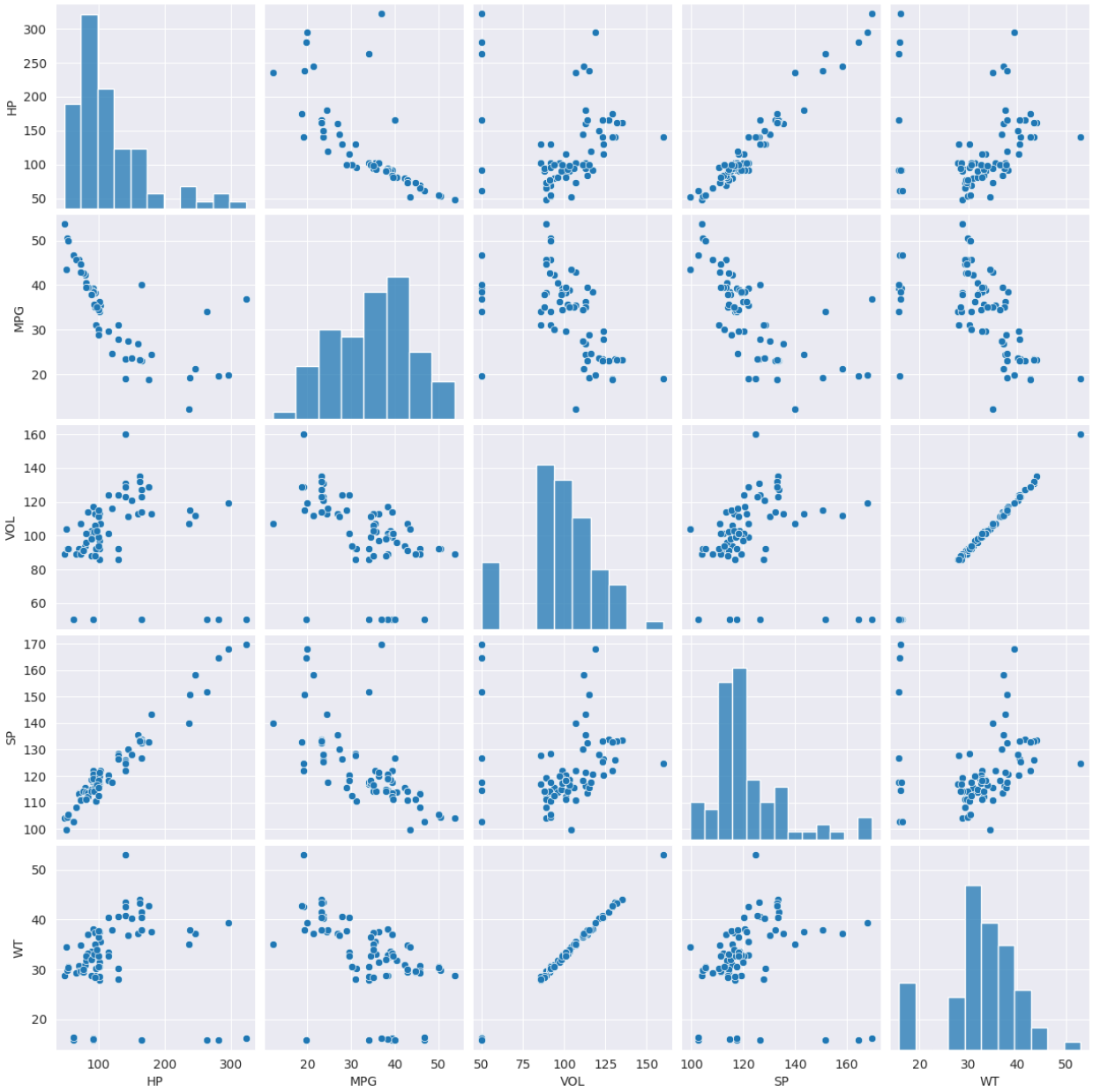
{"summary":"{\n \"name\": \"cars_new\",\n \"rows\": 81,\n
\"fields\": [\n   {\n   \"column\": \"MPG\",\n
\"properties\": {\n   \"dtype\": \"number\",\n   \"std\":
9.131444731795982,\n   \"min\": 12.10126289,\n   \"max\":
53.70068138,\n   \"num_unique_values\": 50,\n   \"samples\":
[\n   38.31060597,\n   23.10317168,\n
27.85625194\n   ],\n   \"semantic_type\": \"\",\n
\"description\": \"\"\n   }\n   },\n   {\n   \"column\":
\"VOL\",\n   \"properties\": {\n   \"dtype\": \"number\",\n
\"std\": 22,\n   \"min\": 50,\n   \"max\": 160,\n
\"num_unique_values\": 34,\n   \"samples\": [\n   98,\n
102,\n   127\n   ],\n   \"semantic_type\": \"\",\n
\"description\": \"\"\n   }\n   },\n   {\n   \"column\":
\"SP\",\n   \"properties\": {\n   \"dtype\": \"number\",\n
\"std\": 14.18143157452861,\n   \"min\": 99.56490661,\n
\"max\": 169.5985128,\n   \"num_unique_values\": 68,\n
\"samples\": [\n   115.5765794,\n   113.8291446,\n
113.1853528\n   ],\n   \"semantic_type\": \"\",\n
\"description\": \"\"\n   }\n   },\n   {\n   \"column\":
\"WT\",\n   \"properties\": {\n   \"dtype\": \"number\",\n
\"std\": 7.492812997393198,\n   \"min\": 15.71285853,\n
\"max\": 52.99775236,\n   \"num_unique_values\": 81,\n
\"samples\": [\n   37.04235003,\n   28.7620589,\n
32.6758277\n   ],\n   \"semantic_type\": \"\",\n
\"description\": \"\"\n   }\n   }\n]\
n}","type":"dataframe","variable_name":"cars_new"}

```
cars.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 81 entries, 0 to 80
Data columns (total 5 columns):
 #   Column  Non-Null Count  Dtype  
---  ------  --------------  -----  
 0   HP      81 non-null     int64  
 1   MPG     81 non-null     float64
 2   VOL     81 non-null     int64  
 3   SP      81 non-null     float64
 4   WT      81 non-null     float64
dtypes: float64(3), int64(2)
memory usage: 3.3 KB
```

```
sns.set_style(style='darkgrid')
sns.pairplot(cars)
```

```
<seaborn.axisgrid.PairGrid at 0x7f842a0f5350>
```




```
import statsmodels.formula.api as smf
model = smf.ols('MPG~WT+VOL+SP+HP',data=cars).fit()

model.params
```

```
Intercept    30.677336
WT            0.400574
VOL          -0.336051
SP            0.395627
HP           -0.205444
dtype: float64
```